On Structured Object Representations: Benefits for Autonomous Agents and Real-World Discovery

Dissertation

der Mathematisch-Naturwissenschaftlichen Fakultät
der Eberhard Karls Universität Tübingen
zur Erlangung des Grades eines
Doktors der Naturwissenschaften
(Dr. rer. nat.)

vorgelegt von
Maximilian Johannes Seitzer
aus Bamberg

Tübingen
2024

Gedruckt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der Eberhard Karls Universität Tübingen.

Tag der mündlichen Qualifikation:	14.10.2024
Dekan:	Prof. Dr. Thilo Stehle
1. Berichterstatter:	Prof. Dr. Georg Martius
2. Berichterstatter:	Prof. Dr. Martin Butz
3. Berichterstatter:	Assoc. Prof. Dr. Efstratios Gavves

Für Alex und Ben

Contents

List of Figures

Overview of Publications

This dissertation is based on the following publications:

Andrii Zadaianchuk*, **Maximilian Seitzer***, and Georg Martius (2021). "Self-Supervised Visual Reinforcement Learning with Object-Centric Representations". In: *International Conference on Learning Representations (ICLR)*. *equal contribution

A summary of this publication can be found in Chap. 4.

Contributions: AZ and MS contributed equally and share the first author position. AZ and GM initiated the project, and AZ implemented and ran the initial SCALOR experiments. The design, implementation, and experiments of the SMORL agent was in equal parts due to AZ and MS, with input from GM. AZ and MS wrote the first draft with input from GM, and all authors contributed to the final version of the paper.

Maximilian Seitzer, Bernhard Schölkopf, and Georg Martius (2021). "Causal Influence Detection for Improving Efficiency in Reinforcement Learning". In: *Conference on Neural Information Processing Systems (NeurIPS)*

A summary of this publication can be found in Chap. 5.

Contributions: The project was initiated by MS and GM, and MS led the project. MS developed the theory, implemented, ran, and analyzed the experiments, with input from GM and BS. MS wrote the first draft with input from GM, and all authors contributed to the final version of the paper.

Maximilian Seitzer, Max Horn, Andrii Zadaianchuk, Dominik Zietlow, Tianjun Xiao, Carl-Johann Simon-Gabriel, Tong He, Zheng Zhang, Bernhard Schölkopf, Thomas Brox, and Francesco Locatello (2023). "Bridging the Gap to Real-World Object-Centric Learning". In: *International Conference on Learning Representations (ICLR)*

A summary of this publication can be found in Chap. 6.

Contributions: The project was initiated by MS, MH and DZ, MS led the project, and FL was senior advisor. The direction of the project was shaped by MS, MH, AZ, DZ, CJSG, TB, and FL, with advise from TX, TH, ZZ, and BS. MS and MH implemented the codebase. DZ contributed datasets. MS ran and analyzed all experiments besides the SLATE baseline, ran by AZ. MS, MH, AZ, DZ, and FL wrote the first draft. MS designed all figures besides the model figure, designed by DZ. MS, MH, AZ, DZ, CJSG, TB, BS, and FL contributed to the final version of the paper.

Andrii Zadaianchuk*, **Maximilian Seitzer***, and Georg Martius (2023). “Object-Centric Learning for Real-World Videos by Predicting Temporal Feature Similarities”. In: *Conference on Neural Information Processing Systems (NeurIPS)*. *equal contribution

A summary of this publication can be found in Chap. 7.

Contributions: AZ and MS contributed equally and share the first author position. AZ and MS initiated and led the project. MS implemented the initial codebase, with contributions from AZ. MS and AZ designed, ran, and analyzed the exploratory experiments. MS and AZ designed the final experiments with input from GM which AZ ran and analyzed. AZ and MS wrote the first draft with input from GM, and all authors contributed to the final version of the paper.

Other Work

During my PhD, I worked on the following publications not included in this thesis:

Maximilian Seitzer, Arash Tavakoli, Dimitrije Antic, and Georg Martius (2022). “On the Pitfalls of Heteroscedastic Uncertainty Estimation with Probabilistic Neural Networks”. In: *International Conference on Learning Representations (ICLR)*

Maximilian Seitzer, Sjoerd van Steenkiste, Thomas Kipf, Klaus Greff, and Mehdi S. M. Sajjadi (2024). “DyST: Towards Dynamic Neural Scene Representations on Real-World Videos”. In: *International Conference on Learning Representations (ICLR)*

and the following manuscript:

Aniket Didolkar*, Andrii Zadaianchuk, Anirudh Goyal, Mike Mozer, Yoshua Bengio, Georg Martius, and **Maximilian Seitzer*** (2024). “Zero-Shot Object-Centric Representation Learning”. In: *arXiv:2408.09162*. *equal contribution

Abstract

This thesis is motivated by the shortcomings of contemporary AI models compared to human cognitive abilities. Its central hypothesis is that these limitations arise from the model's inability to learn and maintain structured object representations of the world. The overarching research question is how to integrate such representations with today's leading AI models, specifically neural networks. This thesis is concerned with a class of methods that attempt to solve this problem, called object-centric representation learning. In particular, its contributions are twofold: (1) enabling the unsupervised discovery of objects in complex, real-world data, and (2) demonstrating the benefits of object-centric representations for autonomous agent learning.

A significant limitation of object-centric representation learning was that it only successfully discovered objects on simple, synthetic datasets. The major contribution of this thesis is an approach that overcomes this limitation and, for the first time, allows models to decompose natural images and videos into object-centric representations. To achieve this, the thesis introduces mechanisms that integrate semantic inductive biases into object-centric models. These mechanisms work by training the model to predict targets derived from pre-trained semantic features, which can be obtained from self-supervised learning methods in a fully unsupervised way. Specifically for video data, an alternative prediction target encoding temporal correlations between pre-trained features is proposed, introducing an additional inductive bias toward grouping objects by consistent motion. The resulting models achieve state-of-the-art results and scale to real-world datasets such as PASCAL VOC, COCO, and YouTube-VIS.

As further contributions, this thesis presents two case studies that illustrate the advantages of object-centric representations for applications, specifically in the domain of autonomous agent learning with reinforcement learning (RL). In the first case study, an algorithm for self-supervised RL is introduced; leveraging object-centric representations, the agent learns to compose simple sub-goals to accomplish tasks in complex, multi-object environments. In the second case study, a measure of causal influence between agent and objects is derived; this measure can be integrated into RL algorithms in various ways to significantly improve their sample efficiency. The results highlight the potential of object-centric representations as an inductive bias for agent systems in the physical world, demonstrating important properties such as interpretability, generalization, and data efficiency.

Through these contributions, the thesis advances the field of object-centric representation learning, demonstrating its potential and paving the way for practical real-world applications.

Zusammenfassung

Diese Dissertation ist von den Unzulänglichkeiten zeitgenössischer KI-Modelle im Vergleich zu den kognitiven Fähigkeiten des Menschen motiviert. Ihre zentrale Hypothese ist, dass diese Einschränkungen aus der Unfähigkeit der Modelle resultieren, strukturierte Objektrepräsentationen der Welt zu lernen und aufrechtzuerhalten. Die übergeordnete Forschungsfrage ist, wie solche Repräsentationen in die heutigen führenden KI-Modelle, insbesondere neuronale Netzwerke, integriert werden können. Diese Dissertation befasst sich mit einer Klasse von Methoden, die versuchen, dieses Problem zu lösen, genannt objektzentrisches Repräsentationslernen. Dabei liefert sie zwei Beiträge: (1) die unüberwachte Entdeckung von Objekten in komplexen, realen Daten zu ermöglichen und (2) die Vorteile objektzentrischer Repräsentationen für das Lernen autonomer Agenten aufzuzeigen.

Eine bedeutende Einschränkung des objektzentrischen Repräsentationslernens war, dass es Objekte nur in einfachen, synthetischen Datensätzen erfolgreich entdeckte. Der Hauptbeitrag dieser Dissertation ist ein Ansatz, der diese Einschränkung überwindet und es erstmals ermöglicht, natürliche Bilder und Videos in objektzentrische Repräsentationen zu zerlegen. Um dies zu erreichen, führt die Dissertation Mechanismen ein, die semantische "inductive biases" in objektzentrische Modelle integriert. Diese Mechanismen funktionieren, indem das Modell darauf trainiert wird, Ziele vorherzusagen, die aus vortrainierten semantischen Repräsentationen abgeleitet werden, welche durch selbstüberwachtes Lernen gewonnen werden können. Speziell für Videodaten wird ein alternatives Prädiktionsziel vorgeschlagen, das zeitliche Korrelationen zwischen vortrainierten Repräsentationen enkodiert und einen zusätzlichen "inductive bias" hin zu einer Gruppierung von Objekten durch konsistente Bewegung einführt. Die resultierenden Modelle erreichen erstklassige Ergebnisse und skalieren auf reale Datensätze wie PASCAL VOC, COCO und YouTube-VIS.

Als weitere Beiträge präsentiert diese Dissertation zwei Fallstudien, die die Vorteile objektzentrischer Repräsentationen für Anwendungen veranschaulichen, insbesondere im Bereich des Lernens autonomer Agenten mit Reinforcement Learning (RL). In der ersten Fallstudie wird ein Algorithmus für selbstüberwachtes RL vorgestellt; durch die Nutzung objektzentrischer Repräsentationen lernt der Agent, einfache Teilziele zu kombinieren, um Aufgaben in komplexen Umgebungen mit mehreren Objekten zu erfüllen. In der zweiten Fallstudie wird ein Maß für den kausalen Einfluss zwischen Agent und Objekten abgeleitet; dieses Maß kann auf verschiedenste Weisen in RL-Algorithmen integriert werden, um deren Dateneffizienz erheblich zu verbessern. Die Ergebnisse heben das Potenzial objektzentrischer Repräsentationen als "inductive bias" für Agenten in der physischen Welt hervor und demonstrieren wichtige Eigenschaften wie Interpretierbarkeit, Generalisierung und Dateneffizienz.

Durch diese Beiträge treibt die Dissertation das Feld des objektzentrischen Repräsentationslernens voran, demonstriert dessen Potenzial und ebnet den Weg für praktische Anwendungen in der realen Welt.

Acknowledgements

As is tradition, the first acknowledgement must go out to one's advisor. And what an advisor it is! Georg, I consider myself extremely lucky to have eventually ended up in your group. You were always understanding and supportive of my situation, provided ample excellent scientific guidance, and gave me the freedom to explore my interests. And of course, you were ready to jump in when things got hot. I will always fondly remember the weekend where we decided to write a paper on Friday and submit it on Monday (the reader may guess which one that is).

I would like to extend my gratitude to the members of my thesis advisory committee for their time and support: Bernhard Schölkopf, Matthias Hein, and Wieland Brendel. Furthermore, a huge "thank you" to Martin Butz and Efstratios Gavves for agreeing to review my PhD thesis and gracefully overlooking my arduous timeline. I also appreciate Andreas Geiger for agreeing to chair my PhD defense.

It was sheer coincidence to start my PhD journey by sharing an office with a person who shared so many of my research interests and views. Andrii is the best partner for scientific self-play I could have wished for, and I very much hope that we continue the journey we started together in some form. Outside of science, he of course also became a good friend.

Then I want to thank all the former and present members of the Autonomous Learning group — my PhD was made significantly nicer and enjoyable by being part of this great group of people! Here, again, it is Georg who I must thank for forming the collaborative and easy-going spirit that is so characteristic for the group. A special thanks goes out to Cansu, Nuria, and Andrii, who helped me by proofreading and providing feedback on this manuscript. Thanks to all the staff at the MPI-IS, especially to the great people tending to the cluster.

I was fortunate to have had two great internships during my PhD. Thank you to the people at Amazon Tübingen: Dominik Zietlow, CJ Simon-Gabriel, Francesco Locatello, and Thomas Brox, and especially Max Horn for mentoring me. Many thanks also to the people I worked with at Google Brain Berlin, namely Alexey Dosovitskiy, Aravindh Mahendran, Thomas Kipf, Sjoerd van Steenkiste, and in particular, Mehdi Sajjadi and Klaus Greff for mentoring me.

Last but not least, I thank my family and friends for supporting me during this time and before. This especially includes my parents: my mother, for being an always supportive source of advise & solace, and my father, for inadvertently setting me on this path very early on. Then, my grandfather, for being my academic idol whose accomplishments were always inspiring me, and my grandmother, for cheering me on since I was little. Very important for the completion of this thesis were multiple babysitting trips made by Paul, Brix, Babsy, Sina, Zoë, Naïma,

Jonas, and Sophia — I would like to wholeheartedly thank them all for jumping to our support! Jonas also deserves a special shout-out for lending me his keen eye for design to improve the layout of this thesis.

Then finally, the two people who missed me the most during this time, but still unconditionally supported me until the end: my son Ben and my wife Leonie. Ben, thanks for pulling me out my office when you have had enough of it. Leonie, thanks for everything — the patience, the love, the care — I am not sure how I can make it up to you, but I will certainly try.

Originality

This thesis is my original work. For its writing, I utilized AI tools (specifically GPT-4o) to check for grammatical errors and improve the quality and readability of the text. All drafts were written by myself. This is with the exception of the German translation of the abstract, which was first translated from the English abstract by GPT-4o and then edited by myself. All AI-generated text was carefully checked for correctness. Furthermore, I utilized image generation with DALL·E 3; the images are annotated as such in the text. Finally, parts of this manuscript have been proofread by colleagues.

Chapter 1

Introduction

The world around us is full of *structure*. From the microscopic — atoms, molecules, cells — to the macroscopic — biosystems, weather patterns, geologic formations — each level of observation is rich in structure. Most important for us humans, the world we[1] live in, sense, and interact with, is highly structured in terms of *objects*[2]: animals, plants, vehicles, houses, furniture, tools, etc. We are surrounded by objects, natural and artificial.

As a consequence, the sensory data we receive from the outside — whether visual, auditory, or even tactile — is imbued with this structure. Astonishingly, in our early years, our minds pick up on the structure and mirror it in the mental models we are building of the world. We learn to perceive the world in terms of objects, events, actions, and relationships (Spelke and Kinzler, 2007). Beyond perception, this extends to the higher functions of our cognition: *language* is structured (Chomsky, 2002), and *thought* is conjectured to have a structured, symbolic basis as well (Fodor, 1975). All the acts of reasoning that humans do effortlessly, such as simulating counterfactual scenarios in our mind, forming analogies and abstractions, understanding new situations, are believed to be grounded in mental structures that match the physical reality (Johnson-Laird, 2010).

For artificial intelligence (AI) systems to perceive, reason about, and act in our human world, it appears natural that they, too, should *structure their internal representations in terms of the external structures they represent*. This was the central idea behind the AI paradigm of *symbolism*[3] (Newell and Simon, 1976). However, it eventually became apparent that manually creating a symbolic description of the world, with all its complexities, nuances and inconsistencies, was too hard a task (Russell and Norvig, 2020). A particular problem is grounding the symbols in low-level sensory inputs, which is required for AI systems to interact with the physical world.

Instead, the dominant approach to AI today is based on statistical machine learning (ML), in particular the paradigm of *connectionism*,[4] which eschews symbols in favor of *distributed representations learned from data* (Bengio et al., 2013). This proved to be especially advantageous for learning directly from high-dimensional sensory data. But ma-

[1] In this thesis, I will mostly use "*we*" in the inclusive sense to refer to the reader and myself, but sometimes also in the exclusive sense, when referring to my co-authors and myself. It should, hopefully, always be clear from context.

[2] The notion of "objects" is the central concept in this thesis, and I will discuss it in detail in Chap. 2. For now, let us think of objects in terms of independent entities existing in the physical world.

[3] Also referred to as "good old-fashioned AI" (Haugeland, 1985).

[4] Today mostly known as deep learning (Goodfellow et al., 2016).

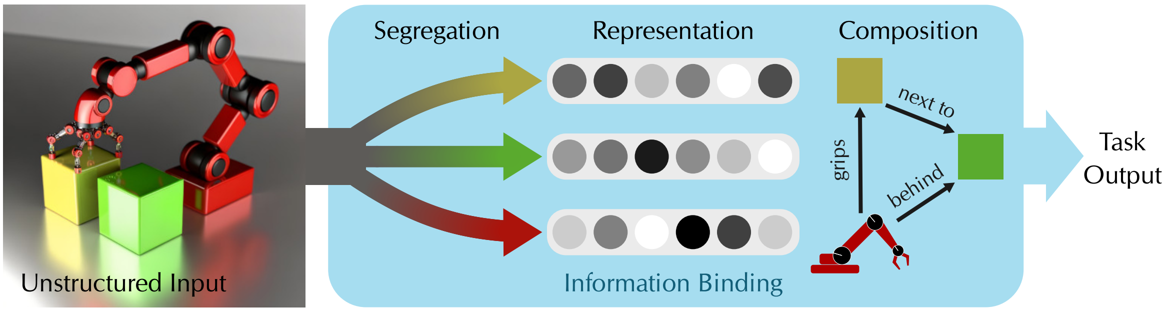


Figure 1.1: The binding problem. How can neural models dynamically combine information from unstructured inputs into a structured representation? Object-centric representation learning aims to address this question by designing mechanisms and inductive biases that allow for (1) *segregating* high-dimensional sensory data into meaningful entities; (2) *representing* information in a modular, symbol-like form; and (3) *composing* structures to solve tasks of interest and enable systematic generalization. This thesis makes contributions to the first and last aspect. Figure design adopted from Greff et al. (2020, Fig. 2), robot image created with DALL·E 3.

chine learning may also have its limits: so far, human-level, systematic generalization remains elusive. Ultimately, being based on statistical learning, the model learns only to the level required to explain the data, without necessarily capturing the underlying concepts.[5] Any further out-of-distribution generalization is inevitably based on *inductive biases* (Mitchell, 1980) — indicating that the inductive biases necessary to achieve human-level generalization are still missing.

What, then, is the missing piece? One position is that it is the ability to *discover, represent, and process* symbol-like structures in high-dimensional data in an autonomous manner (Lake et al., 2017; Greff et al., 2020; Schölkopf et al., 2021; Goyal and Bengio, 2022; Smolensky et al., 2022; Hinton, 2023). This is consistent with the only known learning system that exhibits systematic generalization, the human brain. But to avoid repeating the failures of symbolic AI, rather than engineering such a model from the ground-up, this ability should be directly integrated into the most successful AI systems of today: neural networks. In other words, the question is what kind of inductive biases empower neural models to learn and process structured representations from unstructured data. Greff et al. (2020) call this the "*binding problem of artificial neural networks*", and divide it into three aspects: (1) *segregation*, (2) *representation*, and (3) *composition* (see Fig. 1.1).

In this thesis, I focus on a certain class of structured representations — *object-centric representations* — capturing and isolating objects in visual scenes into a symbol-like structure. In particular, I will investigate how neural networks can learn such representations from data *without human supervision*. This problem, usually referred to as *object-centric representation learning*, has been the subject of study in recent years, growing in popularity.[6] Besides their inherent interpretability, object-centric representations have been shown to exhibit systematic generalization (Dittadi et al., 2022; Wiedemer et al., 2024), robustness to distribution shifts (Dittadi et al., 2022; Yoon et al., 2023), and data-efficient learning (Driess et al., 2023; Jiaqi Xu et al., 2024). This makes object-centric representations a

[5] This is known as "*shortcut learning*" (Geirhos et al., 2020): the model picks up on any useful statistical regularities in the training data, whether they are "spurious" or hold more generally. A common critique is that the model thus *only* learns a "surface-level understanding", but this formulation perhaps underappreciates the capabilities that can emerge from statistical modeling at scale (Brown et al., 2020).

[6] For example: Eslami et al. (2016), Greff et al. (2016, 2017), Steenkiste et al. (2018), Burgess et al. (2019), Greff et al. (2019), Engelcke, Kosiorek, et al. (2020), Z. Lin et al. (2020), Locatello et al. (2020), Kipf et al. (2022), and Singh, Deng, et al. (2022).

promising basis for supporting the features of human cognition, without losing the advantages of neural networks.

Let us discuss object-centric representation learning through the lens of the binding problem and relate it to this thesis:

- *Representation* — the current standard is to model the data as a *set of latent vectors* called *"slots"*: each slot should capture the "description" of exactly one object. *The work presented in this thesis* similarly uses slots as the core representational mechanism, and builds upon established object-centric models (Jiang et al., 2020; Locatello et al., 2020; Kipf et al., 2022) to extract them.

- *Segregation* — significant work has gone into deriving sophisticated inference procedures and inductive biases that allow neural networks to discover objects from high-dimensional, unstructured visual data (Yuan et al., 2023). However, despite all the efforts, previous methods have been limited to discover objects in simple, synthetic, or otherwise constrained datasets of low complexity. *This thesis* (1) proposes to overcome this restriction by introducing a *semantic inductive bias*; (2) shows that such a bias can be derived fully unsupervised; and (3) experimentally demonstrates that this significantly extends the scope of object-centric models to natural, real-world images and videos.

- *Composition* — object-centric representations have been shown to form a robust basis for downstream tasks, as broad and diverse as visual reasoning (Mondal et al., 2023; Webb et al., 2023), visual question answering (D. Ding et al., 2021; Mamaghan et al., 2024; Jiaqi Xu et al., 2024), image segmentation (Y. Zhou et al., 2021; Jiarui Xu et al., 2022), retrieval (Weinzaepfel et al., 2022; Kim, Kim, and Kwak, 2023) and generation (Singh, Deng, et al., 2022; Jiang et al., 2023; Jabri et al., 2024), physics modeling (Kipf et al., 2020; Kossen et al., 2020; Wu, Dvornik, et al., 2023), reinforcement learning (RL) (Veerapaneni et al., 2019; Yoon et al., 2023), and robotics (Heravi et al., 2022; Driess et al., 2023). *This thesis* further demonstrates the utility of object-centric representations with two applications in autonomous agent learning. Specifically, it introduces algorithms for composing such representations to (1) form complex goals from simpler ones; and (2) infer causal relationships between agents and objects.

Summary of Contributions This thesis advances the field of object-centric representation learning along the two dimensions of segregation and composition.

For segregation, its major contribution is an approach that, *for the first time*, allowed decomposing unconstrained real-world data into object-centric representations. Based on the analysis that previous methods failed on complex data due to a bias towards surface-level image statistics such as color, this approach consists of injecting a *semantic bias* into the model by predicting targets based on pre-trained features. Such features can be obtained by modern self-supervised learning methods in a fully unsupervised way while exhibiting a strong level of internal semantic consistency (Caron et al., 2021). The resulting models, Dinosaur (Seitzer et al., 2023) and VideoSAUR (Zadaianchuk, Seitzer, et al., 2023), yield state-of-the-art results and are demonstrated to scale to real-world image datasets such as PASCAL VOC (Everingham et al., 2010) or COCO (T. Lin et al., 2014), and video datasets such as DAVIS (Pont-Tuset, Perazzi, et al., 2017) or Youtube-VIS (L. Yang et al., 2021).

For composition, the thesis presents two applications of object-centric representations to autonomous agent learning. In the first, an algorithm for self-supervised multi-goal RL from images is proposed; with the help of object-centric representations, the agent learns to compose simple sub-goals to achieve tasks in complex multi-object environments (Zadaianchuk et al., 2021). In the second, by interpreting structured object representations as variables in a causal graph, a measure of the *causal influence* of the agent on objects is derived. This measure can be used to equip RL agents with structural knowledge about the environment, greatly enhancing their sample efficiency (Seitzer et al., 2021).

Individually, these works also constitute independent contributions to the field of reinforcement learning, specifically to the emerging areas of self-supervised RL (Colas et al., 2020) and causal RL (Zeng et al., 2024). Together, they serve as two case studies that exemplify the benefits of object-centric representations for downstream applications: (1) how their grounding and interpretability allow the design of informed algorithms; (2) how their compositionality enables generalization to scenarios not encountered during training; and (3) how the assumption of structure is an excellent inductive bias that can greatly increase data efficiency.

Outline This thesis is organized into four parts, each examining a different aspect around structured object representations. Part I introduces structured object representations: what they are, why they are interesting (Chap. 2) and how neural networks can learn them (Chap. 3). Part II centers around structured object representations for autonomous agent learning, in particular for self-supervised discovery of complex goals (Chap. 4) and enhancing sample efficiency (Chap. 5). Part III introduces methods for the unsupervised discovery of objects in real-world images (Chap. 6) and videos (Chap. 7). Part IV summarizes the research in this thesis and discusses the potential of structured object representations in context with the broader developments in the field of machine learning.

Part I

On Structured Object Representations

Chapter 2

Structured Object Representations: An Introduction

In this first part of the thesis, I introduce its central topic, *structured object representations*. The discussion is separated into two chapters. This chapter gives a broad introduction to the idea of representing data in terms of objects. The next chapter reviews the major framework for implementing structured object representations with neural networks, object-centric representation learning.

This chapter is organized as follows. First, Sec. 2.1 introduces what structured object representations are, followed by Sec. 2.2 motivating their relevance in today's machine learning landscape. Then, Sec. 2.3 details the properties that give object representations their broad benefits. Finally, Sec. 2.4 discusses how objects can be discovered from data.

2.1 Introduction

Let us start by dissecting the somewhat convoluted term "structured object representation" word by word: starting at the end with the question of "representation", circling around to the role of "structure", and finally ending up with the centerpiece, the "object".

2.1.1 The Question of Representation

A fundamental question in machine learning is that of *representation*, that is, how a model "perceives" its input data — images, sounds, or text, for example. A good representation supports the task that the model is trying to solve, for example correctly classifying the object in an image. But usually, we expect more from a representation: we want it to *generalize* to new situations outside the data it was trained on. This requires understanding *how* the new situation is different from before; to support this, the representation needs to capture the *abstract factors of variation* that can explain these differences (Bengio et al., 2013). Thus, the process of finding a good representation can be viewed as the *search for abstractions*.

Today, the most effective method to find good representations is to *learn* them from data — a process called *representation learning* — using deep neural networks.[1] Neural networks learn distributed representations: high-dimensional, continuous vectors that can express a variety of concepts at once.[2] These representations have several advantages. First, their large capacity allows for *redundant encoding* of concepts, leading to a level of robustness to noise and missing information. For example, an image representation might capture factors like "wheels", "windows", "headlights", together forming the concept "car"; this way, even if one of the factors is missing (e.g. due to occlusion), the overall concept encoding remains stable. Second, they can induce a space in which closeness corresponds to *semantic similarity*. For example, the representation for the concept "car" might be similar to the one for the concept "truck" as they share most of the factors. Crucially, this similarity property supports *generalization* through a form of implicit deductive reasoning: upon learning "cars drive on roads", this knowledge will be associated with "trucks" as well.

However, the type of generalization that distributed representations allow is limited. A model might still categorize a car as being able to drive even if it is missing wheels; similarly, a model that has never encountered boats might wrongly conclude "boats drive on roads" because "boat" is represented similarly to "car" — the representation does not reflect that wheels are necessary to drive on roads. So far, this kind of *systematic generalization* is difficult for neural networks, as we will discuss in Sec. 2.2.1. Distributed representations lack inherent *structure*, which makes them susceptible to shortcut reasoning when training data is missing (Geirhos et al., 2020).

2.1.2 The Role of Structure in Machine Learning

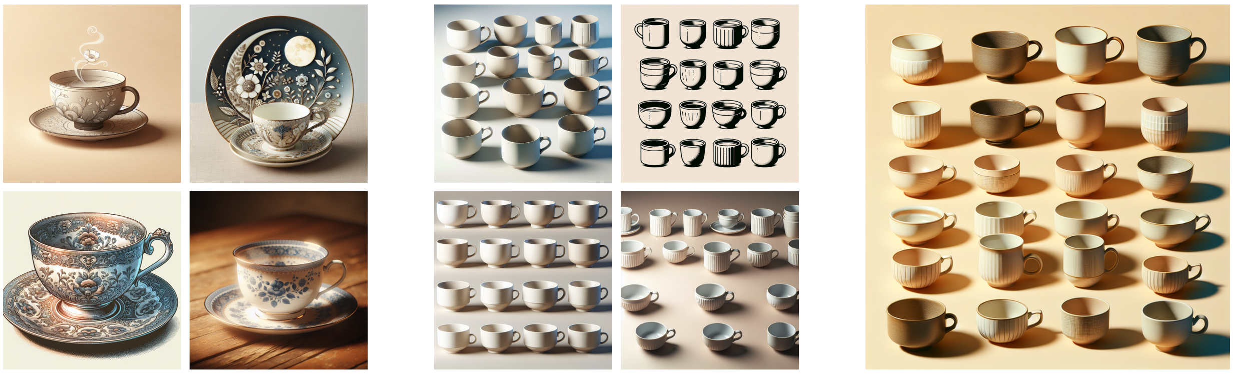
Most models have a form of *structure* that reflects some real-world properties of the data.[3] We can view the structure as a scaffolding that guides the model's learning and information processing. Just like we can hold onto a scaffold when other support is missing, the model can fall back on the structure when data is missing, unclear, or different from before. This is because a good structure has an actual *meaning*, it is grounded in reality — it is trustworthy. On the other hand, structure is also rigid; it can constrain the model into certain boundaries even if the data speaks otherwise. Thus, a structure that is too naive or only superficially represents the world harms the model's ability to correctly fit the data.

Where does the structure come from? The answer is that it is us, as model designers, who put in the structure — in machine learning often called the *inductive bias* (Mitchell, 1980). Until recently, much of machine learning research has been concerned with identifying appropriate inductive biases for different types of data (Goodfellow et al., 2016). With ever-growing compute capabilities and amounts of data, an underlying

[1] While earlier approaches hand-designed representations (called feature engineering), the advent of deep learning shifted the focus towards automatically learned, hierarchical representations. In fact, much of the success of deep learning can be attributed to its ability to learn powerful abstract representations of high-dimensional data (Goodfellow et al., 2016).

[2] In a distributed representation, each element can be independently varied, in contrast to symbolic one-hot representations; this way, one can represent an exponential number of configurations with a linear number of parameters (Bengio et al., 2013).

[3] A classic example of a structured model is a probabilistic graphical model, which represents probability distributions as graphs over random variables (Koller and Friedman, 2009).



"An image of a cup with a saucer on top."

"An image showing 12 plain tea cups from the same set, all identical in style and design."

"How many cups are in this image?" Answer: *"20 cups."*

Figure 2.1: Failures of foundation models. Left: The image generation model DALL·E 3 (Betker et al., 2023) fails to follow the geometric relation "on top of"; this is likely because of a bias towards the training data, where images with saucers covering cups are underrepresented. **Center:** DALL·E 3 exhibits problems with counting and the concept of "sameness". **Right:** The GPT-4 model (OpenAI, 2024) also fails to count correctly. While these particular errors may be solved by future models or different prompting techniques, they still illustrate certain failure modes around compositionality and reasoning; similar observations have been made elsewhere (Arkoudas, 2023; Kobayashi et al., 2024; Majumdar et al., 2024; Tong et al., 2024; Z. Xu et al., 2024).

trend has been to weaken these biases more and more: finding broad, flexible assumptions that apply across all kinds of data without overly limiting the model's capacity.[4] This culminated in what Sutton (2019) poignantly called the "*bitter lesson*": the observation that scaling model and data ultimately trumps any structural assumptions. Indeed, the success of large-scale foundation models supports that view (Brown et al., 2020; Bubeck et al., 2023). This might indicate that all research on structure is futile. But foundation models still fall short of human cognition (see Fig. 2.1); doubt has been expressed that this hurdle can be overcome by scaling alone (Greff et al., 2020; Schölkopf et al., 2021; LeCun, 2022).

In this thesis, I take the position that structure is an essential part of general AI, but that models must largely learn to discover that structure by themselves. The research challenge lies in defining structures that are as general as possible, together with the mechanisms that enable models to capture them from data.

2.1.3 Objects as Composable Modular Units

One such general structure is *modularity*. At its core, the world is composed of independently existing elements that can combine and reconfigure in various ways. Thus, it is sensible to organize the model's information processing following the principle of modularity. Central to this concept is what I refer to as a *structured object representation*. In such a representation, the main structural units are "objects": recurring patterns that are "*self-contained and separate from one another such that they can be related and assembled into structures without losing their integrity*" (Greff et al., 2020).

[4] A prime example is the Transformer model (Vaswani et al., 2017), now universally applied across all data modalities, making less assumptions about the data compared to e.g. convolutional neural networks.

This is, intentionally, quite a broad and abstract definition of objects. While actual, physical objects fall under its umbrella, it also includes "*non-visual entities such as spoken words, imagined or remembered entities, and even more abstract entities such as categories, concepts, behaviors, and goals*" (Greff et al., 2020). Nevertheless, the most prominent and accessible instantiation of structured object representations is indeed about capturing objects from visual data. It is also the one we will be concerned with in the following — the term "object" can thus be understood in its literal sense for the purposes of this thesis.

2.2 The Need for Structured Object Representations

2.2.1 The Benefits of Object Representations

Organizing models around structured object representations promises intriguing advantages in several areas — in particular areas where conventional models without such representations are known to struggle. We now briefly discuss these areas, motivate how object structure could help, and relate them to issues in current neural models.

Note that (1) empirical results regarding these benefits are somewhat sparse, not least because research around object representations is still burgeoning. Nevertheless, we list some encouraging supporting evidence in the side margin; and (2) it is difficult to draw conclusions on the weaknesses of deep learning models on the basis of any particular result or benchmark, as scaling rapidly advances today's state-of-the-art. However, I believe that the mentioned issues are more fundamental and require solutions outside the scaling paradigm.

- *Data efficiency.*[5] With a modular structure, knowledge acquired once can be re-used in different contexts and re-purposed for new skills. For example, a robot that learned to grasp a cup may then also be able to grasp a plate with no or minimal adaptations. In contrast, current models are data hungry, in particular compared to humans (Tsividis et al., 2017; Udandarao et al., 2024).
- *Systematic generalization.*[6] The compositional nature of object representations allows recognizing and processing concepts in configurations unseen or unrepresented in the training data. For example, the robot should be able to handle a situation with a plate *on* a cup, even if the training data only contained examples with plates *under or beside* cups; similarly, it should keep working in situations with 20 cups, even if it has never encountered that many cups at once before. In comparison, current models struggle with such relational and compositional reasoning (e.g. Bahdanau et al., 2019; Hupkes et al., 2020; Yuksekgonul et al., 2023; Kobayashi et al., 2024; Z. Xu et al., 2024, see also Fig. 2.1 for examples).
- *Robustness to distribution shifts.*[7] Learning a representation grounded in real-world structures rather than surface-level statistics should

[5] Heravi et al. (2022), Driess et al. (2023), and Jiaqi Xu et al. (2024).

[6] Santoro et al. (2018), Dittadi et al. (2022), Stanić, Tang, et al. (2023), and Wiedemer et al. (2024).

[7] Romijnders et al. (2021), Dittadi et al. (2022), and Yoon et al. (2023).

lead to models that are better equipped to handle nuisances or shifts in the input. In contrast, current (vision) models are known to learn shortcuts that do not generalize out-of-distribution (Geirhos et al., 2020; Hendrycks et al., 2021), to be biased towards weak local regularities (Brendel and Bethge, 2019), and to be sensitive to common image corruptions (Hendrycks and Dietterich, 2019; S. Wang et al., 2023), texture changes (Geirhos et al., 2019), or backgrounds (X. Li et al., 2023).

- *Interpretability*. Symbol-like representations simplify introspecting the model's decision process. This improves general model understanding and aids verification for safety-critical applications; it also allows the *design of informed algorithms* that rely on the properties of such representations.[8] On the other hand, conventional neural representations are often inscrutable due to their distributed nature and may require sophisticated techniques to understand (Olah et al., 2020; Elhage et al., 2021).
- *Objects as a modeling language for the physical world*.[9] For many applications dealing with the human world, objects are the natural level of abstraction; this includes embodied agents such as household robots, or self-driving cars, but also more generic tasks such as image generation, video understanding, or predictive world modeling. Besides the fact that many tasks relevant to humans are inherently about objects, objects simplify the modeling of dynamics and interactions, and they can be grounded in natural language to offer an interface between agent and user. While all such tasks have also been approached with conventional neural models, object representations could provide a particularly useful inductive bias, leading to improved performance as well as better data and computational efficiency.[10]

2.2.2 Objects as Pragmatic Causal Representations

These advantages, in particular generalization and robustness, are consistent with those motivating a *causal approach to machine learning* (Schölkopf, 2022). This is no coincidence. The principle of *independent causal mechanisms* states that the world *"is composed of autonomous modules that do not inform or influence each other"* (Peters et al., 2017). This is close to our definition of objects as independent, modular, and composable structures (cf. Sec. 2.1.3). Causal representation learning (Schölkopf et al., 2021) aims to recover the causal mechanisms that generated the data — in other words, learning structure from data. The high-level goals behind learning structured object representations and learning causal representations are thus similar.[11]

The major difference is that a causal representation supports an explicit notion of "*intervention*", that is, an external action changing one or several of the causal mechanisms (Pearl, 2009). As a consequence, causal learning focuses on identifiability[12] to maintain correctness guarantees about the model's behavior under interventions. In contrast, structured object

[8] I will demonstrate several examples of such informed algorithms in Part II of this thesis.

[9] Heravi et al. (2022), Driess et al. (2023), Kim, Kim, Lan, et al. (2023), Mamaghan et al. (2024), Hamdan and Guney (2024), Gu et al. (2024), Jiaqi Xu et al. (2024), Z. Wu et al. (2024)

[10] I speculate that for some applications such as long-term video modeling or building open-world scene graphs, object representations (and similar abstractions) may even be *necessary* to avoid intractable computational costs.

[11] Schölkopf et al. (2021) also draw the connection between objects and causality: "*Objects are constituents of scenes that in principle permit separate interventions. A disentangled representation of a scene containing objects should probably use objects as some of the building blocks of an overall causal factorization*".

[12] The conditions under which the correct causal model can be recovered from observed data.

representations have no such guarantees; in exchange, this yields more flexibility for developing models that learn representations with the desired properties (e.g. independence, composability). This way, some of the advantages of a causal representation could be retained in practice: for instance, an object representation that is sufficiently modular could allow interventions by replacing one of the objects with another; using a representation modified in this manner is akin to a *counterfactual simulation*.

I see object representations as "pragmatic causal representations" for visual data, foregoing causal guarantees in return for a more tractable and scalable approach. In Chap. 5, we will further discuss how structured object representations can be embedded in a causal framework to derive causal relationships between objects.

2.2.3 Learning Structured Object Representations

So far, we have not specified *how* a structured object representation can be obtained, nor its particular *format*. In accordance with the discussion in Sec. 2.1.1, we would like to leverage the advantages of deep learning-based neural representations as much as possible. Consequently, this thesis focuses on *learning structured object representations with neural networks*. Methods aimed at achieving that goal are commonly grouped under the term *object-centric representation learning*.[13] In most approaches, the representation is structured as a *set of vectors*, where each vector is the distributed representation of an object. We will discuss these methods in detail in Chap. 3.

Current neural models do not learn structured object representations naturally — the whole raison d'être behind object-centric representation learning. In particular, they are lacking the mechanisms to (1) discover modular object structures in unstructured data; (2) dynamically extract representations for such structures; and (3) utilize the representations in a compositional manner. These are the aspects of segregation, representation, and composition, together constituting the binding problem of neural networks (Greff et al., 2020, cf. also Fig. 1.1). The remainder of this chapter discusses the aspects of representation and segregation.

2.3 Desiderata for Object Representations

In the last section, we discussed the extensive benefits of structured object representations. We now define several properties that enable structured object representations to be such effective parts of larger systems. It is important to note that these properties may not be fully realizable in practice; nevertheless, they can form the basis for robust, generalizing, interpretable, and sample efficient models.

- *Separation*: individual objects representations are context-independent without interference between them. For example, changing an object in the input should lead to a change in a single object representation. Each

[13] There is no universally accepted name for the topic of learning object representations with neural networks. Depending on the focus, it has been referred to as symbol-like representation learning (Greff et al., 2017), multi-object representation learning (Greff et al., 2019), (unsupervised) scene decomposition (Burgess et al., 2019), compositional scene representation learning (Yuan et al., 2023), object discovery (Bao et al., 2023), or simply object-centric learning (Locatello et al., 2020). I adopt the term object-centric representation learning, possibly first used in this context by Engelcke, Kosiorek, et al. (2020), as it emphasizes the aspect of "representation".

representation is a modular building block, facilitating generalization when composing them in novel combinations.

- *Completeness*: object representations capture all required information about the objects. What is required depends on the task, and ranges from just encoding the existence of the object (e.g. for a counting task) to being able to faithfully reconstruct the full object (e.g. for image editing). In the absence of prior task knowledge, a guiding principle is to preserve as much information as practicable, starting with the most significant factors of variation (Bengio et al., 2013).[14]

- *Common format*: object representations share a representational format, resulting in a common interface that increases generalization and sample efficiency. For example, if an object's position is always represented in the same fashion, an "A is close to B" operator learned on a few data points directly generalizes to all possible combinations of objects.

- *Disentanglement*: object representations internally isolate different explanatory factors in their latent dimensions. For example, an object's position should be represented separately from its size, shape, or color. Besides better interpretability, disentangled representations may support compositional generalization to previously unseen combinations of factors[15] (Higgins, Matthey, et al., 2017).

When modeling temporal data, two more properties can be desirable:

- *Consistency*: object representations are assigned the same object instance across time and maintain a stable representation of that object. This is helpful for tracking or re-identification of objects and is required for aggregating information about objects over time.

- *Predictability*: it is possible to forecast the object's future evolution from its representation (to the extent possible without external interference). This requires the representation to fulfill a *Markov property*, that is, to "summarize" the past to the degree necessary to predict the future.[16] For example, to predict an object's future position, its representation must contain its current velocity and acceleration.

2.4 Discovering Objects from Data

To extract representations of objects, a learning system must first be able to recognize them. So far, we have omitted the question of how exactly this takes place; within Greff et al.'s (2020) framework of the binding problem, this is the aspect of *segregation*. Is it even feasible to discover objects solely from data? In this section, I will first discuss the nature of objects from an information-theoretic perspective, and relate them to human perception (Sec. 2.4.1). Then I introduce several principles that together result in a "toolbox of inductive biases" (Sec. 2.4.2) — indeed enabling models to discover objects.

[14] Put another way, the goal is to make the representation as *invertible* as possible.

[15] For example, correctly representing a polar bear as "white+bear" having only seen examples of black bears and white sheep.

[16] This can be seen as a temporal version of the *completeness* property

2.4.1 The Nature of Objects

What actually are objects? Avoiding metaphysical discussions, we will focus on aspects tangible for machine learning. First, we can posit two fundamental properties of objects: (1) they are largely *independent of their context*; and (2) they exhibit strong *internal predictive structure*. Intuitively, the former means that an object is difficult to predict from its surrounding context (and vice versa the context from the object); the latter means that different "*views*"[17] of the object are highly predictive of other views. Stated succinctly, objects have low external, and high internal dependencies; both of these aspects are captured by the concept of mutual information (Cover and Thomas, 2006), as we will see in the next section.

It is worth pondering about how humans perceive objects — let us discuss the example in Fig. 2.2, left. We can almost perfectly fill in the partially masked balloon in our mind's eye; the same with the occluded parts of the background. This is because there is a *shared underlying structure* that "explains" both the visible and the occluded parts. Our experience allows us to mentally infer this explanation and complete the missing parts from it. In contrast, we do not expect the presence of the fully masked balloon just from the surrounding sky — without any further hints, this balloon's existence is unlikely.

These observations are consistent with theories of how human perception is organized. For instance, the *likelihood principle* states that our interpretation of a (visual) stimulus tends to be its *most likely* one (Helmholtz, 1962). In the example, this matches the analysis that perception is guided by experience and expectations. A competing theory is the *simplicity principle*[18]: based on Gestalt psychology (Wertheimer, 2012), it states that human perception tends to favor the *simplest* interpretation of a stimulus, or equivalently, the one of minimum complexity (Pomerantz and Kubovy, 1986). In the example, the simplicity principle suggests that we can perceive the occluded balloon as a whole because we prefer the underlying, simple explanation of a complete, intact balloon. While there have been major debates about which theory is correct, they may be "*two sides of the same coin*" — indeed, it is possible to reconcile them from the perspective of information theory (Chater, 1996).

The simplicity principle suggests a further property of objects besides context-independence and internal predictability: that objects (should) have minimal information content, i.e. they can be efficiently encoded. This idea is widely used as an inductive bias to learn object representations with neural networks (Engelcke, Jones, et al., 2020).

Object Hierarchies Before moving on, we make the observation that context-independence and internal predictability actually lie on a continuum; they form a trade-off that suggests a *hierarchy of objects*. Consider the example in Fig. 2.2 (right): from the wheel, we can predict the existence of the car; from the tow truck, we can (weakly) predict the car.[19] While context-independence increases with the size of the group,

[17] Here, "view" refers to an observation of the object under a particular condition, e.g. an occlusion, viewpoint, point in time, or lighting condition.

[18] Also known as the minimum, or the prägnanz principle.

[19] Similarly, in the balloon example, the existence of the fully-masked balloon was actually made slightly more likely by the presence of the other balloons.

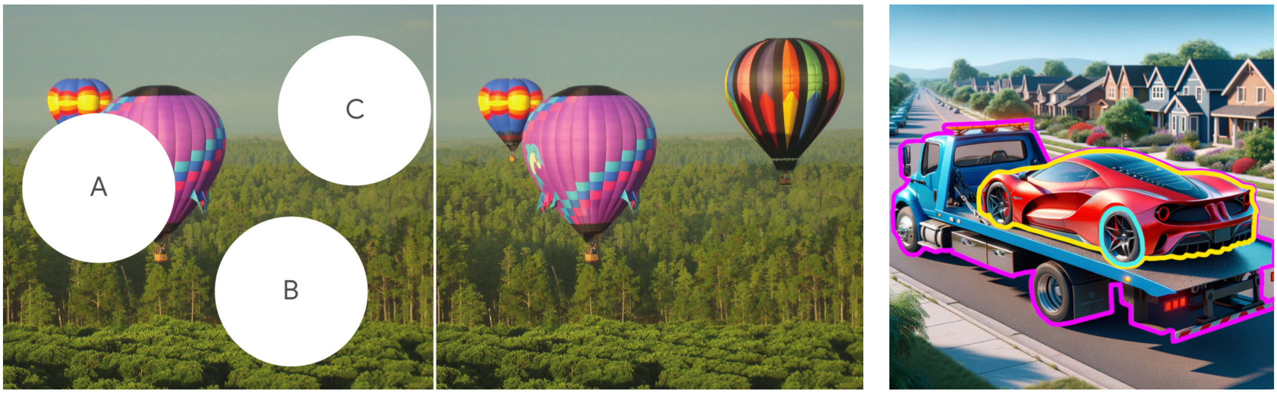


Figure 2.2: The information structure of objects. Left: Objects are *internally predictive* and *context-independent*. Occluded parts of objects (A) or the background (B) can be predicted from visible parts, whereas fully-occluded objects cannot be predicted from the surrounding context (C). Example from Greff et al. (2020) under CC-BY 4.0 license. **Right:** The trade-off between predictiveness and independence induces a *hierarchy of objects* related by their dependency structure. Groupings into parts (the wheel), wholes (the car), or composites (tow truck with car) are all valid decompositions, characterized by how strongly related and context-independent they are. The colored outlines show the "borders of predictability". Image generated with DALL·E 3.

internal predictability decreases, because we group parts together that are only weakly related. Greff et al. (2020) suggest "*that this trade-off induces a Pareto front of valid decompositions*". Essentially, objects can be grouped by segregating them following the "borders of predictability" (colored outlines in Fig. 2.2 (right)). In the example, all three groupings are valid — which grouping to select depends on its intended purpose.

2.4.2 Principles to Discover Objects in Data

We will now review several organizing principles that reveal objects in data — each principle can be considered a characteristic "footprint" that objects leave behind in the data. These principles can then be transformed into optimization criteria or biases for object discovery; most of these criteria have been employed in practice. We will keep the discussion on a high-level here, and explore how some of these principles can be implemented in neural networks in the next chapter.

We start with three generic principles for structuring information into objects derived from the previous discussion about the nature of objects. These principles are all applicable across modalities and could also serve to organize information in more abstract spaces:

- *Compression*: objects have a simple description, i.e. they can be encoded with less information compared to competing hypotheses. This suggests finding a separation of the input into objects that *compresses well* under a low-dimensional representation bottleneck.[20] Compression is probably the principle most used in practice. Two forms of bottlenecks can be distinguished: *explicit*, corresponding to the dimensionality of the object representation, and *implicit*, corresponding to the capacity of the used model. Both play a significant role for object discovery (Engelcke, Jones, et al., 2020; Papa et al., 2022).

[20] A strong form of compression is the principle of *typicality*, i.e. to assume that objects come from prototypical sets of categories, appearances or shapes. This leads to the idea of capsules (Hinton et al., 2011), and has been used for segmentation (Zadaianchuk, Kleindessner, et al., 2023) and object discovery (Wen et al., 2022; Kori et al., 2024).

- *Independence*: objects are independent of their surroundings; this suggests enforcing statistical independence between the representations. This can be turned into an optimization criterion by minimizing a form of mutual information between object representations (Y. Yang et al., 2020; Zoran et al., 2021). Another instantiation of this principle is to select representations with *mutually exclusive dependencies* in the functions that map from and to the data space.[21] While seldom explicitly enforced, most models contain an architectural bias towards this form of independence.
- *Predictiveness*: different views of an object are dependent on each other; this suggests maximizing the mutual information between the representations of different object views. This principle has seen little explicit application so far.[22]

Specifically for *temporal data*, the following two principles apply:

- *Temporal consistency*: most properties of an object are stable over time (e.g. shape or appearance), or change slowly (e.g. position). This suggests selecting objects such that it is possible to sparsely update their representation over time,[23] which can be implemented by encouraging representations of consecutive time steps to be close (Bao et al., 2022; Traub et al., 2023).
- *Temporal sparseness of interactions*: objects tend to *interact sparsely* in time, which suggests selecting objects such that their temporal evolution can be modeled with as little information from other objects as possible.[24] This principle is so far little used for discovering objects, but could be implemented using suitable architectural biases (Goyal et al., 2021).

The next two principles make use of *human knowledge* about objects:

- *Describable*: humans label meaningful objects with specific names (or *symbols*); thus, if a particular pattern is repeatedly associated with the same label, it should be captured as an object. As a criterion, this can take the form of a grounding loss that aligns textual descriptions with object representations (Jiarui Xu et al., 2022; Kim, Kim, Lan, et al., 2023).
- *Supervision*: an explicit way to discover objects is to simply *annotate* what constitutes an object in the data; such annotations typically take the forms of segmentation masks that mark the pixels belonging to the objects. Object representations can then be optimized or biased towards capturing regions aligning with the annotations (Kipf et al., 2022; Kim, Choi, et al., 2023).

The remaining principles use that objects *physically exist* in space & time:

- *Spatial locality*: in 3D space, objects occupy local connected regions; for 2D projections of objects, this generally holds as well (except for occlusions). Locality can be turned into a criterion by spatially

[21] Each element in the data space affects and is affected by a single object representation, which can be seen as constraints on the encoding and decoding functions respectively (Brady et al., 2023; Wiedemer et al., 2024).

[22] Exceptions include limited approaches based on contrastive learning (Kipf et al., 2020; Baldassarre and Azizpour, 2022). However, this principle is widely used for self-supervised representation learning (Aaron van den Oord et al., 2019; Shwartz Ziv and LeCun, 2024), and was applied for segmentation (Isola et al., 2014; Ji et al., 2019; Wen et al., 2022).

[23] More or less, a temporal variant of the *predictiveness principle*.

[24] More or less, a temporal variant of the *independence principle*.

restricting or biasing object selection to local and/or connected regions of space (Chakravarthy et al., 2023; Foo et al., 2023).

- *Temporal locality*: objects have a limited range of motion, i.e. they cannot move arbitrarily far per time step, which can be seen as an extension of spatial locality to 4D space-time. This can be implemented by requiring that the spatial regions an object occupies in two consecutive time steps are close (Jiang et al., 2020).
- *Coherent motion*: elements of an object generally move consistently together (the principle of "common fate"[25]), which can be seen as a manifestation of the predictiveness principle for temporal data. This can be turned into a criterion by grouping elements (e.g. pixels) with similar motion into the same object (Kipf et al., 2022; Tangemann et al., 2023).

The final two principles concern the behavior of objects under *actions*. These may be best suited for domains with an agent model, e.g. reinforcement learning; however, I am not aware of any successful applications of these principles for object discovery.

- *Interventions*: objects can be *independently intervened on*, that is, an agent can change the state of an object without influencing other objects.[26] An instantiation of this principle would select for object representations that change sparsely under an agent's actions; the difficulty lies in determining whether the agent has had impact on its environment.[27]
- *Affordances*: objects can be characterized in terms of the actions that can be performed on them. More generally, objects take part in temporally abstract "*events*", which can be seen as an agent causing a state-change in an object (Gärdenfors, 2014). It is not obvious how to instantiate this principle; one idea is to select objects such that their state-action trajectories can be grouped into distinct clusters representing the affordances.

There are certainly more principles for guiding object discovery that could be added here. Particularly the field of computer vision has developed a rich set of ideas that could be utilized to discover objects. For instance, one could integrate prior knowledge on object shapes, e.g. convexity (Royer et al., 2016) or learned shape models (Elich et al., 2022), or utilize the fact that objects are often symmetric with repeating parts (Bagon et al., 2008).

However, I argue that structural assumptions should be kept as general as possible, i.e. to use a few generic principles to organize information, for three reasons. First, many assumptions are redundant and can be subsumed by more generic principles. For example, assumptions like symmetry or convexity can be captured by the predictability or compression principles. Second, hand-designed assumptions are likely to be too brittle and thus harm generalization. Third, assumptions tend to reduce model scalability, i.e. the model's ability to keep absorbing the training data with increased sizes of model and data. For these reasons, it is necessary to employ generic principles that guide the model to (1) discover *generalizing* structures; and (2) learn *better* structures when scaled.

[25] One of the Gestalt laws of human perception, the common fate principle states that elements that move together tend to be perceived as a group (Wertheimer, 2012).

[26] More or less, a causal variant of the *independence principle*; cf. also the principle of independent causal mechanisms (Sec. 2.2.2).

[27] In Chap. 5, we will discuss how the agent's causal influence on objects in the environment can be measured.

Chapter 3

Building Object-Centric Neural Networks

This chapter provides a comprehensive overview of all aspects involved in constructing neural networks that learn object-centric representations: model design, training, and evaluation. In Sec. 3.1, we will begin by sorting the object-centric landscape along several different categories. Then, in Sec. 3.2, we review the training process of object-centric neural networks. To exemplify the discussed ideas, Sec. 3.3 presents three case studies of object-centric models — Slot Attention (Locatello et al., 2020), SAVi (Kipf et al., 2022), and Scalor (Jiang et al., 2020) — which also form the basis for the work presented in Parts II and III of this thesis. Finally, in Sec. 3.4, we discuss the important topic of evaluating object-centric representations.

For the remainder of this thesis, we will limit our discussion to models that learn *slot-based* object-centric representations. Slot-based representations structure the input into a *set of vectors*. There is no inherent order to the object vectors; to emphasize their interchangeability, the vectors are called *"slots"*. Slot-based representations are learned by *neural models* trained end-to-end with *gradient-based optimization*. We would like to highlight that alternative neural object representations exist (see Greff et al., 2020, Section 3.3); in particular, implicit object representations based on complex-valued networks have recently gained some traction (Löwe et al., 2022, 2023; Stanić, Gopalakrishnan, et al., 2023).

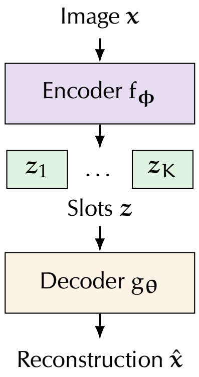


Figure 3.1: Slot model based on autoencoding.

Preliminaries Slot-based models represent an image $\mathbf{x} \in \mathbb{R}^{H \times W \times 3}$ with a set of K slot vectors $\mathbf{z} = \{\mathbf{z}_k \in \mathbb{R}^{D_{\text{slots}}}\}_{k=1}^{K}$. Typically, they consist of an encoder $f_\phi : \mathbb{R}^{H \times W \times 3} \mapsto \mathbb{R}^{K \times D_{\text{slots}}}$ that maps from image to slot space and a decoder $g_\theta : \mathbb{R}^{K \times D_{\text{slots}}} \mapsto \mathbb{R}^{H \times W \times 3}$ that maps from slot to image space, where ϕ and θ are learnable parameter vectors. Associated with the slots is a set of (soft) *segmentation masks*[1] $\mathbf{m} = \{\mathbf{m}_k \in [0,1]^{H \times W}\}_{k=1}^{K}$ that signifies the location and shape of each slot in the image; these are additionally produced by the decoder and/or encoder. Training, in most cases, amounts to optimizing a reconstruction criterion, e.g. the

[1] Such soft masks can be interpreted as probability distributions over which slots occupy which pixels; most object-centric methods are able to provide such masks.

Table 3.1: Comparison of slot-based models. We compare selected slot-based models along four major characteristics: employing scene-based slots or spatial slots (Sec. 3.1.1), inferring slots in parallel or sequentially (Sec. 3.1.2), extracting slots from pixels or from features (Sec. 3.1.3), and image or video inputs (Sec. 3.1.4). We refer to the main text for details.

	Model	Reference	Parallel –or– Sequential	Scene –or– Spatial	Pixels –or– Features	Predecessor
Image-based	AIR	Eslami et al. (2016)	Sequential	Spatial	Hybrid	
	N-EM	Greff et al. (2017)	Parallel	Scene	Pixels	
	MONet	Burgess et al. (2019)	Sequential	Scene	Pixels	AIR
	IODINE	Greff et al. (2019)	Parallel	Scene	Pixels	N-EM
	SPAIR	Crawford and Pineau (2020b)	Sequential	Spatial	Hybrid	AIR
	SPACE	Z. Lin et al. (2020)	Hybrid	Hybrid	Hybrid	SPAIR
	GENESIS	Engelcke, Kosiorek, et al. (2020)	Hybrid	Scene	Pixels	MONet
	Slot Attention	Locatello et al. (2020)	Parallel	Scene	Feats.	IODINE
	GENESISv2	Engelcke et al. (2021)	Sequential	Scene	Feats.	GENESIS
	SLATE	Singh, Deng, et al. (2022)	Parallel	Scene	Feats.	Slot Attention
	Dinosaur	Seitzer et al. (2023)	Parallel	Scene	Feats.	Slot Attention
Video-based	SQAIR	Kosiorek et al. (2018)	Sequential	Spatial	Hybrid	AIR
	OP3	Veerapaneni et al. (2019)	Parallel	Scene	Pixels	IODINE
	Scalor	Jiang et al. (2020)	Parallel	Spatial	Hybrid	SQAIR
	ViMON	Weis et al. (2021)	Sequential	Scene	Pixels	MONet
	SAVi	Kipf et al. (2022)	Parallel	Scene	Feats.	Slot Attention
	STEVE	Singh, Wu, et al. (2022)	Parallel	Scene	Feats.	SLATE
	Loci	Traub et al. (2023)	Parallel	Spatial	Pixels	
	VideoSAUR	Zadaianchuk, Seitzer, et al. (2023)	Parallel	Scene	Feats.	Dinosaur

mean squared error loss (see Sec. 3.2). We can consider this to be a typical autoencoder setup, with the slots forming a lower-dimensional bottleneck representation of the data (cf. Fig. 3.1). However, the structure of encoder, slot representations, and decoder — as well as the inductive biases that enable object discovery — can differ considerably between approaches, as we will see in the following. A commonality is the heavy use of *weight sharing*; to ensure that the slots share a *common representational format*, many of the modules that produce and process the slots are shared across them.

3.1 Characteristics of Slot-Based Neural Networks

There is, by now, a myriad of approaches[2] to learn slot representations. A comprehensive discussion of all these methods is beyond the scope of this thesis — for an extensive review, we refer the reader to Yuan et al. (2023). Instead, this section will focus on a few major characteristics that differentiate the existing methods: *scene-based* or *spatial* slot representations, *parallel* or *sequential* slot inference, *pixel* or *feature space* extraction of slots, and *image* or *video* inputs. In Table 3.1, we present a selection of methods classified according to our scheme.

[2] Yuan et al. (2023) enumerate 48 different methods; the author of this thesis is aware of a comparable number of works more.

3.1.1 Scene-based or Spatial Slots

On the representational level, we can differentiate between *scene-based* or *spatial slots*. Scene-based slots model arbitrary segments of an image, with location and scale *implicitly* encoded within the slot representation. The underlying assumption is that of a *scene-mixture model*, i.e. that the image can be represented by a mixture of a finite number of component images; each slot encodes one such component image. As inference for scene-based slots is implemented by predicting a form of segmentation mask, complex object morphologies can be modeled. A disadvantage of this flexibility is that the model carries no inherent locality bias, which e.g. means that spatially disjoint image segments can be grouped into the same slot.[3] Furthermore, the model does not distinguish between foreground and background; the background will also be decomposed into one or more slots.

In contrast, *spatial slots* are tied to an image location, with location and scale *explicitly* represented, usually as the bounding box[4] containing the modeled object (Eslami et al., 2016; Crawford and Pineau, 2020b; Jiang et al., 2020). In this case, each slot z_k is further structured as the triplet $z_k = (z_k^{\text{where}}, z_k^{\text{what}}, z_k^{\text{pres}})$,[5] where z_k^{where} contains the location information as the bounding box's center position z_k^{pos} and size z_k^{scale}, z_k^{what} models appearance and pose of the object, and the binary z_k^{pres} indicates the presence of the object. In some models, z_k^{where} contains a further component z_k^{depth} modeling the depth position of the object to deal with occlusions. Furthermore, in most approaches, there is also an explicit background representation z^{bg}.

To facilitate the additional structure in the representation, spatial slots are typically paired with suitable encoder and decoder designs. For example, after inferring z_k^{where}, a corresponding "glimpse" of the input image is extracted using a *spatial transformer network*,[6] and processed by an appearance encoder to yield z_k^{what}. For decoding, each object is separately decoded into "canonical RGB space" from z_k^{what}, and stitched together into a composite image, again applying a spatial transformer with z_k^{where}. This design has the advantage that it encodes certain invariances; modeling an object's appearance is decoupled from modeling its position and scale.

Spatial slots encode useful biases for object discovery such as the locality of objects (cf. Sec. 2.4.2). For visually simplistic scenes, this enables them to *scale to dozens of objects* (Jiang et al., 2020). Furthermore, the *additional structure* spatial slots provide, in terms of disentangling position and appearance, can be advantageous for certain downstream applications; we will discuss one such application in Chap. 4. On the other hand, methods using spatial slots usually make strong assumptions about the size of objects to facilitate object discovery, which limits scaling to more heterogeneous scenes. Similarly, the assumptions of locality and of a single object per bounding box may be violated in more complex scenes with frequent (partial) occlusions. Finally, for natural scenes, modeling the background as a single latent component may not

[3] In most cases, objects form a connected region in 2D space (cf. the Gestalt principle; Koffka, 2013), although there are exceptions, e.g. occlusions.

[4] Other forms such as spheres are also possible (Biza et al., 2023; Traub et al., 2023).

[5] We adopt commonly used notation (Eslami et al., 2016; Kosiorek et al., 2018; Jiang et al., 2020).

[6] Spatial transformers (Jaderberg et al., 2015) offer a fully differentiable way to apply geometric operations such as translation, scaling, or rotation to feature maps. Not to be confused with Transformers (Vaswani et al., 2017).

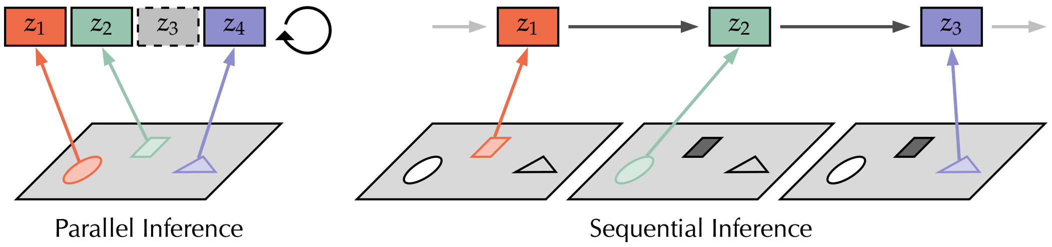


Figure 3.2: Different types of slot inference. Parallel inference binds objects to slots in an identical and independent process, often over several iterations. Sequential inference binds slots one at a time, creating dependencies between slots but allowing for dynamic capacity. Figure adapted from Greff et al. (2020, Fig. 6).

scale effectively. For all these reasons, spatial slots have fallen out of favor compared to the more general scene-based slots.[7] However, these limitations may stem from specific architectural choices; recent work has successfully integrated principles of spatial slots (e.g. invariances, locality) into scene-based slots (Biza et al., 2023; Traub et al., 2023, 2024).

3.1.2 Parallel or Sequential Slot Inference

The task of the encoder is to extract object slots from the image, for which two main paradigms have emerged: *parallel* or *sequential* inference. See Fig. 3.2 for an illustration.

In parallel inference, each slot is derived by an identical and mostly independent process, which thus can be executed in parallel over slots. As a consequence, slots have no inherent ordering and preferences for the modeled content[8]; this makes them very general (Greff et al., 2020). However, this generality introduces a *routing problem*: because slots are indistinguishable, separating information from the input image requires a form of *symmetry breaking*. In parallel inference, this is often achieved by updating the slot representation over several iterations[9] — depending on the specific instantiation, the resulting process structurally resembles the EM algorithm (Greff et al., 2017), amortized variational inference (Greff et al., 2019), iterative clustering (Locatello et al., 2020), or a fixed point procedure (Chang et al., 2022).

In sequential inference, the slots are derived step-by-step, each depending on the previously extracted slots. This has the advantage that the amount of computation is not fixed, and that a dynamic number of slots can be modelled in principle. However, the sequential process imposes an ordering on the slots, which may make *slot separation* more difficult (Greff et al., 2020). Furthermore, sequential inference is inherently less *computationally efficient* than parallel inference, as the latter can run slot inference concurrently. There also exist hybrid models that parallelize parts of the inference to increase efficiency (Engelcke, Kosiorek, et al., 2020; Z. Lin et al., 2020).

[7] Further contributing factors may be the complexity of implementing spatial slot methods, as well as the number of hyperparameters that need to be set; scene-based slot methods are often considerably simpler.

[8] However, spatial slots with parallel inference are still biased to certain spatial properties.

[9] Additional symmetry breaking is introduced by stochasticity, e.g. random slot initialization (Locatello et al., 2020) or sampling latent distributions (Greff et al., 2019).

3.1.3 Pixel or Feature Space Extraction

A further distinction we can make is between *pixel space* or *feature space* extraction of slots, i.e. the level at which *the separation into objects is modelled*. Most earlier approaches with scene-based slots process the full image once or several times for each slot, masking already modelled parts of the image (Greff et al., 2017; Burgess et al., 2019), or conditioning on some side information (Greff et al., 2019). From a perspective of computational efficiency, this approach appears wasteful — most of the processing is redundant. Consequently, later approaches instead inferred slots from features shared between slots (Locatello et al., 2020; Engelcke et al., 2021); this way, most of the computation can be shared between slots.

Similarly, methods utilizing spatial slots extract "glimpses" at the pixel level to model object appearance. The computational burden is less severe than for scene-based slot methods since each glimpse is already lower dimensional. Because spatial slot methods model object positions on the level of features, we categorize them as "*hybrid*" methods in this context (cf. Table 3.1).

3.1.4 Extensions to Video

For almost all combinations of the previously discussed characteristics, a video-based method has been developed. With few exceptions (e.g. Kabra et al., 2021), these methods process the video $\{x^t\}_{t=1}^T$ sequentially and extract the current set of slots z^t from the previous set of slots z^{t-1}, and the current frame x^t. For scene-based slot methods, this is typically as simple as connecting the slots recurrently (see Fig. 3.3), potentially with a simple transition function to model slot movements and interactions (Greff et al., 2019; Weis et al., 2021; Kipf et al., 2022). In the Sec. 3.3.2, we will discuss this by the example of turning the image-based Slot Attention (Locatello et al., 2020) into the video-based SAVi (Kipf et al., 2022) method.

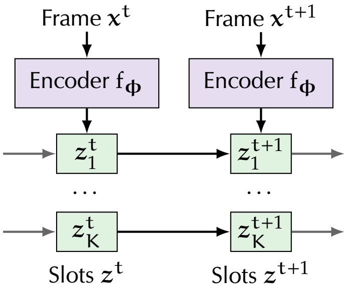


Figure 3.3: Slot model for video. Slots are recurrently connected through time, creating temporal consistency.

The recurrent connections naturally encourage *temporal consistency* among individual slots (cf. Sec. 2.3), thus providing some level of object tracking "*almost for free*" (Greff et al., 2019). The underlying assumption is that the set of objects is stable throughout the video, which is often not the case. To model appearing and disappearing objects, some spatial slot-based methods have sophisticated mechanisms to propagate slots, discover new objects and solve the re-identification problem (Kosiorek et al., 2018; Crawford and Pineau, 2020a; Jiang et al., 2020).

Some methods also support *dynamics modeling*, i.e. predicting the evolution of slots forward in time, which can be useful for some applications such as reinforcement learning (Veerapaneni et al., 2019; Jiang et al., 2020). For others, generic dynamics prediction modules have been developed that can be applied top of learned slot representations (Wu, Dvornik, et al., 2023).

3.2 Training Object-Centric Neural Networks

As discussed before, slot-based models are usually autoencoders, that is, they are trained end-to-end by learning to reconstruct the input data. We can differentiate between *generative* and *non-generative* approaches.

Generative Approaches In generative approaches, the scene x is modeled as a composition of multiple individual latent components z. The (now probabilistic) decoder parametrizes the conditional distribution $p_\theta(x \mid z)$, with the slots constituting the latent variables z, and the (probabilistic) encoder forms a variational approximation to the posterior $q_\phi(z \mid x)$ — we can view such models as structured extensions of variational autoencoders (Kingma and Welling, 2014). Consequently, they are trained with amortized variational inference by maximizing a lower bound to the data log likelihood $\log p(x)$[10]:

$$\mathcal{L}^{\text{ELBO}}(\theta, \phi, x) = \mathop{\mathbb{E}}_{q_\phi(z|x)}[\log p_\theta(x \mid z)] - D_{\text{KL}}(q_\phi(z \mid x) \parallel p_\theta(z)), \quad (3.1)$$

where $p_\theta(z)$ is the prior distribution over the latents. An interesting feature of probabilistic generative approaches is the modeling of (potentially multi-modal) distributions over the slots, which allows the expression of uncertainty and of different interpretations of ambiguous or multistable inputs (Greff et al. (2020); see also Fig. 3.4). Furthermore, such models can also *compositionally sample new scenes*, which has promising applications in controllable generation and scene editing (Jiang et al., 2023; Yanbo Wang et al., 2023; Wu, Hu, et al., 2023).

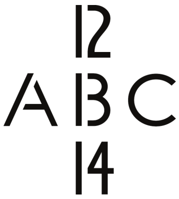


Figure 3.4: Multistability is a phenomenon in human perception where the same input is interpreted in several mutually exclusive ways (Attneave, 1971). Image from Greff et al. (2020) under CC-BY 4.0 license.

Non-Generative Approaches In contrast, non-generative approaches do not assume any particular model that generated the data. Instead, they can be trained using a reconstruction criterion such as the mean squared error (MSE) loss:

$$\mathcal{L}^{\text{rec}}(\theta, \phi, x) = \|x - g_\theta(f_\phi(x))\|^2. \quad (3.2)$$

It is noteworthy that non-generative approaches can also be made stochastic by injecting noise (e.g. Locatello et al., 2020), bringing them closer to probabilistic generative approaches but not optimizing any principled objective. Compared to generative approaches, non-generative approaches are generally simpler to formulate, implement, and extend with alternative or auxiliary objectives, e.g. to inject certain biases. We will see examples for this below.

Regularization One type of auxiliary objectives are *regularization losses* on the slots. They promote certain slot properties, such as sparsity (Fan et al., 2024), locality (Chakravarthy et al., 2023), suppression of similar slots (Nanbo and Fisher, 2021), bias towards human object annotations (Kim, Choi, et al., 2023) or moving objects (Bao et al., 2022), or cycle consistency between features and slots (Didolkar, Goyal, et al., 2024).

[10] The famous evidence lower bound (ELBO).

Another form of regularization enhances the model's compositional generalization by enforcing consistent encoding and decoding when combining slots from different samples, either in image space (Jung et al., 2024), or in slot space (Wiedemer et al., 2024).

Alternative Targets Another type of objective that has emerged in recent years is to *predict alternative targets* instead of reconstructing the input image. Examples for such targets include optical flow (Kipf et al., 2022), depth maps (Elsayed et al., 2022), and discrete tokens (Singh, Deng, et al., 2022). These targets can be considered as being related to the image through a (usually unknown) transformation, and so predicting them can be viewed as modeling that transformation.[11] The transformed targets offer a *different view* on the input data that may be substantially easier to predict from the slots compared to the original image. Furthermore, in this view, objects "stand out" more (cf. Fig. 3.5), *biasing the discovery process towards them*. Finally, specific targets embody certain *principles of object discovery* (see Sec. 2.4.2): for instance, optical flow encodes *coherent motion*; depth maps encode *spatial locality*. In this way, these targets have allowed object-centric models to handle more complex data. Indeed, in Part III of this thesis, we will discuss how using powerful self-supervised features as targets enables scaling to unconstrained real-world data.

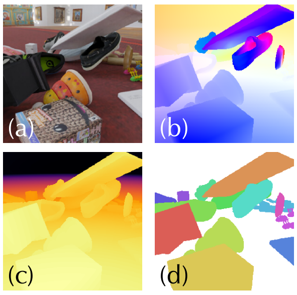


Figure 3.5: Different views on a scene: (a) RGB, (b) optical flow, (c) depth, (d) ground truth segmentation. Images from Greff et al. (2022) under CC-BY 4.0 license.

Beyond Autoencoding Besides autoencoding (and related paradigms such as predicting alternative targets), there are few other approaches to training. Notable exceptions include using the *EM algorithm* for training (Greff et al., 2017; Steenkiste et al., 2018), or various forms of *self-supervision*, such as contrastive losses (Kipf et al., 2020; Baldassarre and Azizpour, 2022), student-teacher prediction (R. Qian et al., 2023), cycle consistency (Ziyu Wang et al., 2023), or maximizing inter-slot variance while minimizing intra-slot correlations (Foo et al., 2023). For the self-supervised approaches, an interesting aspect is that they do not require a decoder, which reduces computational costs and frees them from modeling the (potentially complex) data domain. However, compared to the autoencoding framework, self-supervised methods are less explored and so their future potential is unclear.

3.3 Case Studies

In this section, we will conduct three case studies about object-centric models intended to make some of the principles discussed in the last section more concrete. They also serve as background material for Parts II and III of this thesis, in which we will use and build upon the models discussed here. In particular, we will discuss the Slot Attention model from Locatello et al. (2020) (used in Chap. 6), its extension to video, SAVi,[12] from Kipf et al. (2022) (used in Chap. 7), and the probabilistic SCALOR[13] model from Jiang et al. (2020) (used in Chap. 4).

[11] As these targets maintain the spatial layout of the image, the same decoders used for image reconstruction can be applied.

[12] SAVi ≙ **S**lot **A**ttention for **Vi**deo.

[13] SCALOR ≙ **SCAL**able **O**bject-oriented **R**epresentations.

3.3.1 Slot Attention

At the time of writing, Slot Attention (Locatello et al., 2020) is the most popular model for learning object-centric representations. Its popularity stems from its ease of use, flexibility and performance, which has allowed integration with a range of different tasks (e.g. Jiarui Xu et al., 2022; Reddy et al., 2023; Deng et al., 2024; Jiaqi Xu et al., 2024). In terms of our characterizations, Slot Attention performs *parallel inference* of *scene-based* slots in *feature space*. These choices make Slot Attention relatively efficient compared to other models. Slot Attention forms the basis of our Dinosaur model, which we will present in Chap. 6.

We need to distinguish between the *Slot Attention model for object discovery*, and the *Slot Attention module*, which were both proposed in Locatello et al. (2020). The Slot Attention *model* is an architecture for unsupervised discovery of object representations; in line with what was discussed in the previous section, it is based on autoencoding and trained through reconstruction with the MSE loss (Eq. (3.2)).[14] Its core component is the actual Slot Attention *module*, intended to be a generic, learnable, differentiable interface between perceptual features and symbol-like entities. As such, it is designed to be flexibly combinable with different neural components and loss functions.

Slot Attention Module Concretely, the Slot Attention module performs M iterations of attention between the set of slots $\boldsymbol{z}^i \in \mathbb{R}^{K \times D_{\text{slots}}}$ at iteration $i \in \{1, \dots, M\}$, and a set of input features $\mathbf{h} \in \mathbb{R}^{L \times D_{\text{feats}}}$. The input features are produced by an arbitrary upstream module that processes the input image, e.g. a convolutional neural network (CNN). If the features have spatial extent, as is the case for a CNN feature map, the feature map is flattened into a set after adding a *positional encoding* to retain information about the spatial positions of the individual feature vectors. The core of the module is an *inverted attention mechanism* that uses the slots as queries and the features as keys and values:

$$\operatorname{InvAtt}(\boldsymbol{z}, \mathbf{h}) := \underset{L}{\operatorname{WMean}}(\mathbf{A})\, \mathbf{h}\mathbf{W}_v, \tag{3.3}$$

where $\mathbf{A} \in [0,1]^{K \times L}$ is the matrix of attention weights,

$$\mathbf{A} = \underset{K}{\operatorname{softmax}}\left(\frac{1}{\sqrt{D_{\text{slots}}}} \boldsymbol{z}\mathbf{W}_q (\mathbf{h}\mathbf{W}_k)^T\right), \tag{3.4}$$

$\mathbf{W}_q \in \mathbb{R}^{D_{\text{slots}} \times D_{\text{slots}}}$, $\mathbf{W}_k \in \mathbb{R}^{D_{\text{feats}} \times D_{\text{slots}}}$, $\mathbf{W}_v \in \mathbb{R}^{D_{\text{feats}} \times D_{\text{slots}}}$ are learnable query, key, and value transforms, and WMean (weighted mean) and softmax are defined as[15]

$$\underset{L}{\operatorname{WMean}}(\mathbf{A})_{ij} = \frac{A_{ij}}{\sum_{l=1}^{L} A_{il}}, \qquad \underset{K}{\operatorname{softmax}}(\mathbf{X})_{ij} = \frac{\exp(X_{ij})}{\sum_{k=1}^{K} \exp(X_{kj})}. \tag{3.5}$$

Crucially, the softmax operation is computed over the *queries* instead of the *keys* as in standard attention (Vaswani et al., 2017); we will discuss

[14] It is thus a non-generative method, though a generative variant has also been developed (Yanbo Wang et al., 2023).

[15] Subscripts denote matrix indices.

the implications of this *inverted attention* below. A single iteration of slot attention is then succinctly described by

$$z^{i+1} = z^{i} + \mathrm{MLP}(\overline{\mathrm{GRU}}(z^{i}, \mathrm{InvAtt}(\bar{z}^{i}, \bar{h}))), \tag{3.6}$$

where MLP is a multi-layer perceptron with a single hidden layer, GRU is a Gated Recurrent Unit (Cho et al., 2014), and we let a bar denote the application of LayerNorm (Ba et al., 2016). The initial slots z^0 are either learnable parameters, or sampled from a normal distribution with learnable mean and variance; the latter has the advantage that the *number of slots can be varied at test time*. The complete Slot Attention module is described by:

$$\mathrm{SlotAtt}_\theta(h, z^0)\colon \mathbb{R}^{L\times D_{\text{feats}}} \times \mathbb{R}^{K\times D_{\text{slots}}} \mapsto \mathbb{R}^{K\times D_{\text{slots}}} := z^{M}. \tag{3.7}$$

Abstractly, we can view this operation as a mapping between *sets of different cardinalities* that is *permutation invariant* with respect to the input, and *permutation equivariant* with respect to the initial slots (Locatello et al., 2020).

Discussion Intuitively, Slot Attention *softly partitions* the input features between the slots. This is because the inverted attention creates *competition* between the slots to explain the inputs, which, over the iterations, leads to each input being approximately assigned to one slot — akin to a crystallization process.[16] Structurally, Slot Attention resembles a soft version of the k-means algorithm; it can thus be viewed as a learned, differentiable clustering algorithm. An important role for optimization stability is played by the weighted mean operation; its effect is that all slots receive an *equally large update* regardless of their success in the competition, and are neither "starved", nor "overloaded".[17] Indeed, recent work has shown that the two key ingredients necessary for object discovery with Slot Attention are the inverted attention and weighted mean operations (Y.-F. Wu et al., 2023).

Slot Attention Model The Slot Attention model employs a *structured decoder* that produces an independent image reconstruction per-slot, then combines the individual reconstructions via alpha compositing (Porter and Duff, 1984). In particular, letting $(\hat{x}_k, \hat{\alpha}_k) = g_\theta^{\text{slot}}(z_k)$ denote reconstruction and logits of the alpha masks, the final reconstruction $\hat{x} = g_\theta(z)$ is given by a weighted sum over the per-slot reconstructions $\hat{x}_k$:

$$\hat{x} = g_\theta(z) = \sum_{k=1}^{K} \alpha_k \odot \hat{x}_k, \qquad \alpha_k = \underset{K}{\mathrm{softmax}}(\hat{\alpha})_k, \tag{3.8}$$

where $\odot$ denotes the Hadamard product, and the alpha masks α are computed by a softmax over the slots. The slot-wise decoding function g_θ^{slot} is implemented by a *spatial broadcast decoder* (Watters et al., 2019): each slot is first broadcast to a 2D grid, embedded with a positional encoding, then upsampled to the resolution of the image with several

[16] This is supported by empirical observations that the iterations converge to a fixed point, i.e. $z^T \approx \mathrm{SlotAtt}_\theta(h, z^T)$ for sufficiently large T (Chang et al., 2022).

[17] A (potentially undesirable) consequence is that Slot Attention is setup to utilize *all available slots*, and does not naturally create *empty slots*.

[18] I hypothesize that in the broadcast decoder, the parallel decoding of spatial positions with weight sharing causes a bias towards low-frequency outputs, which translates to a *locality bias* when modeling objects. Some evidence for this is the stronger preference of the broadcast decoder towards modeling instances compared to other decoders, as we will discuss in Chap. 6.

deconvolutional layer.[18] Importantly, decoding is independent (except for the softmax), which limits the amount of information sharing that can happen between slots, and provides an inductive bias for slot separation (cf. the principle of *independence*, Sec. 2.4.2).[19] Furthermore, alpha compositing can be viewed as the slots competing for pixels to render to; in this sense, the composition at the end of decoding is the structural analogue to the decomposition at the end of encoding (within Slot Attention).

Finally, for purposes of introspection and evaluation, each slot representation z_k has two associated masks: the attention masks $\mathbf{A}_k^i$ (one set per iteration), and the decoding alpha masks $\boldsymbol{\alpha}_k$. Whereas the attention masks may be more helpful to interpret what content is encoded into the slots, the alpha masks are typically of higher quality (e.g. fit more closely to objects), and thus are used for evaluation purposes.

3.3.2 SAVi: Slot Attention for Video

SAVi (Kipf et al., 2022) is the extension of the Slot Attention model for sequential data, and thus also performs *parallel inference* of *scene-based* slots in *feature space* in terms of our characterizations. It is able to provide temporally consistent slot representations $\{z^t\}_{t=1}^T$ for videos $\{x^t\}_{t=1}^T$. In accordance with what was discussed in Sec. 3.1, this is achieved by recurrently connecting slots over time, which I will now describe in more detail. SAVi also forms the basis of the VideoSAUR model, which I will present in Chap. 7.

In particular, Slot Attention is applied to an encoding $\mathbf{h}^t$ of the current frame x^t and the slots from the previous time step z^{t-1} to yield the slots for the current step z^t:

$$z^t = z^{t,M} = \text{SlotAtt}_\theta(\mathbf{h}^t, z^{t,0}), \qquad z^{t,0} = z^{t-1} \tag{3.9}$$

where the initial slots z^0 for the first frame can be randomly initialized (as in Slot Attention). This process consists of two update loops: an outer loop over time (T steps), and an inner loop within Slot Attention (M steps); by mentally flattening the loops, we can interpret this as a single application of Slot Attention for $T \cdot M$ iterations with varying inputs $\mathbf{h}^t$. Thus, the previous slots z^{t-1} generally already provide a good estimate for the current slots z^t, and the number of iterations within Slot Attention can be reduced to $M = 1$. The recurrent initialization also facilitates *temporal consistency* of the slots, as it strongly biases Slot Attention to re-identify the object a slot previously belonged to in the current frame.

To *model temporal dynamics*, Kipf et al. (2022) propose to employ a learnable *transition module* $\mathrm{t}\colon \mathbb{R}^{K \times D_{\text{slots}}} \mapsto \mathbb{R}^{K \times D_{\text{slots}}}$ that takes in the slots of the previous time step and produces Slot Attention's initialization for the current step, i.e. $z^{t,0} = \mathrm{t}(z^{t-1})$. Kipf et al. (2022) instantiate the transition module with a Transformer encoder (Vaswani et al., 2017); its self-attention mechanism is able to model interactions between objects (e.g. collisions). Note that the transition module is generally *not* able

[19] Recent theoretical works have even shown that independent decoding facilitates provably (1) identifying the ground truth representation (Brady et al., 2023), and (2) compositional generalization (Wiedemer et al., 2024).

to predict the future evolution of the slots, as there is no explicit loss training it to do so. Functionally, the role of the transition module is only to produce a *good initialization* for Slot Attention, which may or may not coincide with the true slot dynamics.

3.3.3 SCALOR

SCALOR (Jiang et al., 2020) is a probabilistic generative model for learning slot representations of videos. In terms of our characterizations, SCALOR performs *parallel inference* of *spatial* slots both in *pixel and feature space*; it was designed to scale to many objects and demonstrated to handle up to 100 objects in simple scenarios. We will demonstrate how the structured representations SCALOR infers can be utilized for self-supervised reinforcement learning in Chap. 4.

As discussed in Sec. 3.1.1, spatial slot methods add further structure to the slots. In SCALOR's case, it is assumed that the slots z^t that generate the video frame x^t factorize into a background latent variable $z^{\text{bg},t}$ and the foreground object slots $z^{\text{fg},t} = \{z_k^t\}_{k \in \mathcal{X}^t}$, where $\mathcal{X}^t$ is the set of object indices for time t. Each foreground slot is then further structured as $z_k^t = \left(z_k^{\text{where},t}, z_k^{\text{what},t}, z_k^{\text{pres},t}\right)$, encoding respectively location, appearance, and existence of an object, where the location is given by $z_k^{\text{where},t} = \left(z_k^{\text{pos},t}, z_k^{\text{scale},t}, z_k^{\text{depth},t}\right)$, i.e. the center position, scale, and relative depth to the camera.

Generative Process In the SCALOR model, the joint distribution $p(x^{1:T}, z^{1:T})$ temporally factorizes as follows[20]:

$$p(x^{1:T}, z^{1:T}) = \prod_{t=1}^{T} p(x^t, z^t \mid z^{<t}), \tag{3.10}$$

where the per-frame conditional is given by $p(x^t, z^t \mid z^{<t})$

$$= \begin{cases} \underbrace{p(x^1 \mid z^1)}_{\text{rendering}} \quad \underbrace{p(z^{\text{bg},1})}_{\text{background prior}} \quad \underbrace{p(z^{\mathcal{D},1})}_{\text{discovery prior}} & \text{for } t = 1, \\ \underbrace{p(x^t \mid z^t)}_{\text{rendering}} \underbrace{p(z^{\text{bg},t} \mid z^{\text{bg},<t}, z^{\text{fg},t})}_{\text{background transition}} \underbrace{p(z^{\mathcal{D},t} \mid z^{\mathcal{P},t})}_{\text{discovery}} \underbrace{p(z^{\mathcal{P},t} \mid z^{<t})}_{\text{propagation}} & \text{for } t > 1. \end{cases} \tag{3.11}$$

Here, $z^{\mathcal{D},t}$ is the set of slots for objects newly *discovered* in the current frame, and $z^{\mathcal{P},t}$ is the set of slots *propagated* from the previous frame (with $\mathcal{D} \cup \mathcal{P} = \mathcal{X}$, i.e. $z^{\text{fg},t} = z^{\mathcal{D},t} \cup z^{\mathcal{P},t}$). Thus, the generative process decomposes into prior, propagation, discovery, background transition, and rendering modules, which in turn can be further decomposed; for more information, we refer to Jiang et al. (2020). Here, we only provide some high-level comments: what differentiates SCALOR from previous slot-based models for video (e.g. Kosiorek et al., 2018) is its parallelization of the discovery process, resulting in efficient processing and enabling

[20] Unless necessary, we omit the parametrization θ, ϕ from p_θ, q_ϕ for notational clarity.

scalability to many objects. In addition, it contains a *propagation-discovery model*, in which new objects are proposed and subsequently rejected if they spatially overlap with propagated objects. For proposing new objects, the image is divided into latent grid cells, each containing a potential object proposal.

Inference We now discuss the encoding direction, i.e. how slots are inferred for a video. As the true posterior distribution $p(\mathbf{z}^{1:T} \mid \mathbf{x}^{1:T})$ is intractable, Scalor makes a variational approximation to the posterior, which decomposes similarly to the generative model:

$$q(\mathbf{z}^{1:T} \mid \mathbf{x}^{1:T}) = \prod_{t=1}^{T} q(\mathbf{z}^{t} \mid \mathbf{z}^{<t}, \mathbf{x}^{\leqslant t}) \tag{3.12}$$

$$= \prod_{t=1}^{T} \underbrace{q(\mathbf{z}^{\mathrm{bg},t} \mid \mathbf{z}^{\mathrm{fg},t}, \mathbf{x}^{t})}_{\text{background}} \underbrace{q(\mathbf{z}^{\mathcal{D},t} \mid \mathbf{z}^{\mathcal{P},t}, \mathbf{x}^{\leqslant t})}_{\text{discovery}} \underbrace{q(\mathbf{z}^{\mathcal{P},t} \mid \mathbf{z}^{<t}, \mathbf{x}^{\leqslant t})}_{\text{propagation}}. \tag{3.13}$$

The posterior uses an analogue propagation-discovery process to the generative direction. Furthermore, within the posterior propagation and discovery modules, spatial transformer networks (Jaderberg et al., 2015) are used to *attend* to rectangular regions of the feature map and the image, both to identify object locations $\mathbf{z}^{\mathrm{where},t}$ and to encode appearances $\mathbf{z}^{\mathrm{what},t}$. *Training* proceeds by maximizing the following evidence lower bound (an instantiation of the general ELBO in Eq. (3.1)):

$$\begin{aligned}\mathcal{L}^{\mathrm{ELBO}}(\theta, \phi, \mathbf{x}^{1:T}) = \sum_{t=1}^{T} \mathop{\mathbb{E}}_{q_\phi(\mathbf{z}^{<t} \mid \mathbf{x}^{<t})} \Big[& \mathop{\mathbb{E}}_{q_\phi(\mathbf{z}^{t} \mid \mathbf{z}^{<t}, \mathbf{x}^{\leqslant t})} \big[\log p_\theta(\mathbf{x}_t \mid \mathbf{z}_t)\big] \\ & - D_{\mathrm{KL}}\big[q_\phi(\mathbf{z}^{t} \mid \mathbf{z}^{<t}, \mathbf{x}^{\leqslant t}) \,\big\|\, p_\theta(\mathbf{z}^{t} \mid \mathbf{z}^{<t})\big]\Big].\end{aligned} \tag{3.14}$$

Discussion Scalor has a sophisticated model of the generative process, in principle capable of capturing events such as the appearance, disappearance and occlusion of objects. However, the model's complexity makes inference challenging: Jiang et al. (2020) report a phenomenon they call *propagation collapse*, wherein the model re-discovers the objects each frame rather than propagating them. While the authors introduce measures to mitigate this issue, in our own project (Chap. 4), we found Scalor's tracking of objects to be unreliable. Furthermore, the numerous modules involve many hyperparameters that must be correctly set; while this allows Scalor to be fine-tuned to specific datasets, it also makes the model brittle and labor-intensive to get to work. Finally, underlying the model are many simplifying assumptions (e.g. objects have a certain scale; objects can only move a certain distance per step); while these assumptions encode useful inductive biases that enable successful learning of object representations on simpler scenes, they often fail in more complex scenes. These factors may explain why Scalor has not yet been scaled to more real-world scenarios and has fallen out of favor compared to simpler approaches that make less assumptions.

3.4 Evaluating Object-Centric Representations

For all representation learning methods, the question of evaluation is important but challenging. It is important, because although representations are often learned in isolation, they cannot be considered in a vacuum — instead, representations are supposed to support a narrow or broad set of *downstream tasks*. To develop better representations, we thus need a way to measure progress towards these tasks of interest. It is challenging, because as a consequence of being unsupervised, a natural evaluation metric is missing. Specifically, the value of the training objective is often not predictive of the downstream performance.[21] Thus, the crux of evaluation is finding metrics that measure what we actually care about.

In this section, we are going to introduce different ways of evaluating object-centric models. First, we review the currently most popular approach to evaluation, namely how well the models *discover objects* (Sec. 3.4.1). Then, we consider approaches to measure *representational content* (Sec. 3.4.2). After discussing *model robustness and generalization* (Sec. 3.4.3), we also consider the direct evaluation of representations on *downstream tasks* (Sec. 3.4.4).

3.4.1 Object Discovery

The primary approach to how slot representations have been evaluated is in terms of object discovery, that is, how well the learned representations capture the objects within the input image or video. To do so, annotations marking the objects in the target dataset are required; typically, human-labeled segmentation masks are used for this purpose. By comparing the object masks associated with the slots with these ground truth annotations, we aim to measure (1) how accurately individual slots fit the *position and shape* of individual objects; and (2) how well the slots *split* the different objects. Both qualities assess how well the discovered objects align with the human-defined notion of objects on the data, but serve different purposes: while (1) measures the quality of mask fit, (2) evaluates how well the objects are *segregated from each other* within the representation. The two qualities are correlated with each other, making it challenging to measure them in isolation; nevertheless, different metrics balance the two aspects differently. For the projects discussed in this thesis, we mainly relied on two metrics: to measure quality (1), the *mean best overlap*, and to measure quality (2), the *adjusted rand index*.

Recall that $\mathbf{m} = \{\mathbf{m}_k \in [0,1]^{H \times W}\}_{k=1}^K$ is the set of soft segmentation masks associated with the slots. We define $\mathcal{S} = \{\mathcal{S}_k\}_{k=1}^K$ as the set of pixels for which $\mathbf{m}_k$ has the highest probability[22] and $\mathcal{T} = \{\mathcal{T}_t\}_{t=1}^T$ to

[21] E.g. for object-centric models trained with reconstruction, the loss was shown to not correlate well with the metrics of interest (Dittadi et al., 2022).

[22] $\mathcal{S}_k = \{j \in \{1, \ldots, HW\} \mid k = \arg\max_i \mathbf{m}_{i,j}\}$.

be the set of ground truth segmentation masks, where $\mathcal{T}_t \in \{0,1\}^{H \times W}$ contains the set of pixels marked in mask t.

Mean Best Overlap The mean best overlap (mBO) (Uijlings et al., 2013; Pont-Tuset, Arbeláez, et al., 2017) is a metric that measures how much the predicted and ground truth object segmentation masks overlap. It is based on the well-known intersection-over-union (IoU) metric, also known as the Jaccard index. The IoU between two sets $\mathcal{A}$ and $\mathcal{B}$ is defined as

$$\mathrm{IoU}(\mathcal{A}, \mathcal{B}) := \frac{|\mathcal{A} \cap \mathcal{B}|}{|\mathcal{A} \cup \mathcal{B}|} \in [0,1]. \tag{3.15}$$

Notably, the IoU is scale invariant and thus it can be meaningfully averaged when comparing both small and large objects within an image.

The predicted masks lack a specific order, making a direct one-to-one comparison with the ground truth masks impossible; thus, a way of *matching* is necessary. Specifically, mBO assigns each predicted mask to the ground truth object mask with the highest overlap, and normalizes by the number of ground truth masks:

$$\mathrm{mBO}(\mathcal{S}, \mathcal{T}) := \frac{1}{|\mathcal{T}|} \sum_{\mathcal{T}_t \in \mathcal{T}} \max_{\mathcal{S}_k \in \mathcal{S}} \mathrm{IoU}(\mathcal{S}_k, \mathcal{T}_t) \in [0,1]. \tag{3.16}$$

The mBO has also been referred to as mean segmentation covering (Arbeláez et al., 2011; Engelcke, Kosiorek, et al., 2020; Dittadi et al., 2022). An alternative to maximum mask matching is to solve a bipartite matching problem, e.g. using the Hungarian method (Kuhn, 1955), which is referred to as mIoU in the object-centric literature (e.g. Karazija et al., 2021). The two matching variants mostly[23] exhibit similar behavior, with mBO upper-bounding mIoU: $\mathrm{mBO}(\mathcal{S}, \mathcal{T}) \geqslant \mathrm{mIoU}(\mathcal{S}, \mathcal{T})$.

Adjusted Rand Index The adjusted rand index (ARI) (Hubert and Arabie, 1985) is a measure of *cluster similarity*. Note that *segmenting* an image can be interpreted as partitioning its pixels into clusters; if $\mathcal{P} = \{1, \ldots, HW\}$ is the set of image pixels, we can interpret $\mathcal{S} = \{\mathcal{S}_1, \ldots, \mathcal{S}_K\}$ and $\mathcal{V} = \{\mathcal{T}_1, \ldots, \mathcal{T}_T\}$ as two clusterings (partitions) of $\mathcal{P}$. Thus, the ARI can be used to compare the two clusterings $\mathcal{S}$ and $\mathcal{T}$.

ARI[24] is based on *counting of pairs of items* of $\mathcal{S}$ that are assigned to the same or different clusters within $\mathcal{S}$ or $\mathcal{T}$. In particular, let N_{11} be the number of pairs that are in the same cluster in both $\mathcal{S}$ and $\mathcal{T}$ and N_{00} the number of pairs that are in different clusters in both $\mathcal{S}$ and $\mathcal{T}$. With this, the *rand index* (Rand, 1971) is defined as

$$\mathrm{RI}(\mathcal{S}, \mathcal{T}) := \frac{N_{00} + N_{11}}{\binom{HW}{2}} \in [0,1], \tag{3.17}$$

that is, the fraction of all pairs of items for which $\mathcal{S}$ and $\mathcal{T}$ are *in agreement*. An RI of 1 indicates that $\mathcal{S}$ and $\mathcal{T}$ are identical up to permutation of the clusters.

[23] If the number of predicted masks is less than the number of true masks for an image, mBO may be overly optimistic, whereas mIoU penalizes the unassigned true masks (false negatives).

[24] The following exposition is based on Vinh et al. (2010).

As the rand index's baseline value shifts strongly depending on the number of clusters and their items (Fowlkes and Mallows, 1983), the adjusted rand index normalizes the rand index such that a random clustering yields an expected value of zero. Let denote N_{01} the number of pairs in the same cluster in $\mathcal{S}$ but in different clusters in $\mathcal{T}$, and N_{10} the number of pairs in different clusters in $\mathcal{S}$ but the same cluster in $\mathcal{T}$. Then the adjusted rand index is defined as

$$\mathrm{ARI}(\mathcal{S}, \mathcal{T}) := \frac{2(N_{00}N_{11} - N_{01}N_{10})}{(N_{00}N_{01})(N_{01}N_{11}) + (N_{00}N_{10})(N_{10}N_{11})} \leqslant 1, \qquad (3.18)$$

with a value of 0 for a random clustering in expectation, and a value of 1 for identical clustering up to permutation; negative values are also possible.

In the context of object-centric learning, the *foreground adjusted rand index* (FG-ARI) is mostly used: this variant only takes the ground truth masks of foreground objects into account. Thus, pixels lying in the background are fully ignored by the metric, which focuses the metric on correct separation of objects, but does not take the into account how tight the predicted masks fit the objects. Because this creates certain pathological cases,[25] several works have questioned the use of FG-ARI and expressed a preference for IoU-based metrics (Engelcke, Kosiorek, et al., 2020; Karazija et al., 2021; Wu, Hu, et al., 2023).

Other Considerations To evaluate object discovery with *video data*, an average of the image-based metrics applied frame-by-frame can be computed. However, for sequential data, it is typically of interest how *temporally consistent* the object representations are; this can be achieved by joining the per-frame masks into "temporal tubes" for each object, and then applying the image-based metrics (Kipf et al., 2022). If slots switch their associated objects in the video, i.e. fail to track them, this would result in an inconsistent tube mask and be penalized in the metrics.

On real-world datasets, the ground truth segmentation masks typically have an associated *class label*. By joining all object (instance) masks with the same class label, a *semantic segmentation mask* can be obtained. In Seitzer et al. (2023) (see Chap. 6), we proposed to compute a *semantic mBO* using semantic segmentation masks. Comparing the semantic mBO with the mBO computed with instance masks allowed us to evaluate whether models are *biased towards semantic segmentation* (cf. Fig. 3.6).

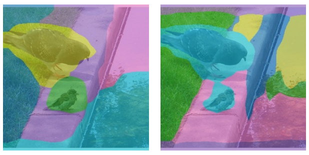


Figure 3.6: Instance vs. semantic slot grouping. The two modes can be distinguished by comparing *semantic mBO* with *instance mBO*. From Seitzer et al. (2023).

3.4.2 Representation Content

Another approach to evaluating slot representations is to measure their *information content*. Optimally, each slot should *completely "describe"* the object it is assigned to, and be *independent* of all other objects in the image (the properties of completeness and separation, see Sec. 2.3). What exactly the description of an object should include is dependent on the data domain as well as the task the representation is intended for. Typically evaluated properties include the position, size, shape, color,

[25] For example, for an image containing a single object, predicting a single mask that covers the full image yields an FG-ARI of 1, while predicting even a single pixel of the mask wrongly yields a value of 0.

type, or class of the object (Dittadi et al., 2022). More fine-grained object properties[26] are also conceivable, but have so far not been studied in the object-centric literature.

Completeness Each slot should encode *all* properties of the object it captures. Checking for completeness amounts to training a classifier or regressor per target property, predicting that property from the slot.[27] The average accuracy of the predictors can be taken as a measure of completeness. Furthermore, predictors of different complexity can be employed (e.g. linear, or a shallow MLP), which can give an estimate of how easily *accessible* the object properties are within the slots.

Note that as the same predictors are shared between slots, their accuracy is also affected by whether the slots share a *common representational format*. For example, if an object's location is encoded in different ways for different object types, a shared predictor will not be able to consistently "read-out" that property. The author of this thesis is not aware of any metrics measuring the degree of "representation sharing". However, as linear predictors will be particularly affected by a lack of shared representational format (as they can not compensate for it internally), their performance could be used as an indicator for it.

Disentanglement In addition to measuring whether a slot encodes certain object properties, it may be desirable to evaluate how well the individual properties are *isolated* in the representation.[28] This is referred to as *disentanglement*, and a variety of metrics have been proposed to measure it (e.g. Higgins, Matthey, et al., 2017; R. T. Q. Chen et al., 2018; Eastwood and Williams, 2018). Note that it has been shown that unsupervised disentanglement is impossible in general; there also is conflicting evidence about its practical benefits (Locatello et al., 2018; Dittadi et al., 2021).

Separation A further consideration is the separation, i.e. the independence, of the slots from each other.[29] The author of this thesis is not aware of any works attempting to measure slot independence. Measuring separation involves testing that information that is encoded in one slot is *not* encoded in any of the other slots. For this purpose, Dang-Nhu (2022) propose a metric derived from the disentanglement metric DCI (Eastwood and Williams, 2018). However, separation is rarely evaluated in practice.

3.4.3 Robustness and Generalization

For any ML model, it is important to understand how the model responds to unforeseen inputs, i.e. inputs that lie outside the training distribution. A central argument for object-centric models is that they are more robust and generalize better than models with less structure; several studies have tried to validate this claim (Karazija et al., 2021; Dittadi et al., 2022; Yoon et al., 2023). This involves evaluating object discovery and representation content on test data that has undergone certain *distribution shifts*

[26] E.g. using the COCO Attributes dataset (Patterson and Hays, 2016): male/female, young/old, etc.

[27] As the slots have no particular order, training and evaluation is phrased as *the prediction of a set of properties for the target image*, using Hungarian matching to assign predicted to true properties (Dittadi et al., 2022).

[28] Intuitively, each property should be assigned an individual dimension in the representation's vector space.

[29] Separation can be seen as *inter-slot disentanglement*.

compared to the training data. For simulated datasets, controlled changes to the data distribution are possible; examples of such modifications include varying the number of objects, introducing new object types, new object colors or textures, or backgrounds. With such data, it is also possible to systematically evaluate *compositional generalization* (Wiedemer et al., 2024). A specific form of compositional generation is *compositional generation*, which can be evaluated by modifying one or multiple slots (e.g. combining slots from different images), and then testing whether the modified slots are decoded into a meaningful composite image (Jiang et al., 2023; Yanbo Wang et al., 2023; Wu, Hu, et al., 2023).[30]

A more top-down approach to test for generalization is to train the model on one dataset, and evaluate its performance on a target dataset from a different data distribution. If the model performs well on a sufficiently broad range of target datasets, this is evidence that the model also generalizes "in-the-wild" — such *zero-shot generalization* is a hallmark of so-called foundation models. In a recent study, we demonstrated that object-centric models based on the work presented in Part III of this thesis indeed exhibit some zero-shot capabilities (Didolkar, Zadaianchuk, et al., 2024, see also Sec. 8.2.3 for a longer discussion).

3.4.4 Downstream Tasks

Eventually, slot representations are supposed to be part of a larger system serving a particular purpose. Downstream evaluations judge the quality of the representation by how well it supports that purpose.[31] For example, slots may be used to encode images in a model for visual question answering (VQA); in that case, the quality of the answers can be used to compare different slot models (Mamaghan et al., 2024). This treats the slot representation as a black box, disregarding the properties of individual slots.

This style of evaluation represents a pragmatic approach, where the ultimate focus is on the outcome. One of its advantages is that the slot representations can be fairly compared to other means of constructing the overall system (e.g. using human supervision). Furthermore, it is also applicable when the target dataset lacks human labels that would allow a more fine-grained evaluation of the slots. However, it may also overlook certain benefits of object-centric representations (e.g. out-of-distribution generalization) if the downstream evaluation is not designed to cover those aspects.

Evaluation through downstream tasks has historically received less attention than the mask-based evaluation of object discovery.[32] This may be due to a lack of suitable downstream tasks, as well as the complexities of implementing end-to-end training and evaluation pipelines. Focusing on a direct evaluation of the representation instead allows researchers to compartmentalize the representation learning problem. From the point of view of engineering ML systems, this also offers practical advantages such as modularizing models into manageable parts, and achieving faster turnaround times. However, it carries the risk of losing sight of

[30] This can be viewed as evaluating the ability to perform causal, counterfactual modeling (cf. Sec. 2.2.2).

[31] Another consideration besides performance is sample efficiency, i.e. the number of data points needed to obtain a certain level of performance.

[32] Exceptions include Veerapaneni et al. (2019) and Yoon et al. (2023) for RL-based evaluation.

the greater purpose for learning representations in the first place, which is, for the most part, solving problems in the real world (cf. Goodhart's law[33]). In Chap. 8, we will discuss two real-world applications for object-centric representations, VQA and robotics, and how the research in this thesis may enable them.

[33] An adage typically stated as "*when a measure becomes a target, it ceases to be a good measure*."

Part II

On the Benefits of Structured Object Representations for Autonomous Agents

Chapter 4

Self-Supervised Visual Reinforcement Learning with Object-Centric Representations

In the last part of this thesis, I introduced structured object representations: what they are, the purpose they are serving, and models for learning them. Part II of this thesis now places structured object representations in the context of a larger learning system.[1] As the area of application, I will focus on *autonomous agent learning*. Given that object representations are particularly suited to describe physical environments, I will discuss embodied agents that interact with such environments: *robots*.[2] In doing so, I will highlight the practical benefits and potential of these representations when integrated into a larger learning system.

In this chapter, I introduce a reinforcement learning (RL) method, Smorl,[3] that is capable of tackling a challenging class of problems through the use of structured representations. In particular, Smorl is designed for visual RL environments with a combinatorial goal structure: by decomposing the goal space with the help of object-centric representations, complex composite tasks are turned into a series of simpler, manageable sub-goals.

This work highlights three interesting aspects enabled by object-centric representations: the ability to understand environment observations in terms of their *latent compositional structure*, allowing agents to operate on an abstraction level that matches the underlying structure; the ability to design *informed algorithms* that leverage that structure effectively; and the ability to *generalize* to scenarios not encountered during training. Within the context of this thesis, this chapter contains the first of two case studies illustrating the benefits of structured object representations through the lens of autonomous agent learning.

A short disclaimer is in order. The next two chapters will assume high-level familiarity with terms and concepts from the field of RL. I opted to omit background material on RL to focus the thesis on structured object representations. Instead, I refer the interested reader to Sutton and Barto (2018) for an extensive introduction to RL.

This chapter summarizes

Andrii Zadaianchuk*, **Maximilian Seitzer***, and Georg Martius (2021). "Self-Supervised Visual Reinforcement Learning with Object-Centric Representations". In: *International Conference on Learning Representations (ICLR)*. *equal contribution

The full publication can be found in Appendix A.

[1] In the framework of Greff et al. (2020) (see Chap. 1), this is the *composition* of object representations *"to construct new inferences, predictions, and behaviors"*.

[2] More specifically: robotic manipulators. For practical purposes, we restrict ourselves to simulated robots.

[3] Smorl ≙ **S**elf-supervised **M**ulti-**O**bject **RL**.

4.1 Motivation

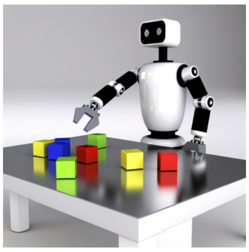


Illustration of the scenario of interest, generated with DALL·E 3.

Imagine the following scenario. A robot is placed in front of a table with multiple objects on it. The only means the robot has for sensing its environment is a fixed camera looking onto the table. When prompted with a "goal image" of the table, the robot should arrange the objects on the table accordingly. The robot should use reinforcement learning to learn this task fully unsupervised; crucially, it is never told whether it succeeded or not by an external observer.

This scenario describes an instance of *self-supervised multi-goal compositional visual reinforcement learning*. At the time this work was conducted, even basic versions of this class of tasks were considered unsolved. Let us pick apart the different aspects to understand why this scenario is so difficult.

(I) *"Self-supervised"*: the environment does not provide a reward signal to the agent. This also means that the tasks the agent should solve are not revealed to the agent a priori. Thus, the agent needs to create its own supervision to prepare for *any* potential task.

(II) *"Multi-goal"*: at test time, the agent's tasks are sampled from a *distribution of tasks*. The agent cannot specialize to a single goal.

(III) *"Compositional"*: the environment exhibits a compositional structure, e.g. it consists of multiple objects. This means that the agent needs to deal with a combinatorial explosion of possible states and goals.

(IV) *"Visual"*: the agent only receives high-dimensional image observations from the environment, without any proprioceptive information about the agent's state. On top of the reinforcement learning problem, the agent needs to solve the *representation learning* problem: organizing its perception in order to act.

Combinations of these aspects are even more challenging. For example, while there exist algorithms for multi-goal RL from *states* (e.g. Andrychowicz et al., 2017), multi-goal RL from images is challenging because in image space, the set of possible goals is too large to explore; an appropriate goal space must be *learned* (A. Nair et al., 2018). Furthermore, for multi-goal RL with a *compositional* state space, the set of possible goals grows exponentially with the number of elements of the state space. Finally, a compositional state space with *image observations* complicates the representation learning problem, because the elements of the state space have to be separated into distinct representational units (i.e. the binding problem (Greff et al., 2020)).

As such, prior work before this project had only tackled subsets of this challenges. Laversanne-Finot et al. (2018), A. Nair et al. (2018), Ghosh et al. (2019), A. Nair et al. (2019), Warde-Farley et al. (2019), and Pong et al. (2020) attempted self-supervised multi-goal RL from images, but they all assumed a monolithic state space that can be encoded into a single vector representation. In their environments, at most a single target

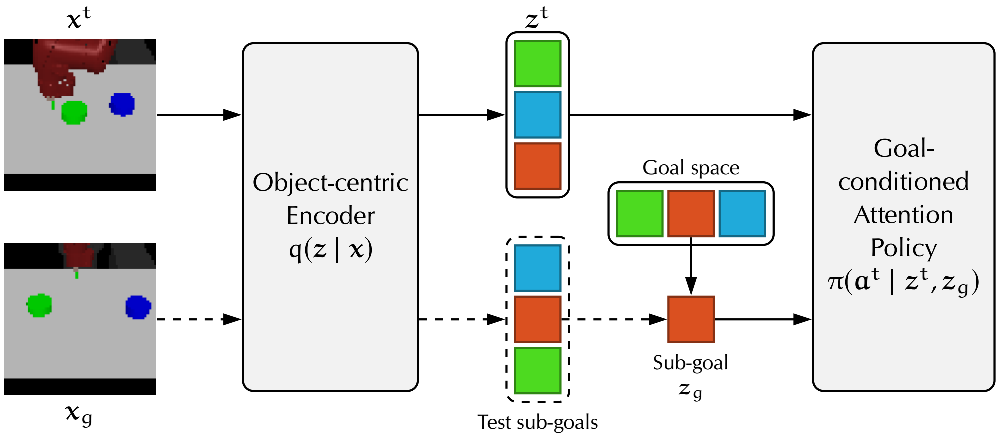


Figure 4.1: The SMORL agent. The current observation x^t is encoded into a set of slot vectors z^t and processed by the goal-conditioned policy $\pi(a^t \mid z^t, z_g)$. During training, sub-goals z_g are sampled from a learned goal space conditional on the first environment observation z^1 (not depicted). At test time (dashed lines), the externally provided goal image x_g is encoded into a set of potential sub-goals, which are then sequentially attempted to be solved by the agent. Figure adapted from Zadaianchuk et al. (2021).

object is present. Several of these works rely on variational autoencoders (VAEs) (Kingma and Welling, 2014) to learn a representation of the state, which is then also used for goal setting and reward shaping. A motivation for this work was that such VAE-based representations are not suitable for multi-object environments, exactly because they fail to solve the binding problem (Greff et al., 2020). Instead, the SMORL agent employs object-centric representations to obtain a more tractable goal and state representation.

4.2 The SMORL Agent

The SMORL[4] agent consists of two main components — an *object-centric encoder* and a *goal-conditioned attention policy* — trained separately in two stages. In the first stage, the encoder is trained offline on a dataset of observation sequences from the environment.[5] In the second stage, the policy is trained online in the environment with reinforcement learning using the SMORL algorithm. At test time, the trained agent is able to solve unseen goals by decomposing them into a sequence of sub-goals. I will first describe the encoder and policy components before detailing the SMORL algorithm. See Fig. 4.1 for an overview.

4.2.1 Architectural Components

Object-Centric Encoder The task of this component is to learn an object-centric representation of the environment's observation space. In this work, we used the SCALOR model[6] (Jiang et al., 2020); in principle, SMORL is flexible in terms of the representation, and thus other models could be used instead of SCALOR. The only requirements are that (1) the representation is structured as a set of slot vectors z; (2) each slot vector

[4] SMORL ≙ **S**elf-supervised **M**ulti-**O**bject **RL**.

[5] Collected by a random policy.

[6] SCALOR was introduced in Sec. 3.3.3

z_k is *additionally structured* in terms of *position* z_k^{where} and *appearance* z_k^{what} of the object[7]; (3) the appearance vectors "*uniquely identify*" objects throughout an episode.

Let us briefly discuss Scalor's training in the context of Smorl. Recall that Scalor is a generative sequence model, that is, it is trained to model a distribution of sequences. In our case, the sequences to be modeled are episodes of environment observations. Thus, we first executed a random policy that collects a dataset of episodes from the environment and then fitted Scalor to this dataset using variational inference (see Sec. 3.3.3). Note that in our case, the modeled sequences come from an *active setting*, i.e. the agent's actions influence the future observations in the sequence. To accommodate this, we extended Scalor to include actions in its transition model.[8]

Goal-Conditioned Attention Policy The object-centric encoder produces a set of slot vectors per time step. How should that set be processed to select the agent's action, i.e. what is a suitable design for the policy? Typically, the policy networks used in RL are MLPs that take a vector as input and map it to an action; these are unsuitable to process sets of vectors. Instead, we designed a neural network based on the *multi-head attention mechanism*[9] popularized by Transformers (Vaswani et al., 2017).[10] In particular, the set of slot vectors is used as *keys and values*; the goal object is used as a *query*. The (fixed-length) output of the attention mechanism is then used as input to an MLP that outputs the agent's action.[11] Intuitively, attention implements a filtering mechanism for retrieving the part of the state that is relevant under the current goal. Note how the use of object-centric representations enables the policy to directly operate on meaningful, task-relevant units of information, alleviating the policy from having to discover this information from the sparse RL training signal.

4.2.2 SMORL Algorithm: Training Phase

With the architectural components in place, we can now turn our attention to the Smorl training algorithm. For any self-supervised multi-goal RL agent, (at least) the following three questions need to be answered:

(I) *Goal space*: what aspects of the environment are used as goals, and how are goals represented to the agent?

(II) *Goal sampling*: how are goals selected during training?

(III) *Reward function*: how are the current goal and observation transformed into a reward?

One particular contribution of Smorl is answering these questions specifically for compositional multi-object environments.

Goal Space Assuming that the environment has a compositional structure that can be inferred with object-centric representations, a natural

[7] Note that object-centric models with scene-based slots (e.g. Slot Attention, see Sec. 3.1.1) do *not* qualify because of requirement (2).

[8] This is what is typically described as a "world model" (Ha and Schmidhuber, 2018).

[9] We also explored other forms of permutation-invariant neural networks (e.g. Deep Sets (Zaheer et al., 2017)), but found that the attention-based design worked best.

[10] At the time, Transformers were not as common in RL, and so this choice was not as obvious as it might seem in retrospect.

[11] A similar design is used for the value function inside the RL algorithm used to train the policy.

idea is to use the same structure for the goal space. Through this compositional structure, the goal space can then be decomposed into simpler "sub-goals", where each sub-goal corresponds to the intended target state of one particular entity. More complex goals can then be achieved by "chaining" sub-goals together (see Sec. 4.2.3). This way, a complex goal is broken into more manageable parts, simplifying the learning process for the agent. The assumptions behind this are that each entity in the environment (1) constitutes a valid goal,[12] and; (2) can be controlled *independently* of the other entities. A consequence of the latter assumption is that the overall goal can be achieved by achieving *any* sequence of sub-goals. Of course, this assumption is usually too naive; we discuss possible resolutions for when it is violated in Sec. 4.4.

Concretely, Smorl uses the representation space of object vectors from the object-centric encoder as the goal space. Thus, the policy is conditioned on exactly one object vector at time, specifying the object to be manipulated. This assumes two properties of the object-centric representations: (1) the object vectors are *interchangeable*, that is, all different types of objects share a common representational format; and (2) the goal object can be *re-identified* within any environment observation, that is, the object vector uniquely describes the goal object such that the current state of that object in the environment can be retrieved from the full set of objects. While property (1) is usually fulfilled due to the way object-centric models are trained, property (2) may be violated if distinct objects "overlap" in representation space. In this case, the policy may confuse which goal object should be manipulated. For the environments used in this work, we verified that in Scalor's appearance space, objects are indeed distinguishable (see Zadaianchuk et al., 2021, App. A.1).

Goal Sampling Having established the goal space, I now describe how the Smorl agent picks goals from that space during training. At the start of each training episode, the initial environment observation is encoded into a set of object vectors. Each of those objects constitutes a potential sub-goal; the Smorl agent simply selects a random object to be manipulated in this episode. To turn the corresponding object vector into a meaningful goal, the *positional component* of the representation (recall that Scalor's representation is structured in terms of position and appearance) is replaced with a new position sampled from a learned distribution over valid environment positions.[13] By using the initial observation to pick goals, we ensure that each sub-goal is feasible in the current episode. This process can be viewed as the agent "imagining" one of the objects in its initial observation at another location and then realizing that imagination by trying to reach that state.

Reward Function What is missing is how the agent is guided toward its self-selected goals, i.e. the reward specification. Previous work (A. Nair et al., 2018; Pong et al., 2020) showed that the *distance in latent space* between the current state and the goal state can be a good reward

[12] An invalid goal, for example, could be an entity in the environment observation that is effectively uncontrollable for the agent, e.g. a tree in the background.

[13] This distribution is fit to Scalor's representation on the initially collected dataset.

signal.[14] However, for SMORL, latent state space (a set of vectors) and goal space (a single vector) are not directly compatible to compute a distance. Instead, the goal object first has to be re-identified among the set of currently observed objects. To do so, the object closest to the goal in the *appearance component* of the representation is selected, assuming that two occurrences of the *same* object are close in this space. The reward is then defined as the distance between the selected object and the goal object solely in terms of the *positional component* of the representation, ensuring a meaningful reward signal.

Training With the goal space, goal sampling and reward function defined, SMORL can be optimized with any multi-goal, model-free RL algorithm. In particular, we chose goal-conditioned soft actor-critic (SAC) (Haarnoja et al., 2018) as a state-of-the-art (at the time) algorithm for continuous action spaces. During off-policy training, we also rely on *goal relabeling techniques* (Andrychowicz et al., 2017) to improve sample efficiency.[15]

4.2.3 SMORL Algorithm: Test Phase

At test time, the trained agent is given goal images showing the state of the environment to achieve. The SMORL agent then operates as follows:

(I) Decompose goal image into set of object vectors using SCALOR.
(II) Pick random unsolved sub-goal from set of goal vectors.
(III) Try to solve sub-goal using trained SMORL policy for some steps.
(IV) Go to (II), repeat until all sub-goals are solved or a timeout is reached.

As discussed above, this simple algorithm relies on the assumption that all sub-goals are actually independently achievable, i.e. that there are no dependencies between sub-goals.[16] Furthermore, while solving a sub-goal, the agent is ignorant to previously made progress — it may inadvertently destroy sub-goals solved beforehand.

4.3 Results

At the time this project was conducted, no suitable benchmark environments for multi-goal, multi-object visual RL existed. Thus, to test the SMORL agent, we created the VISUALREARRANGE environment[17] based on the MuJoCo simulator (Todorov et al., 2012), implementing a continuous control task: the agent has to control a 7 degrees-of-freedom robotic arm to move one or several "puck"-like objects to certain target locations on a table. The agent is only provided with a low-resolution RGB camera image of the scene (see Fig. 4.3).

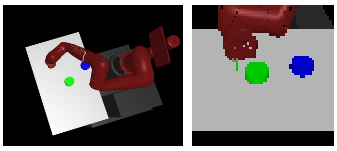


Figure 4.3: VISUALREARRANGE environment from the top (left) and agent observation (right). From Zadaianchuk et al. (2021).

The experiments tried to answer three questions:

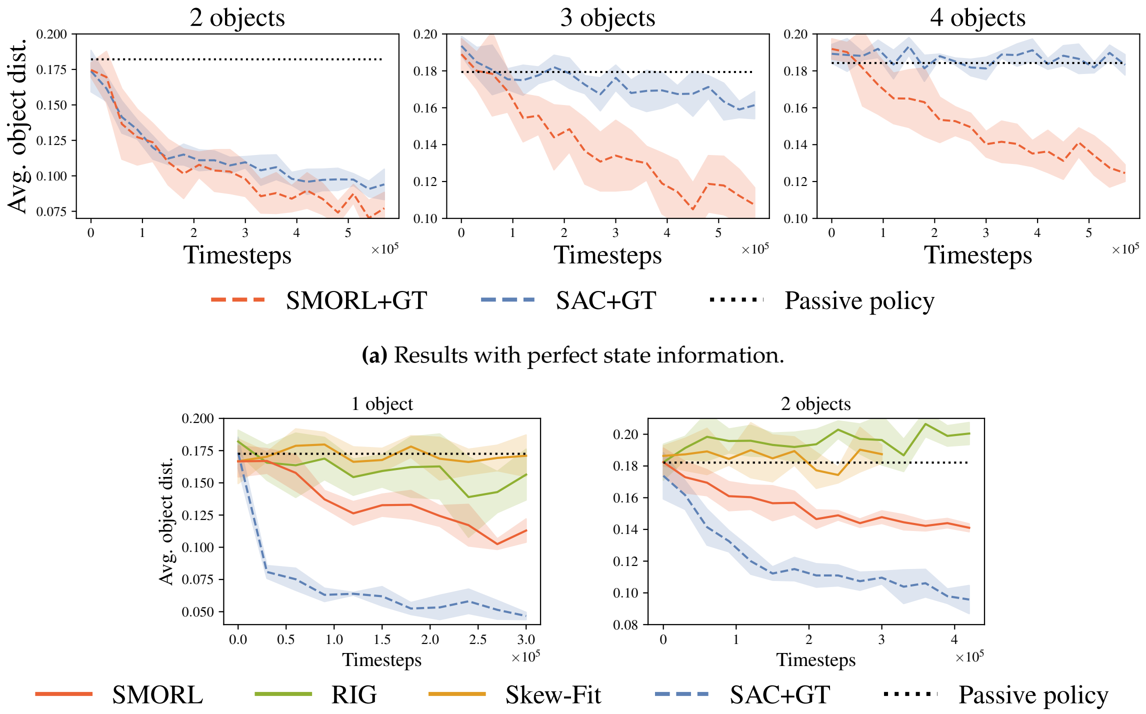
(I) How does the SMORL *algorithm* work in isolation (i.e. given perfect, object-factorized state information)?

[14] The actual usefulness of such a "latent distance reward" depends on the specific properties of the latent space. For example, A. Nair et al. (2018) found that the more disentangled latent space learned by a β-VAE (Higgins, Matthey, et al., 2017) works better than that of a regular autoencoder.

[15] Any recorded environment transition can be replayed with *counterfactual goals* for training; we utilize actually occurring *future states* (known as hindsight experience replay (Andrychowicz et al., 2017)) and *"imagined goals"* (A. Nair et al., 2018) (chosen according to the goal sampling procedure described above) for this purpose

[16] For examples of tasks with dependencies between sub-goals, consider stacking a tower out of blocks, or using a rod to catch a fish.

[17] We also tested SMORL on a simpler environment named VISUALPUSH, which we do not discuss here.



(a) Results with perfect state information.

(b) Results with image observations.

Figure 4.2: Selected results on the VisualRearrange environment. (a): with perfect state information, Smorl successfully rearranges up to 4 objects, with the non-compositional SAC agent not exceeding a "passive policy". **(b)**: with image observations, Smorl performs better than the image-based methods RIG and Skew-Fit, but does not reach the performance of its state-based counterpart. We show the average distance of objects to their goal position as mean and standard deviation over 5 random seeds. Figure from Zadaianchuk et al. (2021).

(II) How does the Smorl agent compare to previous multi-goal visual RL agents (i.e. given image observations)?

(III) Does the Smorl agent *generalize* to out-of-distribution settings?

Smorl with Perfect State Information The goal of this experiment was to show how an object-centric representation can benefit the training of an RL agent, compared to a standard flat vector-based representation. To isolate the contribution of the object-centric representation itself from the imperfections of learning it, we trained a Smorl agent that receives a set with the ground truth state of the objects as input. We compared it with a SAC agent that receives the full state as a single vector. When testing on the VisualRearrange environment with 2, 3, and 4 objects, we found that Smorl scales well with the number of objects, successfully rearranging up to 4 objects (see Fig. 4.2a). The SAC agent, in comparison, worked well with 2 objects, but struggled with 3 objects; with 4 objects, it did not exceed the performance of a passive policy that performs no movements. This shows that with the help of object-centric representations and algorithms utilizing compositionality, even complex multi-object environments become manageable.

Smorl with Image Observations Next, we combined the Smorl algorithm with Scalor representations learned from images, and compared it with two image-based multi-goal RL methods, RIG (A. Nair et al., 2018) and SkewFit (Pong et al., 2020), as well as with the state-based SAC agent. We found that Smorl performed at least as well as the image-based baselines, and was significantly better on the VisualRearrange environments (see Fig. 4.2b) — with 2 objects, the baselines did not perform better than the passive policy, whereas Smorl successfully solved the task. However, there was still a gap to the state-based SAC baseline, and no method was able to manage environments with more than 2 objects; this indicated significant challenges in learning a good representation for visual control.

OOD Generalization We also tested the generalization capabilities of the Smorl agent. In particular, we ran the agent trained on the 2 objects VisualRearrange environment in the 1 object environment. We found that the agent performed as well as a Smorl agent trained with 1 object from the start. While the change from 2 to 1 object may seem like a small change, it was found that the transfer abilities of visual RL agents at the time were susceptible even to minor perturbations in the environment (Higgins, Pal, et al., 2017).

4.4 Discussion

This work provided, to the best of my knowledge, the first successful demonstration of solving multi-object environments from images in the realm of self-supervised multi-goal RL. Previous work in this setting was limited to single-object environments, and, as was shown, does not scale to multiple objects — which I attribute to using unstructured representation & goal spaces. In contrast, the Smorl agent utilizes a structured object-centric representation, allowing it to break down compositional environments into independent entities to facilitate relational reasoning, goal decomposition and efficient exploration. Furthermore, Smorl shows how the access to structured representations enables the *design* of more effective and efficient algorithms, for instance in the goal sampling procedure or the reward function.

4.4.1 Limitations

Naturally, this work only constituted a first step towards general self-supervised RL agents that can learn to achieve goals in complex environments. Out of the limiting factors that I encountered while working on this project, two stand out in particular: (1) the assumption of independence of goals; and (2) flawed representation learning.

In all but the simplest environments, the assumption of independence of (sub-)goals is clearly not fulfilled: goals are directly dependent on each other when one goal depends on achieving another first (e.g. for a block stacking task), but also indirectly, because the *overall* goal depends on achieving all sub-goals jointly. For SMORL, the violation of this assumption manifested in the agent destroying previously achieved goals, as it was ignorant of their existence. A resolution to this problem was proposed in an extension to the SMORL algorithm named SRICS (Zadaianchuk et al., 2022): by estimating dependency graphs between goals, the agent attempts goals in the correct order; moreover, the agent is incentivized to keep previously solved goals intact by an additional reward signal.

The results showed a discrepancy between using perfect state information and the learned SCALOR representation. Indeed, in the experiments, we found that SCALOR's representation exhibited several problems, such as failing to detect objects, failing to track objects through occlusions, and noisy representations, leading to incorrect matching in the reward function. In particular, not handling occlusions makes the environment partially observable from the agent's point of view; this could be dealt with by utilizing recurrent policies. More broadly, I expect SMORL to benefit from general advances in object-centric representation learning to close the gap to perfect state information. Indeed, Haramati et al. (2024) recently showed that combining a SMORL-like architecture a more powerful object-centric representation scales goal-conditioned multi-object RL to up to 10 objects.

4.4.2 Perspective

From today's point of view, the environments and tasks used in this work were simplistic. What would be required to scale SMORL to more complex, real-world environments? For the purpose of this discussion, let us only consider the realm of tabula rasa-style reinforcement learning, that is, an agent is dropped into an unknown environment and has to learn *from scratch*. Here, two ingredients have shown promise: curriculum learning, and improved neural network architectures.

For self-supervised agents in compositional environments, the idea of learning by curriculum[18] appears natural: first learning to control the agent itself, then manipulating individual objects and finally achieving composite goals involving multiple objects. This form of *structured exploration* is more efficient than the uniform exploration strategy that SMORL applies, because the agents avoids learning tasks that either are already mastered or are currently too difficult. Hand-designed curricula have helped multi-goal RL agents to solve difficult multi-object manipulation tasks (R. Li et al., 2019). A curriculum can also be automatically derived, for example by monitoring learning progress (Blaes et al., 2019), maximizing novelty (Sancaktar et al., 2022), or asymmetric self-play (OpenAI et al., 2021). Notably, OpenAI et al. (2021) show how an automatic curriculum can lead to mastering the manipulation of a large number and diverse set of objects. However, all of these works assume access to the ground truth state; how to integrate these methods with image observations or object-centric representations is an open question.

[18] Curriculum learning is the concept of structuring and ordering the tasks of a learning system from simple to challenging to improve its performance.

Architecturally, from today's point of view, it appears natural to aim for a more unified design of the policy, using standard self-attention based Transformer blocks instead of Smorl's one layer of cross-attention between goal and objects. The motivation is two-fold. First, Transformers are a more expressive architecture, e.g. allowing the policy to take object interactions into account. Integrating relational reasoning at the policy level has been shown to enable goal-conditioned RL methods to generalize better (Mambelli et al., 2022; A. Zhou et al., 2022), e.g. to different numbers and combinations of objects than seen during training. Second, Transformers are (in principle) highly scalable and as such provide a basis for tackling complex, large-scale environments. However, it is unclear whether a Transformer-based policy can be optimized with the noisy training signal from the RL objective,[19] especially given the known optimization difficulties of Transformers (Huang et al., 2020).

Taking a step back, the RL tabula rasa setting I assumed in this chapter has unfortunately shown little promise for real-world robot learning so far. Instead, a currently more successful approach is to utilize (large) pre-trained models — primarily vision models providing robust representations (S. Nair et al., 2022; Xiao et al., 2022), but also language models (Ahn et al., 2022; Driess et al., 2023) — and combine them with imitation learning. To apply object-centric representations in the same manner would require methods that robustly work in the real world. In Part III of this thesis, I will introduce models that take a step towards that goal; in Sec. 8.2.4, we will then continue the discussion of the application of object-centric models to robotics.

[19] Integrating additional loss functions, e.g. based on future prediction (Schwarzer et al., 2021) or policy distillation (Bauer et al., 2023), could resolve instabilities; such auxiliary objectives have been shown to enable scaling of RL training (Schwarzer et al., 2023), but also to bootstrap large Transformer-based policies (Bauer et al., 2023).

Chapter 5

Causal Influence Detection for Improving Efficiency in Reinforcement Learning

This chapter continues the discussion of how the assumption of structure leads to better and more efficient agent learning algorithms. Whereas the previous chapter was concerned with how to learn a structured object representation and to integrate it effectively into self-supervised RL agents to achieve goals, this chapter focuses on a different structural aspect, namely the *relationships between entities*. In particular, I present a method for discovering the *causal influence of an agent on objects in the environment*. By interpreting the object structure as causal variables in a causal graph, we can formalize the notion of an agent's causal influence, derive a measure to quantify it, and develop a practical approach to learn to estimate it from data. We then discuss how this measure can be integrated into RL algorithms to inform exploration and learning, leading to strong improvements in data efficiency on robotic manipulation tasks.

This chapter summarizes

Maximilian Seitzer, Bernhard Schölkopf, and Georg Martius (2021). "Causal Influence Detection for Improving Efficiency in Reinforcement Learning". In: *Conference on Neural Information Processing Systems (NeurIPS)*

The full publication can be found in Appendix B.

Departing from the other chapters, in this chapter, we assume a structured object representation of the environment is given, and do not attempt to learn it from data. This lets us focus on the problem of learning causal relations. However, the presented method is agnostic to the underlying representation, and thus I expect it to be combinable with object-centric learning techniques, for example when only image observations are available.

In the context of this thesis, this chapter serves as a second case study to demonstrate the advantages of structured object representations. Fundamentally, structure is a prerequisite for causality; only by starting from structure does it become meaningful to talk about causal relations at all. As in the previous chapter, we use structure to design better RL algorithms: in this case, by informing the agent about its causal influence, we guide the agent's learning in a structured way, freeing it from the need of discovering the environment fully on its own. In essence, the structure acts as a useful inductive bias that encodes knowledge of the world into the agent.

5.1 Motivation

Let us revisit the example given in Sec. 4.1: a robot in front of a table with several objects, supposed to bring the objects into a particular arrangement. To simplify the problem, let us assume that the perception problem is solved, that is, the robot has perfect information about its own state, the state of the objects, and the intended goal. In all other respects, the agent starts with a blank slate, meaning it has to explore the environment and learn when and how its actions impact the objects. The difficulty of this problem stems from the fact that interactions between agent and objects are (initially) rare.

From a causal perspective, this observation can be explained by the fact that *the agent's causal influence over the environment is sparse*. This is a consequence of two basic assumptions about the causal structure of the world:

(I) The principle of *independent causal mechanisms* (Schölkopf, 2022): the world's generative process consists of autonomous modules, or independent entities.

(II) Entities[1] have "*limited interventional range*" (Seitzer et al., 2021): their potential influence over other entities is localized in space and occurs sparsely in time.

Causality is thus a useful framework to understand the interactions of agents with environments. In particular, it allows us to explain and formalize the situation-dependent nature of control (see Sec. 5.3) — the robot can only influence the object when both are close — the core motivation behind this work.

From a RL perspective, the situations where the robot can control the object are crucial: (1) initial control is rare, rendering training inefficient; (2) physical contacts are challenging to model, thus requiring greater effort to learn; and (3) states of control constitute "bottlenecks", states that must be traversed to achieve further goals. Consequently, agents should be made aware of such situations, both during learning and data collection. In Sec. 5.4, we will see three intuitive modifications to the RL algorithm that address this issue: exploring towards states of influence, selecting actions with causal influence, and prioritizing these states during training.

5.2 Background

In this section, we will connect structured object representations with causal modeling and Markov decision processes (MDPs), the basic framework underlying reinforcement learning. The main new aspects are the notions of action (from RL), and intervention (from causality), providing a fresh perspective on object-centric representations as well.

A *Markov decision process* is described by a tuple $\langle \mathcal{S}, \mathcal{A}, P, \rho_0, r, \gamma \rangle$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $P(S' \mid S, A)$ is the transition distribution over the next state $S' \in \mathcal{S}$[2] given the current

[1] Here, "entities" can be broadly interpreted as "physical processes", including actively acting agents.

[2] We (ab)use the notation $X \in \mathcal{X}$ to mean that the random variable X has support set $\mathcal{X}$, i.e. the set of values with non-zero probability — even though some mathematicians regard this notation as "*sloppy, [...] nonsensical, or [...] just plain wrong*" (Huber, 2015).

state $\mathbf{S} \in \mathcal{S}$ and a selected action $\mathbf{A} \in \mathcal{A}$; ρ_0, r, and γ are initial state distribution, reward function, and discount factor, which we will not be concerned with here — we refer to Sutton and Barto (2018) for a comprehensive overview. In our case, we assume the state space is *structured*, that is, it factorizes into N subspaces: $\mathcal{S} = \mathcal{S}_1 \times \mathcal{S}_2 \times \cdots \times \mathcal{S}_N$. Intuitively, we can regard the different subspaces $\mathcal{S}_j$ as describing different *entities* in the environment. Thus, the $\mathcal{S}_j$ directly map onto our formalization of slot-based object representations (see Chap. 3).

We can then define a *causal graphical model* (CGM) (Peters et al., 2017, Def. 6.32) for a single transition step in this MDP, thus containing the set of random variables $\mathcal{V} = \{\mathbf{S}_1, \ldots, \mathbf{S}_N, \mathbf{A}, \mathbf{S}'_1, \ldots, \mathbf{S}'_N\}$.[3] Formally, the CGM consists of a (directed) causal graph $\mathcal{G}$ with nodes $\mathcal{V}$ and a conditional distribution $P(\mathbf{V}_j \mid \text{Pa}(\mathbf{V}_j))$ for each node $\mathbf{V}_j \in \mathcal{V}$, where $\text{Pa}(\mathbf{V}_j)$ is the set of parents of $\mathbf{V}_j$ in the causal graph. The joint distribution $P_{\mathcal{V}}$ over all random variables $\mathcal{V}$ is then given by a density that *causally factorizes* as

$$p(v_1, \ldots, v_{|\mathcal{V}|}) = \prod_{j=1}^{|\mathcal{V}|} p(v_j \mid \text{Pa}(v_j)). \tag{5.1}$$

Intuitively, a CGM is *compositional*: it suffices to describe each variable locally in terms of its direct dependencies. Beyond a purely observational description, we can also express an *intervention* on the system, simply by replacing one of the factors with a new probability distribution and leaving the rest of the distributions unchanged. For example, the agent's interactions with the environment can be modeled as interventions on the action variable with the policy $\pi(\mathbf{A} \mid \mathbf{S})$. In this way, CGMs instantiate the principle of *independent causal mechanisms*, positing that the world is "*composed of autonomous modules that do not inform or influence each other*" (Peters et al., 2017). Note how this matches with our intuitive view of *objects* as being modular and context-independent — this gives some justification to interpreting structured object representations as a coarse-grained causal representation (see also Sec. 2.2.2).

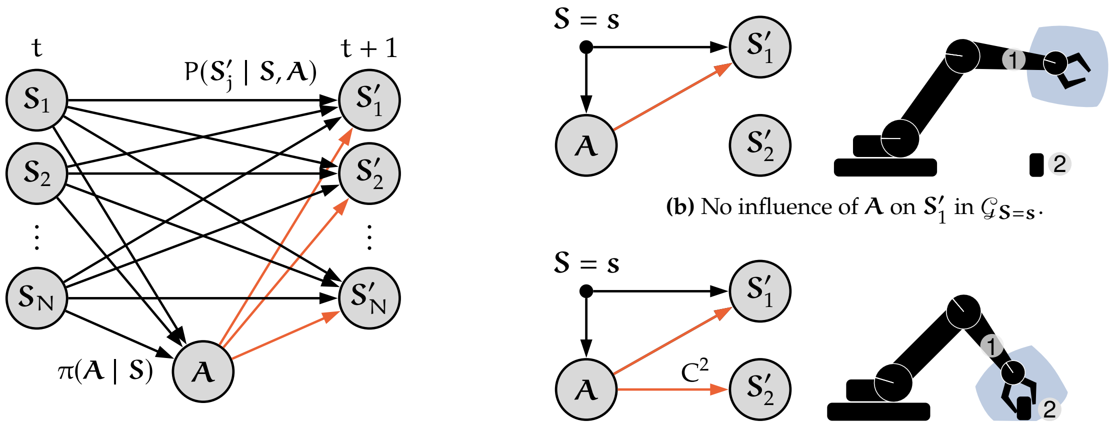
5.3 The Causal Influence of an Agent

The causal graph $\mathcal{G}$ is depicted in Fig. 5.1a; it encodes the assumption that there are no edges between nodes within a single time step (no instantaneous effects). However, between successive time steps, all nodes are connected. This is because, when viewed in aggregate, all entities could potentially influence each other at some point in time, making it necessary to include edges between those entities in the causal graph. Clearly, this global graph is not very useful when we are interested in the agent's influence on its environment.[4]

Instead, we now adopt a local perspective: typically, there are *no* interactions between entities, allowing us to sparsify the graph (by removing edges) when focusing on a particular situation. In particular, recalling the earlier discussion in Sec. 5.1 about the sparsity of influence,

[3] Here, a fully-observed setting is assumed, i.e. the CGM only contains *endogenous* variables.

[4] The aim of traditional causal discovery is to detect the edges of the (global) causal graph (Pearl, 2009). My work departs from this static setting by viewing the causal relations as *dynamic*, which appears more appropriate for physical environments.



(a) Global causal graph $\mathcal{G}$.

(b) No influence of A on S'_1 in $\mathcal{G}_{S=s}$.

(c) Influence of A on S'_1 and S'_2 in $\mathcal{G}_{S=s}$.

Figure 5.1: Causal graphical model for transitioning from state S to S' by action A. We assume the state factorizes into components $\mathcal{S}_1, \mathcal{S}_2, \dots, \mathcal{S}_N$, representing different entities in the environment. **(a)**: Viewed globally over all time steps, all entities and the agent's action can potentially influence the state of entities at the next time step. **(b)**: Given a concrete situation $S = s$, some influences may vanish in the *local causal graph* $\mathcal{G}_{S=s}$. **(c)**: We posit that the important states for an agent are those in which it has causal influence on the environment through its action (orange arrows). The proposed *causal action influence* C^j (Eq. (5.2)) measures the "strength" of those arrows. Figure adapted from Seitzer et al. (2021).

we observe that the agent's "*sphere of influence*" (visualized as blue areas in Figs. 5.1b and 5.1c) is limited: the agent's action only occasionally affects other entities (Fig. 5.1c), but most of the time, its control is confined to itself (Fig. 5.1b). Thus, for this work, we are interested in detecting the influence of the agent's action in any particular state configuration $S = s$, i.e. the existence of the orange arrows in Fig. 5.1.

What does it mean for the agent to have causal influence on an entity through its action? Intuitively, this is the case if the action *causes* the subsequent state of the entity to happen, or in other words, knowing the action is required to determine the outcome for the entity.[5] Note that this definition leads to counterintuitive assessments. The paper gives the example of a robot moving away from an object with its action; this action is still considered a cause for the subsequent position of the object, as a different action would have led to touching and changing the position of the object. This means that under this definition of causal influence, we cannot differentiate between influence of *different actions*, but only whether the agent has influence in a *particular situation*. In the following, we will formalize these ideas and derive an algorithm for measuring the agent's state-dependent causal influence on the environment.

5.3.1 Summary of Formal Results

Local Causal Graphical Model (Seitzer et al., 2021, Def. 1) We first clarify how CGMs behave locally after observing a particular situtation. To this end, we define the *local causal graphical model*, a CGM that is conditional on observing the variables $\mathcal{X} \subset \mathcal{V}$ taking specific values x.

[5] The underlying notion of causality is "sine qua non" causation, captured by the "but-for" test: "A *is a cause of* B *if*, but for A, B *would not have happened*." (Halpern, 2016, p. 3). This can be different from the way humans determine cause and effect, which relies on notions of normality: A *is a cause of* B *if* A *happens and* B *would not* normally *have happened anyway* (Kahneman and Miller, 1986; Halpern, 2016). Both variants invoke counterfactual reasoning, but the second is more difficult to compute, as it requires access to a "normal" version of the world without the agent's interference.

In particular, some of the edges in the associated *local causal graph* $\mathcal{G}_{X=x}$ may vanish (reflecting the lack of causal influence along that edge).

Control of an Agent Next, we characterize the agent having state-dependent causal influence as the agent *being in control* in that state.[6] Formally, we define the agent to be in control of some entity S'_j in state $S = s$ when there is an edge $A \to S'_j$ in the local causal graph $\mathcal{G}_{S=s}$ under all "sufficiently broad"[7] policy interventions (Seitzer et al., 2021, Sec. 4). Furthermore, this edge exists under some policy intervention if and only if $S'_j \not\perp\!\!\!\perp A \mid S = s$, that is, S'_j and A are conditionally dependent upon observing $S = s$ (Seitzer et al., 2021, Prop. 1).

Detecting Control from Interventions (Seitzer et al., 2021, Prop. 2) Finally, we derive conditions under which conclusions resulting from a particular policy intervention allow us to detect control in general. In particular, if we can find any policy intervention under which $S'_j \not\perp\!\!\!\perp A \mid S = s$, the agent is *in control of* S'_j *in* $S = s$. Furthermore, if we can find any "sufficiently broad" policy intervention under which $S'_j \perp\!\!\!\perp A \mid S = s$, the agent is *not in control of* S'_j *in* $S = s$.

5.3.2 Causal Action Influence

Equipped with these results, we now turn to the question of how to detect the control of an agent in practice. As we have seen, control is linked to the conditional dependence $S'_j \not\perp\!\!\!\perp A \mid S = s$. A well-known measure for the degree of dependence between two random variables given another is the *conditional mutual information* (CMI) (Cover and Thomas, 2006). Thus, to measure the agent's amount of control, we define the *causal action influence* (CAI) $C^j : \mathcal{S} \mapsto \mathbb{R}^+_0$ of the agent on entity S'_j in state $S = s$ as

$$C^j(s) := I(S'_j; A \mid S = s) = \mathop{\mathbb{E}}_{a \sim \pi}\left[D_{\mathrm{KL}}(P(S'_j \mid s, a) \parallel P(S'_j \mid s))\right], \quad (5.2)$$

where I denotes the CMI, and D_{KL} is the Kullback-Leibler (KL) divergence. CAI is computed with an expectation over a policy π; the previous results tell us that to detect control, it is sufficient to evaluate a single policy with full support.[8] In this case, CAI is zero exactly if the agent is not in control; thus we can threshold C^j to detect control.

The right-hand side of Eq. (5.2) provides some intuition about CAI. We can see that CAI measures the *average difference* (in terms of KL divergence) between the *outcome* $P(S'_j \mid s, a)$ *for a particular action* a sampled from the policy π, and the *transition marginal* $P(S'_j \mid s)$, i.e. the average distributional outcome over all actions. If all actions result in the same outcome, and this outcome is exactly the transition marginal, the divergence between them is zero, and S'_j and A are independent (in $S = s$).

Practical Implementation The CMI is intractable in general, and we need suitable simplifications to obtain a more practical estimator. We first make two Monte-Carlo approximations.

[6] Inspired by information-theoretic interpretations of control theory (Touchette and Lloyd, 2004).

[7] Policies with full support, i.e. $\pi(A = a \mid S = s) > 0 \; \forall a \in \mathcal{A}$.

[8] In practice, we use a uniform policy over the action space: $\pi(A \mid S) := \mathcal{U}(\mathcal{A})$.

(I) The *outer expectation* $\mathbb{E}_{a\sim\pi}$ is approximated via sampling of K actions:

$$\mathop{\mathbb{E}}_{a\sim\pi}[\dots] \approx \frac{1}{K}\sum_{i=1}^{K}[\dots], \quad \{a^{(1)},\dots,a^{(K)}\} \overset{\text{iid}}{\sim} \pi.$$

(II) The *transition marginal* is approximated with a (finite) mixture distribution:

$$p(s'_j \mid s) = \int p(s'_j \mid s, a)\,\pi(a \mid s)\;da \approx \frac{1}{K}\sum_{k=1}^{K} p(s'_j \mid s, a^{(k)}).$$

We then introduce parametric assumptions on the transition distribution $P(S'_j \mid s, a)$ in order to be able to compute the KL divergence:

(III) We assume *the next state is Gaussian-distributed* with a diagonal covariance matrix:

$$S'_j \mid s, a \sim \mathcal{N}\left(\mu(s,a), \sigma^2(s,a)\right).$$

(IV) The Gaussian distribution is *parametrized by a neural network* f_θ trained to predict the next entity state S'_j by maximizing the log likelihood on a dataset of N environment transitions $\{(s^{(i)}, a^{(i)}, s'^{(i)})\}_{i=1}^{N}$[9]:

$$\theta^* = \arg\max_\theta \sum_{i=1}^{N} \log p\left(s_j'^{(i)};\, \mu(s^{(i)}, a^{(i)}), \mathbf{I}\sigma^2(s^{(i)}, a^{(i)})\right),$$

where $(\mu, \sigma^2) = f_\theta(s, a)$ and $p(s';\, \mu, \mathbf{I}\sigma^2)$ is the multivariate normal density.

(V) The resulting KL divergence between a Gaussian and a mixture-of-Gaussians is *estimated with a closed-form approximation* (see Seitzer et al., 2021, App. A.4).

Together, (I–V) result in the CAI estimator (Seitzer et al., 2021, Eq. 4):

$$\hat{C}^j(s) = \underbrace{\frac{1}{K}\sum_{i=1}^{K}}_{\text{(I)}}\Big[\,\underbrace{D_{\text{KL}}}_{\text{(V)}}\big(\overbrace{p(s'_j \mid s, a^{(i)})}^{\text{(III, IV)}} \;\|\; \underbrace{\frac{1}{K}\sum_{k=1}^{K}\overbrace{p(s'_j \mid s, a^{(k)})}^{\text{(III, IV)}}}_{\text{(II)}}\big)\Big], \tag{5.3}$$

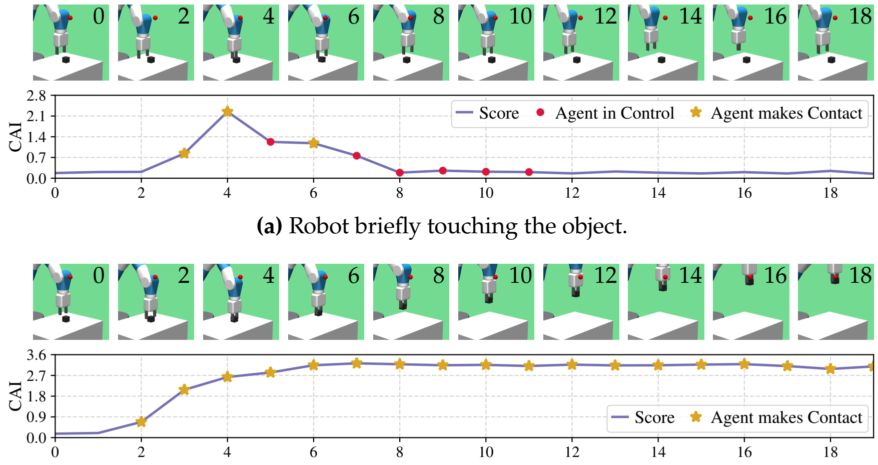
with $\{a^{(1)},\dots,a^{(K)}\} \overset{\text{iid}}{\sim} \pi$. The estimator $\hat{C}^j$ can be shown to be a lower bound of the true CMI becoming tighter with increasing sample size K (Poole et al., 2019), albeit only for the true density $p(s'_j \mid s, a)$. The accuracy also depends on the complexity of the action space; in our experiments, sampling a moderate number of actions (e.g. $K = 64$) for environments with 3-dimensional action spaces[10] was sufficient.

5.3.3 Results

In Fig. 5.2, we illustrate how CAI evolves on the FetchPickAndPlace robotic manipulation environment (Plappert et al., 2018). In Fig. 5.2a, it can be seen that influence peaks when the robot is close to the object and is close to zero when the robot cannot impact the object. In Fig. 5.2b, maximum influence is reached when the robot has lifted the object

[9] This is nothing other than training a forward, or dynamics model of the environment.

[10] For example, x, y, z-coordinates of a position-controlled robot.



(a) Robot briefly touching the object.

(b) Robot picking and lifting object.

Figure 5.2: Causal action influence visualized. We plot the CAI $\hat{C}^j(s_t)$, the causal influence of the agent's action on the object, in the FetchPickAndPlace environment. CAI spikes when interactions between robot and object occur, but is otherwise close to zero **(a)**; it is maximized when the robot holds the object in the air **(b)**. Figure adapted from Seitzer et al. (2021).

into the air. This matches our intuition: the highest degree of control is achieved in a state where the robot's action fully determines what happens to the object. In the paper, we also evaluated the estimator quantitatively by using it as a score for binary classification of control (see Seitzer et al., 2021, Sec. 5), and found CAI to perform effectively.

5.4 Faster Reinforcement Learning with Causal Influence

In Sec. 5.1, we examined several reasons why states of influence are important for an agent exploring and learning to interact with an environment. In Sec. 5.3, we saw that the causal action influence provides a theoretically founded and practical approach to classify such states. In this section, we will discuss three ways how CAI can be integrated into RL algorithms in order to inform an RL agent about its influence on the environment, and see how this can lead to dramatically increased sample efficiency.

5.4.1 Integration into RL Algorithms

Exploration Bonus The simplest way to steer an RL agent is through its reward signal. If we believe that states of influence are helpful for the agent, we can lead the agent to them, simply by using CAI as a reward signal, $r_{\text{CAI}}(s) := \hat{C}^j(s)$. This reward can be directly maximized by the agent as a form of *intrinsic motivation* (Schmidhuber, 1991, 2010), or in conjunction with an external task-specific reward, with r_{CAI} acting as an *exploration bonus*. In the former case, the agent is incentivized to attain control over its environment; in the latter case, seeking out states of influence provides stepping stones towards mastering the task, especially when the task reward is sparse.

Active Action Selection The exploration bonus operates retroactively, i.e. states of influence have to be visited once before they can be reinforced. An alternative is proactive exploration, where the agent *actively plans ahead* to take promising actions. For example, we can pick actions expected to have a large causal impact on the environment; this can be implemented by selecting the action with the highest contribution to the sum in the CAI estimator (Eq. (5.3)):

$$a^{\text{exp}} = \underset{a \in \{a^{(1)}, \dots, a^{(K)}\}}{\arg\max} D_{\text{KL}}\Big(p(s'_j \mid s, a) \,\|\, \frac{1}{K}\sum_{k=1}^{K} p(s'_j \mid s, a^{(k)})\Big), \qquad (5.4)$$

with $\{a^{(1)}, \dots, a^{(K)}\} \overset{\text{iid}}{\sim} \pi$. Intuitively, a^{exp} is the action leading to the highest deviation from the average expected outcome under the model. From a causal viewpoint, this can be interpreted as the agent *conducting experiments that verify its beliefs about the environment* by planning suitable interventions. Should the observed outcome differ from the expected outcome, the resulting data can be used to correct the underlying model in further training iterations.

Prioritized Learning Finally, we can use CAI to prioritize states of influence *in the training loop of the agent* — this integration is specific to off-policy RL algorithms making use of a replay memory.[11] In particular, when sampling states for training policy and value function from the memory, we assign each episode a sampling probability based on the total influence $\sum_t C^j(s_t)$ the agent had in this episode.[12] Intuitively, we expect such episodes to contain salient information for learning to interact with the environment; prioritizing them lets the agent bootstrap faster, especially in the early training phase where interactions are sparse.

5.4.2 Results

We evaluated the proposed influence-aware agents in multi-goal RL environments for robotic manipulation. All modifications were implemented on top of the DDPG[13] algorithm (Lillicrap et al., 2016) with hindsight experience replay (Andrychowicz et al., 2017) achieving state-of-the-art results in this setting at the time. Selected results are shown in Fig. 5.3.

First, we discuss using CAI as intrinsic motivation to explore the environment (Fig. 5.3a). It can be seen that the agent quickly learns to manipulate the object, moving it around 80% of the time after just 2 000 episodes of training, and reliably grasping, lifting and holding it in the air after 4 000 episodes (cf. Fig. 5.2b). This behavior is a natural consequence of maximizing control over the environment, and is well-suited to prepare an agent for further tasks it may encounter in the future.[14]

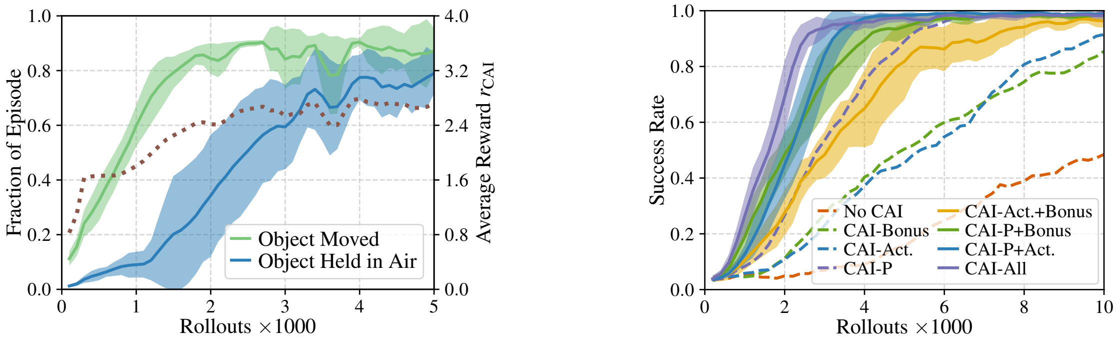
Second, we discuss integrating CAI with task rewards (Fig. 5.3b). All proposed schemes demonstrate significant increases in sample efficiency over the baseline (2× for CAI-BONUS and CAI-ACT, 4× for CAI-P). Furthermore, the improvements combine synergistically, with all schemes together (CAI-ALL) achieving a success rate of 95% in just

[11] A replay memory stores past environment interactions from earlier versions of the agent that can be reused for updating the current agent (Mnih et al., 2013).

[12] Our scheme can be seen as an informed variant of prioritized experience replay (Schaul et al., 2016): by making structural assumptions about the environment, we can use causal influence instead of the general, but less informative *TD error* to sample states.

[13] DDPG ≙ **d**eep **d**eterministic **p**olicy **g**radient.

[14] The concept of maximizing control to prepare for the future is also known as *empowerment* (Salge et al., 2014) — CAI can be seen as a tractable lower bound to one-step empowerment.



(a) CAI as intrinsic motivation.

(b) CAI for task-based learning.

Figure 5.3: Integrating causal action influence into RL agents. CAI can be used as a reward as a form of intrinsic motivation **(a)**, or in conjunction with task rewards **(b)**. In **(a)**, maximizing CAI leads to the agent quickly learning to move (green line) and lift the object into the air (blue line). In **(b)**, using CAI as an exploration bonus (CAI-BONUS), for active action selection (CAI-ACT), or for prioritized learning (CAI-P) all result in strong gains in sample efficiency over the baseline (No CAI), with all improvements combined (CAI-ALL) leading to a 10× speed-up. Both experiments are on the FetchPickAndPlace environment (Plappert et al., 2018), with the results showing mean and standard deviation over 10 random seeds. Figure adapted from Seitzer et al. (2021).

3 000 episodes, a gain in sample efficiency over the baseline of a factor of 10.[15] In the paper, we also benchmarked the improvements against other forms of *exploration bonuses* (maximizing information gain (Houthooft et al., 2016), *reducing model uncertainty* (Pathak et al., 2019)), *explorative action selection* (ϵ-greedy (Sutton and Barto, 2018, Sec. 5.4)), and *prioritized learning* (prioritized experience replay (Schaul et al., 2016), energy-based prioritization (Zhao and Tresp, 2018)), and found that our proposed approaches compare favorably on all tested environments.

5.5 Discussion

This chapter continued our exploration of the benefits of structure for autonomous agents. Specifically, we focused on a different structural property: the causal relationships between entities. Starting from a structured object representation, we reinterpreted the agent-environment framework of reinforcement learning as a causal model. A particularly intriguing aspect of that model is the agent's causal influence on entities in the environment, which we formalized and transformed into a practical measure, the causal action influence. We then discussed various approaches to equip RL agents with this measure to inform learning and exploration, and saw that this can lead to drastic improvements in sample efficiency. This illustrates how a structured representation mirroring certain properties of the world — the existence of entities, the sparsity of agentic influence — serves as a valuable inductive bias *to learn to act*.

[15] It takes around 30 000 episodes for DDPG+HER to solve this environment up to 95% success rate.

5.5.1 Related & Follow-Up Work

Identifying the edges of a causal graph is the problem of causal discovery (Pearl, 2009), traditionally understood as inferring the global causal dependencies of a system from a static dataset.[16] In Sec. 5.3, we have discussed how, in the context of RL, it is more fruitful to model local, situation-dependent causal relationships. As such, various studies have begun to focus on this area (Pitis et al., 2020, 2022; Hwang et al., 2024), some closely inspired by our work (Zizhao Wang et al., 2023; Tung et al., 2024; Urpí et al., 2024). This setting can be seen both as simpler and harder than general causal discovery: it is simpler because the direction of causality is known (progressing forward in time), resolving certain issues with identifiability (Eichler, 2012); it is harder because the local nature of the problem implies that an exponential number of data points could be required in the worst case.[17]

To *infer local causal influence*, several other approaches have been proposed. Pitis et al. (2020) train a Transformer model to predict the factorized next state $\mathbf{s}'$ and use the *total attention* on the factorized inputs $(\mathbf{s}, \mathbf{a})$ to determine the existence of the edges $S_i \to S'_j$ and $A \to S'_j$ in the causal graph. We demonstrated that this heuristic approach is ineffective for detecting action influence, a finding later confirmed by Urpí et al. (2024). In a similar manner, the *Jacobian matrix* of a trained forward model can be used; specifically, by considering the model's local partial derivatives as a measure of influence (Pitis et al., 2020; Zizhao Wang et al., 2023). Yet another method involves *predicting the local causal graph* from $(\mathbf{s}, \mathbf{a})$ by training a forward model with inputs appropriately masked by the predicted graph. This can be performed at the sample level (Hwang et al., 2023) (as in the attention and Jacobian approaches), or at a coarser level, assuming the state-action space can be partitioned into several regions with the same local causal graph (Hwang et al., 2024). Note that if one is interested in the *global* rather than *local* causal graph, general causal discovery methods are applicable; specifically in the time series setting, it is even feasible to simultaneously infer a latent causal representation and graph when assuming sparsity (Lachapelle et al., 2022), or data with targeted interventions (Lippe et al., 2022, 2023).

Since this work, causal influence has been incorporated into RL in various ways. To *enhance exploration*, in concurrent work, R. Zhao et al. (2021) suggested maximizing the mutual information between the agent's state and the surrounding environment state; similar to our formulation, this encourages control of the agent over the environment. Instead of maximizing control, Zizhao Wang et al. (2023) propose to explore by reducing the uncertainty about the local causal graph — this can be interpreted as learning to experiment to maximize information gain about the causal relationships.[18] Another notable application of causal influence is *data augmentation*: new counterfactual experiences to train the agent can be generated by combining components estimated to be locally independent (Pitis et al., 2020, 2022; Urpí et al., 2024). Finally, the concept of causal influence can also be applied to learn robust and generalizing

[16] See Vowels et al. (2022) for an overview of modern causal discovery methods.

[17] Consider that the transition distribution $P(S'_j \mid \mathbf{s}, \mathbf{a})$ underlying CAI can in principle completely change for each pair $(\mathbf{s}, \mathbf{a})$. Furthermore, strictly speaking, each data point $(\mathbf{s}, \mathbf{a})$ is only observed once in continuous spaces. The problem of finding local dependencies is thus intractable in general, requiring assumptions of smoothness of the modeled distributions.

[18] In causal discovery, this approach is known as (Bayesian) experimental design (Lindley, 1956; Agrawal et al., 2019; Tigas et al., 2022).

factorized dynamics models (e.g. for model-based RL), e.g. by training on counterfactually augmented data (W. Ding et al., 2023), independence testing (W. Ding et al., 2022), estimating mutual information (Zizhao Wang et al., 2022), or end-to-end with sparsity regularization (Hwang et al., 2023, 2024).

5.5.2 Limitations & Outlook

In this work, we assumed a fully-observed causal setting. An open question is whether unobserved factors could introduce *confounding*, that is, misattributing an entity's influence to the agent or vice versa. Although this issue was not addressed in this work, I believe that CAI is robust to such confounding because unobserved effects would be accounted for as aleatoric uncertainty, which is integrated out by the KL divergence in Eq. (5.3). Additionally, relaxing the assumption of no instantaneous effects could be beneficial; to this end, approaches such as the one proposed by Lippe et al. (2023) may be relevant.

Rather than estimating the entire causal graph, the work's focus was on the agent's influence on the environment through its actions, as we considered it to be the most relevant for policy learning. Furthermore, it is more tractable since the action distribution is directly controlled by the agent. An interesting extension would be to adapt the CAI estimator to detect influence between entities, similar to Pitis et al. (2020) and Zizhao Wang et al. (2023). This could enable the agent to discover interactions involving complex dependencies, such as using tools to achieve goals. Entity-entity interactions are also crucial for estimating *multi-step influences*, beyond the one-step influence examined in this work. This extension would allow tracing the effect of an agent's action through time, potentially leading to a principled approach for solving the *credit assignment problem*.[19]

The primary limitation of the CAI influence estimator is its reliance on an *accurate model*. Essentially, the problem of detecting influence is shifted onto the model — if the model fails to correctly detect causality, erroneously attributing (or not attributing) influence to the agent, we end up with a "circular dependency". A potential solution could involve iterative improvement of both the model and the influence estimator, possibly making use of active data gathering (Zizhao Wang et al., 2023) to correct model errors.[20]

Finally, we return to the central theme of this thesis: *structured object representations*. In this work, we assumed a known factorization of the state space into entities; in fact, this assumption is common to all aforementioned methods working with causal influence.[21] While this simplification allowed us to focus on the problem of detecting causal influence, it is clearly an unrealistic assumption for agents intended to be deployed in the *real world*. In such cases, the agent would have a perceptual component providing a latent representation of the world. Fortunately, the CAI framework is compatible with such representations, provided they meet the requisite assumptions, such as the independence

[19] How to assign "credit" for success among the many actions that may have been involved in producing it (Minsky, 1961; Sutton and Barto, 2018) — in other words, finding the actions that *caused* an outcome.

[20] Indeed, in the RL experiments in this work, the model was also updated while the agent gathered new data; however, the agent did not actively gather data to improve the model.

[21] Except those supporting causal representation learning: Lachapelle et al. (2022) and Lippe et al. (2022, 2023).

of entities. A natural candidate for such a representation is one that conceptualizes the world in terms of objects — a structured object representation. Thus, in the next part of this thesis, we will explore methods for learning real-world structured object representations, enabling the significant benefits for autonomous agents that became evident in Part II of this thesis.

Part III

On the Real-World Discovery of Structured Object Representations

Chapter 6

Bridging the Gap to Real-World Object-Centric Learning

In the last part of this thesis, we discussed how access to structured object representation brings large benefits for autonomous agents. However, when surveying methods for object-centric representations at the time, it became clear that they were limited to synthetic scenes with simplistic structure. This motivated me to study how we can discover structured object representations on real-world scenes. This is the topic of Part III of this thesis.

In this chapter, we introduce a method, Dinosaur,[1] that is able to learn object-centric representations from complex, natural images. By combining Slot Attention (Locatello et al., 2020) with pre-trained features from modern self-supervised learning methods such as DINO (Caron et al., 2021) or MAE (He et al., 2022), Dinosaur circumvents the scalability issues of prior methods. Dinosaur is the first object-centric model that scales to unconstrained real-world image datasets such as PASCAL VOC (Everingham et al., 2010) or COCO (T. Lin et al., 2014).

This chapter summarizes

Maximilian Seitzer, Max Horn, Andrii Zadaianchuk, Dominik Zietlow, Tianjun Xiao, Carl-Johann Simon-Gabriel, Tong He, Zheng Zhang, Bernhard Schölkopf, Thomas Brox, and Francesco Locatello (2023). "Bridging the Gap to Real-World Object-Centric Learning". In: *International Conference on Learning Representations (ICLR)*

The full publication can be found in Appendix C.

6.1 Motivation

Fields like computer vision and self-supervised representation learning (Balestriero et al., 2023) have long established real-world data as their playing field. In contrast, object-centric representation learning has lagged behind in that regard, mostly being confined to simplistic synthetic datasets. This is unfortunate, as the advantages object-centric representations promise — compositionality, generalization, robustness, interpretability — are significant. However, at some point, these qualities also must be demonstrated in real applications for the field to achieve wider significance. Thus, the motivation behind this work was to "*bridge the gap to the real world*", in order to (1) demonstrate that object-centric learning methods are real-world capable; (2) encourage the object-centric learning community to move beyond synthetic datasets; and (3) prepare the ground for real-world applications of object-centric representations.

[1] Dinosaur $\hat{=}$ **DINO** and **S**lot **A**ttention **U**sing **R**eal-world data.

To better understand the context behind this work, we briefly summarize the state of the object-centric learning field at the time this project was conceived of (early 2022). While there was interest in scaling object-centric representation learning methods to more complex data, real-world data was generally out of reach. Even on complex synthetic datasets such as the MOVi datasets (Greff et al., 2022), we demonstrated that state-of-the-art methods (Locatello et al., 2020; Singh, Deng, et al., 2022) struggled to discover objects. A trend at the time was to forego unsupervised learning by integrating *auxiliary sources of information* such as optical flow (Kipf et al., 2022), motion masks (Bao et al., 2022; Tangemann et al., 2023), depth maps (Elsayed et al., 2022), conditioning with object locations & shape (e.g. bounding boxes; Kipf et al., 2022), or textual scene descriptions (Jiarui Xu et al., 2022). But even including this kind of weak supervision, the datasets these methods were able to model were restricted to domains with limited variety (e.g. autonomous driving).[2] In contrast, this work showed that it is not only possible to scale object-centric learning to unconstrained real-world datasets, it is possible to do so fully unsupervised — going against the general trend toward integrating supervision.[3]

6.2 Real-World Object-Centric Representations with DINOSAUR

6.2.1 What Prevented Scaling to the Real-World?

A Lack of Model Scale? In line with the teachings of deep learning, a natural hypothesis is that the used models lacked sufficient scale to capture the complexity and diversity of natural data. Indeed, the employed neural networks are small by today's standards (roughly in the 1–5 million parameter range). In contrast, the weakly-supervised methods SAVi (Kipf et al., 2022) and SAVi++ (Elsayed et al., 2022) reported some success using larger residual networks (He et al., 2015) for the encoder. However, in our experiments with the (unsupervised) Slot Attention model, we found that replacing the encoder with residual networks or vision transformers (Dosovitskiy et al., 2021) did not result in successful object discovery on the COCO dataset (see Fig. 6.1). While from this, we could not rule out that lack of scale is part of the problem, purely increasing the model size was not sufficient.

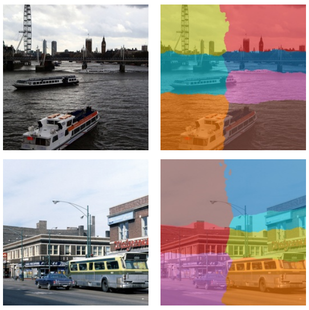


Figure 6.1: Slot Attention with a ResNet encoder fails to discover objects on COCO images (left); the learned slots (right) separate the image into regular patterns. From Seitzer et al. (2023).

Image Reconstruction as the Culprit An alternative hypothesis is that the objective of image reconstruction is to blame for failing to discover objects on real-world data. To understand why this could be the case, we need to understand the learning process towards discovering objects. To learn to group image features to objects, the features need to identify objects; to identify objects, the features need to be updated in a way that increases intra-object similarity and decreases inter-object similarity. The learning process is trying to increase the usefulness of the learned

[2] With the exception of GroupViT (Jiarui Xu et al., 2022), which used strong textual supervision to scale to broader datasets such as PASCAL VOC.

[3] Although there is no objection to the use of supervision if it is available in the target domain, maintaining a strictly unsupervised approach is still desirable: it broadens the applicability of a method by enabling training or fine-tuning on arbitrary data. Furthermore, being unsupervised enables scaling to large-scale unlabeled datasets, a principle that underpins modern foundation models in language (Brown et al., 2020) and vision (Oquab et al., 2023).

slots for the task (i.e. image reconstruction). If the task does not benefit from "object slots", there is no signal towards learning them, and an alternative grouping strategy will emerge.

The usefulness of the slots is determined by several factors, including the bottleneck capacity, the biases of the decoder, and the targets. On image datasets with a few simple, mono-colored, geometric objects, a local grouping by color is often enough to segment objects. When using such images as targets, only a few bits of information need to be captured in the slot for it to be useful; moreover, drawing a simple shape of uniform color at some position is easily learned by the decoder. The "chain" to learn a useful object slot is short. We can contrast this with trying to reconstruct *natural images* with a large variability of object types, shapes, and appearances. Learning a low-dimensional slot representation (and decoder) that allows for accurate reconstruction of such objects is difficult; it is simpler to learn slots that capture only the surface statistics of pixels.

We can find some evidence for this theory in models that use *alternative targets* to RGB images (discussed also in Sec. 3.2). When predicting targets that reveal objects in some form, for example optical flow or depth maps, object discovery often succeeds on more complex data (Elsayed et al., 2022; Kipf et al., 2022; Bao et al., 2023; Traub et al., 2024). For example, in the SAVi model, when switching from optical flow to RGB targets (keeping the input the same), performance drops catastrophically on the MOVi++ dataset; on the simpler MOVi dataset, this intervention has almost no effect (Kipf et al., 2022, Figure 3a).

To summarize, our hypothesis as to why previous object-centric methods failed to scale to real-world data is that pixel-level image reconstruction provides insufficient learning signal towards discovering objects. In particular, this task can be optimized by focusing solely on surface-level image statistics — if these statistics also happen to identify the objects, object discovery succeeds. This perspective suggests that previous methods relying on image reconstruction *only succeeded because the used datasets were simple*.

6.2.2 Self-Supervised Representations as Targets

If image reconstruction is the culprit for failing to scale to real-world data, it appears sensible to optimize an *alternative task*. In particular, if the issue is that image targets can be predicted by learning low-level image statistics, we should instead use targets that require learning high-level semantic information to predict them.[4] We would like such targets to be (1) available without supervision; (2) dense, i.e. each sub-region of the image has a corresponding sub-target; and (3) semantic, i.e. each sub-target is a high-level descriptor of the image at that position.

Candidates fulfilling all those criteria are the representations learned by modern *self-supervised learning methods* (Balestriero et al., 2023). Based on principles such as contrastive learning (He et al., 2019; T. Chen et al., 2020; X. Chen et al., 2021), self-distillation (Grill et al., 2020; Caron et al., 2021), clustering (Caron et al., 2020, 2021; Assran et al., 2022),

[4] The SLATE model (Singh, Deng, et al., 2022) provides support for this idea: by learning to predict discrete VQ-VAE tokens (Aäron van den Oord et al., 2017) describing image patches, it manages to handle more complex data than Slot Attention. However, SLATE still fails to scale to real-world data.

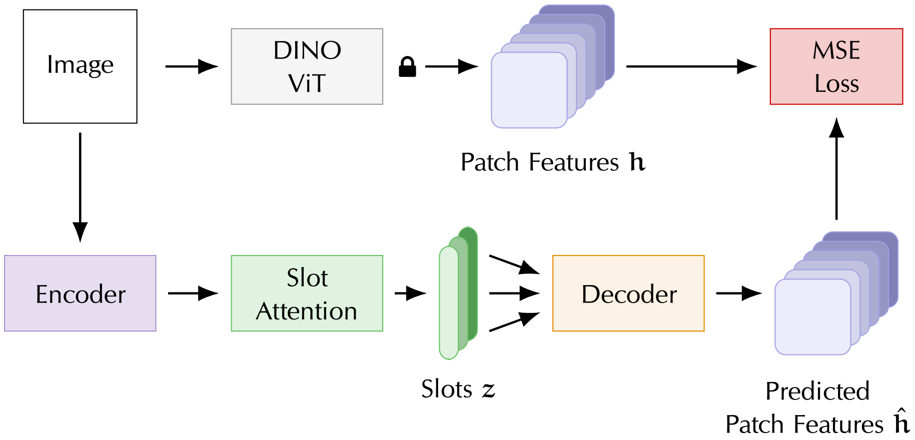


Image
DINO ViT
Patch Features $\mathbf{h}$
MSE Loss
Encoder
Slot Attention
Slots $\mathbf{z}$
Decoder
Predicted Patch Features $\hat{\mathbf{h}}$

Figure 6.2: The Dinosaur model. Figure adapted from Seitzer et al. (2023).

or masked reconstruction (Assran et al., 2022; He et al., 2022), these methods learn flexible, powerful image representations that have been used for a variety of vision tasks such as classification, object detection, image retrieval and more. When paired with vision transformers, it has also been observed for multiple of these methods that the attention maps focus on objects (Caron et al., 2021). Thus, in the Dinosaur model, we propose to use self-supervised representations as the prediction targets instead of images.

6.2.3 The DINOSAUR Model

The Dinosaur model is depicted in Fig. 6.2. At a high level, Dinosaur adapts the Slot Attention model (see Sec. 3.3.1) by substituting image reconstruction with *feature prediction*. Thus, the model follows the usual encoder-decoder structure: first, image features from an encoder are grouped into slots $\mathbf{z}$ by the Slot Attention module. Second, a decoder produces a combined prediction from the slots, in this case a dense feature map $\mathbf{y}$ consisting of a vector for each patch of the image. In a separate step, the targets are computed from the image, namely the patch features $\mathbf{h}$ of a pre-trained ViT (e.g. with the eponymous DINO method (Caron et al., 2021)). The model is trained with a mean-squared-error loss, $\mathcal{L}^{\text{mse}} = \|\mathbf{h} - \hat{\mathbf{h}}\|^2$. I refer the reader to the full publication (Appendix C) for more details about the different modules; some particular design choices are discussed in the next section.

6.3 Results

Comparison to Object-Centric Methods We evaluated Dinosaur in terms of object discovery on challenging synthetic (MOVi-C, MOVi-E) and real-world datasets (PASCAL VOC, COCO), with Dinosaur setting the new state-of-the-art at the time on all of them (see Fig. 6.3 for a representative subset of the results). We compared the model with two previous image-based object-centric models, Slot Attention (Locatello

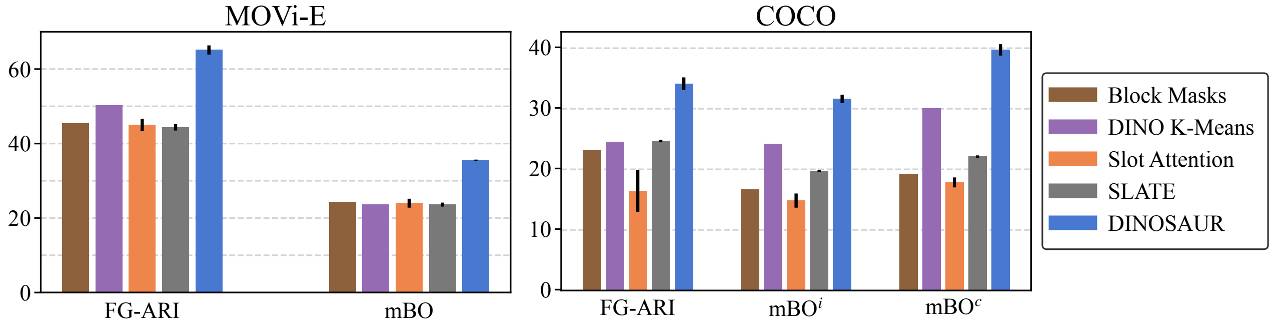


Figure 6.3: Evaluating unsupervised object discovery. On both synthetic (MOVi-E, left) and real-world (COCO, right) data, Dinosaur performs significantly better than the previous object-centric models Slot Attention and SLATE, as well as k-means clustering on DINO features. The previous methods do not perform much better than the naive block mask baseline, indicating that they do not properly discover objects. The plots show mean and standard dev. over 5 seeds for the FG-ARI and mBO metrics (higher is better; see Sec. 3.4.1 for an explanation). Figure adapted from Seitzer et al. (2023).

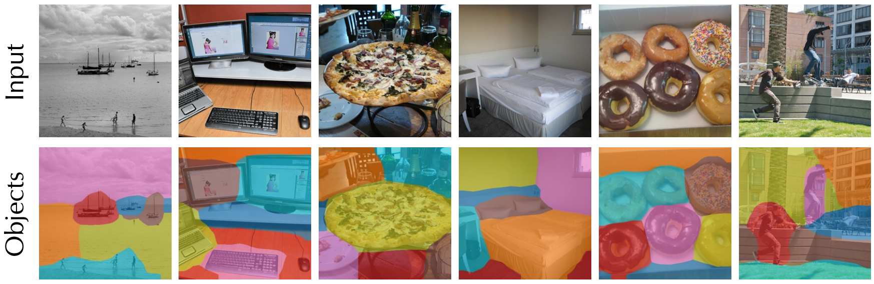


Figure 6.4: Examples of discovered objects on the COCO dataset. From Seitzer et al. (2023).

et al., 2020) and SLATE (Singh, Deng, et al., 2022). Whereas Slot Attention and SLATE struggle even on the synthetic datasets, performing no better than the naive "block masks" baseline, Dinosaur works well on both. On real-world data, Slot Attention fails completely, and SLATE is only better than the naive baseline. In contrast, Dinosaur successfully discovers objects of various types and appearances (see Fig. 6.4).

Comparison to Computer Vision Methods We also compared Dinosaur on related tasks popular in the computer vision community, namely unsupervised object localization (marking objects with bounding boxes) and unsupervised semantic segmentation (assigning a class label to each pixel in the image). This allowed us to compare to various strong baselines from the computer vision literature (e.g. Hamilton et al., 2022; Yangtao Wang et al., 2022; Zadaianchuk, Kleindessner, et al., 2023). Overall, Dinosaur performed competitively, despite being considerably simpler than the baselines often consisting of intricate pipelines with several stages of training.

Analysis of Components Finally, let us discuss some design choices for the different components:

- **Encoder:** randomly initialized residual network, or pre-trained, fixed ViT.[5] Both work similarly, but using the pre-trained ViT trains faster;

[5] In this work, we were not able to train a ViT encoder from scratch because of optimization difficulties; in subsequent work (Didolkar, Zadaianchuk, et al., 2024), we found improved optimization strategies alleviating this issue.

keeping it fixed has the computational advantage of reusing the network producing the targets.

- **Decoder:** MLP or Transformer. The MLP decoder is a variation of a spatial broadcast decoder (Watters et al., 2019) and predicts the target feature independently for each slot and position; the Transformer decoder (Singh, Deng, et al., 2022) auto-regressively predicts the target feature map while attending to the set of slots. We found that the Transformer decoder is biased toward semantic segmentation; the MLP decoder separates instances better but produces less accurate masks.
- **Target Representations:** we evaluated
  - *self-supervised representations* from DINO (Caron et al., 2021), MoCo-v3 (X. Chen et al., 2021), MSN (Assran et al., 2022), and MAE (He et al., 2022). Interestingly, all produce results of similar quality.
  - *supervised representations* resulting from classification on ImageNet, performing clearly worse than self-supervised representations.
  - *different network architectures*, residual networks or ViTs, with the features from the latter clearly performing better. This could be because ViTs localize information better than residual networks, which have large receptive fields.

6.4 Discussion

Dinosaur was the first object-centric representation learning method that successfully discovered objects in unconstrained real-world datasets such as PASCAL VOC and COCO. By utilizing strong pre-trained self-supervised representations, it demonstrated that object-centric methods *can scale to real-world data*. Importantly, this was achieved in a fully unsupervised manner, which allows training the model on large image collections as well as a straightforward integration with other tasks and modalities. A further strength of the method is its conceptual simplicity — the proposed loss function is compatible with any object-centric method that models images and can thus be combined with future methodological advances on the modeling side.

6.4.1 Impact

Dinosaur introduced the principle of using pre-trained self-supervised representations for object-centric learning. Various subsequent works successfully adopted this idea, firmly establishing it as a general approach for scaling to complex data. On the modeling side, Chakravarthy et al. (2023) integrate a locality prior into Dinosaur; Y.-F. Wu et al. (2023) demonstrate that inverted transformer layers are an alternative to Slot Attention in Dinosaur; Fan et al. (2024) propose a method for dynamically adapting the number of slots per image; Kakogeorgiou et al. (2024) improve the transformer decoder in Dinosaur, and Didolkar, Goyal, et al. (2024) propose a cycle-consistency objective to align features with slots. In our own follow-up work, VideoSAUR (Zadaianchuk, Seitzer, et al., 2023), we extended Dinosaur to *real-world videos*; this method will be discussed in the next chapter. Other adoptions to video data are proposed by Aydemir et al. (2023) and Fan et al. (2023). In all these works, the prediction of pre-trained features is used as the basis for real-world scaling, showing the robustness of the proposed mechanism to work with different kinds of models.

While our main proposal was to use self-supervised representations as *targets*, Dinosaur also utilizes them as *inputs* to Slot Attention. This has sometimes produced confusion over what the main factor for scaling to real-world data is. In the paper, we show that pre-trained representations on the target side are *sufficient*, and that they are *not necessary* on the input side; conversely, combining pre-trained inputs with image reconstruction fails to scale (Seitzer et al., 2023, Sec. 4.3). Interestingly, follow-up work has found that pre-trained inputs can also be *sufficient* when coupled with a suitable objective function or architecture. For example, Jiang et al. (2023) and Wu, Hu, et al. (2023) show that diffusion decoding conditioned on slots grouped from DINO features scales well to more complex data. Other options are training with a cycle-consistency objective to match slots with clusters of features (Ziyu Wang et al., 2023) or utilizing rotating features instead of Slot Attention for binding the pre-trained features to objects (Löwe et al., 2023). However, for all these methods, it is unclear how *instance-based* the learned decomposition is; results seem to indicate a *semantic* grouping instead.

Dinosaur has also been successfully used in several *downstream applications*. For tracking objects in real-world videos, Z. Zhao et al. (2023) integrate the slots learned by Dinosaur with a memory module. For the task of referring image segmentation,[6] Kim, Kim, Lan, et al. (2023) augment Dinosaur with a CLIP-style contrastive loss[7] (Radford et al., 2021) to infuse the learned slots with textual information; the resulting model can match slots with queries like "*man standing against the counter on the right*". Relatedly, Fan et al. (2023) align slots with pre-trained CLIP representations to obtain semantic video segmentations. Mamaghan et al. (2024) conduct a large comparison study of different types of representations for visual question answering (VQA) and find that for real-world data, the slots from Dinosaur perform best, even outperforming representations from DINOv2 (Oquab et al., 2023). For video-VQA, Jiaqi Xu et al. (2024) propose SlotVLM: in this work, Dinosaur's slots form a compact *video representation* used as inputs to an LLM[8]; the slots are then fine-tuned end-to-end to be useful for question answering.

6.4.2 Limitations

The complexity and ambiguities inherent in real-world data have revealed several shortcomings of slot-based methods. These limitations, mostly obscured in previous work due to their focus on synthetic data, are not unique to Dinosaur but general to slot-based methods. For instance,

[6] Identifying parts of an image matching a textual expression.

[7] Identifying the matching pair of vision and text representations among a contrast set.

[8] Vicuna-7B (Chiang et al., 2023), i.e. with 7 billion parameters.

in synthetic datasets, the maximum number of objects is known and can be used to set the "number-of-slots" parameter. Conversely, in real-world data, the number of objects per image can be arbitrarily high; simply using a large number of slots would result in significant oversegmentation. Additionally, previous models demonstrated the ability to "disable" unneeded slots (Locatello et al., 2020); however, on real data, the full set of slots is typically utilized as the model operates in an underfitting regime due to the slot bottleneck.[9] Finally, real-world data usually admits several valid decompositions, e.g. on different levels of hierarchy. Aside from the number of slots, current object-centric models offer no means of controlling the decomposition. This also poses an evaluation challenge as the ground truth labels objects in a particular manner; if the model chooses a different, yet equally valid set of objects, it is (erroneously) penalized.

A limitation specific to the Dinosaur model is the use of pre-trained representations as targets, which biases the object decomposition in some manner. In contrast, image reconstruction is unbiased as it forces the model to consider all information from the original image. It remains unclear how much the target bias limits the discovery of objects. Another limitation for certain applications could be the low resolution of the object masks; this stems from the coarse-grained patch resolution of the ViT backbone (e.g. 16 × 16 patches for images with 224 × 224 pixels). In subsequent work, we demonstrated that this issue can be alleviated with a brief fine-tuning phase on higher-resolution images (Didolkar, Zadaianchuk, et al., 2024). Last, for evaluation, we concentrated on mask-based evaluations, and did not investigate the content of the slot representations in detail[10]; nor did we evaluate downstream applications of the learned representation. For a comprehensive understanding of the method, these aspects should be explored in future work — though the afore mentioned successful applications of Dinosaur partially address this already.

6.4.3 Outlook

Dinosaur constituted a significant step forward in terms of the data complexity object-centric models can manage. By providing an "existence proof" of *real-world object-centric representation learning*, we hoped to encourage the community to study this challenging but exciting setting more. While we have seen rising interest in this area, we believe many natural applications of object-centric representations are still underexplored, for example in real-world robotics. Moreover, on the modeling side, there are still substantial open challenges due to the complexities of natural data, some of which we have discussed above. We will further analyze the implications of real-world object-centric representation learning in Sec. 8.2.

[9] Predicting pre-trained features instead of images likely exacerbates this issue; using more powerful decoders or a form of slot complexity regularization could alleviate it.

[10] Besides object property prediction on COCO, see Seitzer et al. (2023), App. B.3.

Chapter 7

Object-Centric Learning for Real-World Videos by Predicting Temporal Feature Similarities

This chapter introduces a method capable of learning object-centric representations from real-world videos. This method, VideoSAUR,[1] combines the SAVi[2] architecture (Kipf et al., 2022) with the Dinosaur model introduced in Chap. 6. Like Dinosaur, VideoSAUR utilizes pre-trained self-supervised features to scale to more complex data. Specifically, we proposed a novel feature similarity loss that encodes temporal and semantic correlations between video frames. This loss elegantly exploits the temporal information available in videos and biases the model toward discovering moving objects. We demonstrated that the model learns, fully unsupervised, to discover and track multiple objects in videos from the YouTube-VIS dataset (L. Yang et al., 2021).

This chapter summarizes

Andrii Zadaianchuk*, **Maximilian Seitzer***, and Georg Martius (2023). "Object-Centric Learning for Real-World Videos by Predicting Temporal Feature Similarities". In: *Conference on Neural Information Processing Systems (NeurIPS)*. *equal contribution

The full publication can be found in Appendix D.

7.1 Motivation

What makes videos interesting for object-centric learning? One reason is the immense amount of videos available as a resource for learning. However, this alone is not sufficient motivation; after all, a video can just as well be treated as a set of independent images. What makes videos much richer than static images is the contained *temporal structure*, providing the opportunity to learn about dynamics, cause and effect, and even 3D structure by inferring depth from observer motion through motion parallax. Specifically interesting for object discovery, consistent patterns of motion tend to reveal objects (the principle of coherent motion, see Sec. 2.4.2). Furthermore, in many contexts in which we would like to deploy object-centric models (e.g. robotics), integrating temporal consistency and motion properties (e.g. velocity) into the representation is crucial. All in all, this makes object-centric video models an important topic to study.

[1] VideoSAUR ≙ **Video** **S**lot **A**ttention **U**sing temporal feature simila**R**ity.

[2] SAVi ≙ **S**lot **A**ttention for **Vi**deo.

At the time this work was done, object-centric video models could not be applied to unconstrained real-world data. This resembled the situation of image-based models before Dinosaur (discussed in Sec. 6.1). However, for video models, some real world demonstrations existed by integrating supervision through motion (Bao et al., 2022) or depth maps (Elsayed et al., 2022). Nevertheless, these models were only successfully applied to datasets of a single domain with limited complexity, namely autonomous driving. As such, there was a need to scale object-centric models to real-world videos.

The prevalent approach to modeling videos with object-centric representations is to treat them as a sequential process, with the object representations dynamically evolving through time (Kosiorek et al., 2018; Jiang et al., 2020; Weis et al., 2021; Kipf et al., 2022; Traub et al., 2023, see also Sec. 3.1.4). In Slot Attention-based architectures, this is commonly implemented by initializing slot discovery for the current frame with the slots of the previous frame (Bao et al., 2022; Elsayed et al., 2022; Kipf et al., 2022; Singh, Wu, et al., 2022). While this recurrence creates a bias towards slot consistency, i.e. maintaining a stable object identity per slot, it does not add a preference towards coherent motion patterns. In fact, there is nothing in the model's task — image reconstruction — that requires temporal information; the task can be solved using the current frame alone. This observation led us to propose a *temporal similarity loss* that forces the model to learn from temporal information, and in this way incentivizes the object grouping to follow coherent motion.

7.2 Method

7.2.1 The VideoSAUR Model

VideoSAUR follows the basic framework for modeling videos popularized by the SAVi model (Kipf et al., 2022, see Sec. 3.3.2 for an introduction): put succinctly, it is a recurrent auto-encoder with a slot bottleneck implemented by Slot Attention. In more detail, each frame x^t is processed by an encoder into patch feature maps h^t, which Slot Attention then groups into slot representations z^t. Critically, Slot Attention is recurrently initialized from the slots of the previous time step, z^{t-1}, with the initialization for the first time step drawn from a normal distribution with learnable mean and variance, $z^0 \sim \mathcal{N}(\mu, \sigma^2)$. A decoder then predicts the model's outputs from the slots, independently for each frame.

In line with our findings from the Dinosaur model (Chap. 6), we utilized a pre-trained, fixed, self-supervised DINO ViT as the *encoder*. As the *decoder*, we utilized the SlotMixer (Sajjadi et al., 2022): this decoder assigns slots to spatial positions before decoding using a Transformer. This design is more efficient than the common spatial broadcast decoder, as it requires only a single decoding pass instead of a pass per slot. In the context of video models, we found this improved efficiency to be especially critical.

7.2.2 Object Discovery by Predicting Temporal Similarities

We now discuss a novel loss function, called the *temporal similarity loss*, that uses the motion cues available in videos to enhance object discovery. Intuitively, parts of a video that consistently move together can be considered to belong to the same object. We instantiate this principle of common fate (Wertheimer, 2012; Tangemann et al., 2023) by letting the model *predict the motion of image patches*. On a high level, the idea is that patches that exhibit similar motion patterns are likely to belong to the same object; predicting the motion thus incentivizes grouping all those patches together into one object. An overview is given in Figure 1 of Appendix D.

Concretely, we first compute an *affinity matrix* $\mathbf{A}^{t,t+k}$, containing the pairwise cosine similarities[3] between the L patch features of the current frame $\mathbf{h}^t$ and some future frame $\mathbf{h}^{t+k}$, where the features are given by the pre-trained ViT encoder:

$$\mathbf{A}^{t,t+k} = \frac{\mathbf{h}^t}{\|\mathbf{h}^t\|} \cdot \left(\frac{\mathbf{h}^{t+k}}{\|\mathbf{h}^{t+k}\|}\right)^\top, \quad \mathbf{A}^{t,t+k} \in [-1,1]^{L\times L}. \tag{7.1}$$

Then, we transform the affinity matrix $\mathbf{A}^{t,t+k}$ into a matrix of probabilities $\mathbf{P}^{t,t+k} \in [0,1]^{L\times L}$ by applying a row-wise softmax operation[4]:

$$P_{ij}^{t,t+k} = \frac{\exp(A_{ij}^{t,t+k}/\tau)}{\sum_{m=1}^{L} \exp(A_{im}^{t,t+k}/\tau)}, \tag{7.2}$$

where τ is the softmax's temperature. We can interpret entry $P_{ij}^{t,t+k}$ as the probability of patch i having moved to patch j after k frames. Finally, we use the resulting probability distributions as *targets* for the model to predict, defining the temporal similarity loss as the cross-entropy H between $\mathbf{P}^{t,t+k}$ and the model's output $\hat{\mathbf{P}}^{t,t+k}$:

$$\mathcal{L}^{\text{sim}} = \sum_{l=1}^{L} H\left(\mathbf{P}_l^{t,t+k}, \hat{\mathbf{P}}_l^{t,t+k}\right) = \sum_{l=1}^{L}\sum_{j=1}^{L} -P_{lj}^{t,t+k} \log \hat{P}_{lj}^{t,t+k}. \tag{7.3}$$

It is instructive to compare the temporal similarity loss with prior work utilizing *optical flow targets*, where the model is required to predict the movements of individual pixels (e.g. SAVi, Kipf et al., 2022). In both scenarios, the task of predicting motion introduces a bias towards grouping parts with similar movements together. However, unlike optical flow prediction, the feature-based approximation of motion also takes *semantic aspects* into account; this yields a useful signal towards object grouping even for *static* parts of the video, as the model needs to predict which patches are semantically similar. Furthermore, whereas optical flow prediction requires the model to output a single precise motion estimate (i.e. regression), the similarity loss is framed in terms of *modeling a probability distribution over possible movements*; we opted for this formulation as we expect it to be more robust in the face of inaccurate or ambiguous targets. Indeed, the SAVi model was found to degrade when

[3] Here, an underlying assumption is that these similarities reveal both the semantic and the spatial closeness of patches; we experimentally verified that this indeed holds for DINO features.

[4] In practice, the affinity matrix is thresholded at zero such that only patches with positive similarities are considered.

Table 7.1: Evaluating unsupervised video object discovery. On both synthetic (MOVi-C, left) and real-world (YouTube-VIS, right) data, VideoSAUR performs significantly better than the previous object-centric models SAVi and STEVE. The table shows mean ± standard dev. over 5 seeds for the FG-ARI and mBO metrics (higher is better; see Sec. 3.4.1 for an explanation). Metrics are computed over the full video, and thus also measure tracking consistency. Adapted from Zadaianchuk, Seitzer, et al. (2023).

	MOVi-C		**YouTube-VIS**	
	FG-ARI	mBO	FG-ARI	mBO
SAVi	22.2 ± 2.1	13.6 ± 1.6	11.1 ± 5.6	12.7 ± 2.3
STEVE	36.1 ± 2.3	26.5 ± 1.1	20.0 ± 1.5	20.9 ± 0.5
VideoSAUR	**64.8** ± **1.2**	**38.9** ± **0.6**	**39.5** ± **0.6**	**29.1** ± **0.4**

camera motion is introduced (Greff et al., 2022), whereas VideoSAUR was resilient to this change. A final significant difference is that the temporal similarity loss is fully unsupervised, and thus can be readily applied to any video — although optical flow can be estimated from videos (e.g. Stone et al., 2021), doing so robustly from in-the-wild videos is challenging.

Let us now discuss the role of the two *hyperparameters* of the loss, the *time shift into the future* k and the *softmax temperature* τ. They can be seen as complementary in the sense that the former has a temporal effect and the latter a spatial. Both affect the difficulty of the prediction task, and thus the model's learning signal. The time shift k should be chosen such that significant movements occur, but are not unpredictable; this depends on properties of the video such as the sampling rate. The effect of the softmax temperature τ is more intricate. To a first degree, it controls how concentrated the target distribution is around its maximum. Effectively, this trades off between two tasks: accurately predicting patch motion (low τ), or predicting full patch-to-patch similarities (high τ). Whereas the latter mode might be important to maintain a meaningful prediction task for almost static scenes, the former mode attenuates semantic similarity and enhances spatial similarity, which I conjecture plays an important role for separating objects of the same class.

7.3 Results

Let us briefly discuss the experimental results. We tested VideoSAUR on four video datasets, the synthetic datasets MOVi-C and MOVi-E (Greff et al., 2022), and the real-world datasets YouTube-VIS (L. Yang et al., 2021) and DAVIS (Pont-Tuset, Perazzi, et al., 2017), and compared it against two recent (at the time) video models, SAVi (Kipf et al., 2022) and STEVE (Singh, Wu, et al., 2022). See Table 7.1 for a representative subset of the results. In terms of unsupervised video object discovery, VideoSAUR model set a new state-of-the-art, improving performance over prior work by a significant margin. Both SAVi and STEVE struggle on the synthetic datasets, and fail to discover objects on real-world

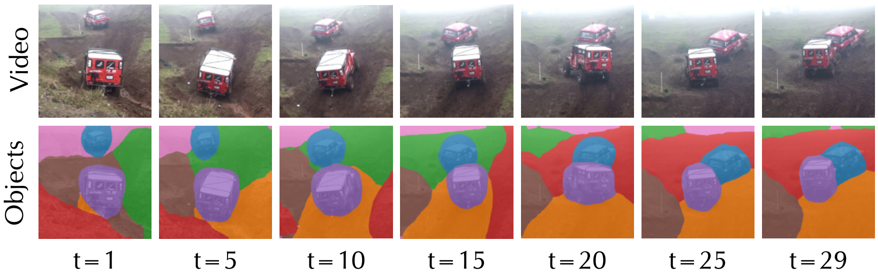


Figure 7.1: Object discovery and tracking on a video from the YouTube-VIS dataset. Adapted from Zadaianchuk, Seitzer, et al. (2023).

videos. These results demonstrate the importance of (1) using pre-trained features; and (2) exploiting temporal information to scale object discovery on videos.

Analysis VideoSAUR can be trained using the temporal similarity loss, Dinosaur's feature reconstruction loss, or both in conjunction. While we observed the model to achieve state-of-the-art results even solely with feature reconstruction, the temporal similarity loss provides further significant improvements. Interestingly, the two losses interact differently depending on the dataset: on the synthetic MOVi datasets, predicting temporal similarities brings drastic improvements over feature reconstruction (e.g. +20 FG-ARI on MOVi-C), and adding the latter does not yield additional benefits. In contrast, on the real-world YouTube-VIS dataset, we found feature reconstruction to be necessary for good performance; adding temporal similarity then also brings some further improvements. I conjecture this is because feature reconstruction introduces an additional *semantic bias* that is necessary for object discovery on real-world data. Furthermore, motion might be sparser and harder to predict for real videos, reducing the usefulness of the temporal similarity loss for object discovery.

7.4 Discussion

VideoSAUR was the first unsupervised object-centric learning method for unconstrained, YouTube-like videos (together with concurrent work, discussed below), and also achieved state-of-the-art results for unsupervised video object discovery on the challenging MOVi datasets. The main technical contribution was the novel temporal similarity loss, exploiting the temporal information available in videos for object discovery. By utilizing pre-trained self-supervised representations to construct motion targets, the loss incorporates both temporal and spatio-semantic correlations in a fully unsupervised manner. Thereby, the semantic nature of the underlying representations acts as a "fallback", and allows the loss to scale gracefully to scenes with little or no movements. Through its probabilistic formulation, the loss also naturally handles ambiguous or inaccurate targets.

As VideoSAUR is a direct follow-up to Dinosaur, the two models share obvious similarities: for instance, the focus on real-world data, or employing strong pre-trained supervised representations to scale to more complex data. Regarding the latter, VideoSAUR's temporal similarity loss introduces another way of utilizing these representations, demonstrating their versatility for object discovery. Beyond that, in both works, a key idea is to bias the model through prediction, adding a semantic bias by predicting features, or a motion bias by predicting temporal similarities. I believe this concept — *infusing the model with biases for object discovery through appropriate prediction targets* — to be an important emerging design principle for building object-centric models. Note that even though the two losses are constructed with the same underlying representations, they act complementarily: the losses provide *different views* on the underlying object structure, reducing uncertainty about the true decomposition by filtering out incompatible hypotheses.

7.4.1 Concurrent Work

Concurrently with VideoSAUR, two other works focused on scaling object-centric representations to real-world video: SOLV (Aydemir et al., 2023) and SMTC (R. Qian et al., 2023), both also utilizing pre-trained self-supervised features. SOLV, similar to VideoSAUR, employs a Dinosaur-style feature reconstruction loss but has a non-recurrent architecture: SOLV predicts intermediate frames from past and future frames — making it unsuitable for scenarios where frames need to processed in a streaming fashion, such as in RL or robotics. SMTC is a decoder-free architecture that employs student-teacher self-distillation (e.g. Grill et al., 2020) to learn to extract time-consistent masks and object representations. Interestingly, SMTC also computes inter-frame feature similarities (cf. Eq. (7.1)), but uses them to augment the *inputs* to Slot Attention instead of the *targets*. Both SOLV and SMTC are able to discover objects on real-world videos; in terms of performance, they are overall similar to VideoSAUR (Zadaianchuk, Seitzer, et al., 2023, App. A.2). The success of these three — quite different — methods demonstrates that there are many ways to exploit the temporal information in videos, but that the underlying self-supervised representations are essential for real-world scalability.

7.4.2 Limitations

Even though VideoSAUR is able to model relatively complex scenes from the YouTube-VIS dataset, this dataset is still curated (e.g. limited number of semantic classes, limited number of objects per video), and does not capture the full variety present in a true open-world setting. Another restriction concerns the short- and long-term consistency of slots, i.e. maintaining a stable object identity through time. In particular, VideoSAUR is limited in tracking objects through short-term occlusions; the model also cannot re-identify objects after their long-term absence because it is missing a memory module. A related problem is handling appearing and disappearing objects: SAVi-style recurrent architectures assume that all objects present in the first frame will stay present throughout the video, and that no new objects will be introduced. When this assumption is violated, identity switches of slots necessarily need to happen, violating the goal of consistency. Thus, a proper mechanism to model this situation, e.g. by appropriately adding or removing slots, is required.

7.4.3 Outlook

In this project, we only evaluated the discovery and tracking of objects through videos. An important next step is to use VideoSAUR's representations in downstream applications such as RL or robotics. I anticipate that such a "field test" will reveal more shortcomings of the model, some of which we discussed earlier.

There are many interesting directions for extending VideoSAUR: designing new loss functions or architectural inductive biases for enhancing consistency, integrating memory modules to allow long-term tracking, or adding mechanisms to handle (dis-)appearing objects. Another possibility is to derive a *variational Bayesian formulation* of VideoSAUR, similar to probabilistic recurrent state-space models (Karl et al., 2017; Doerr et al., 2018). The notion of uncertainty that comes with such a formulation would be helpful for learning *dynamics models*, i.e. forecasting how slots evolve and interact with each other through time. In turn, predicting the future would likely lead to more physics-aware representations, and help to address consistency problems. A further step would then be to learn *structured world models* (Ha and Schmidhuber, 2018; Kipf et al., 2020), that is, models that can predict the future following hypothetical actions. Such a model can be used to plan action sequences or as a simulation to train RL agents (Sutton, 1991). Modeling actions, or *interventions*, could also lead to interesting new biases for object discovery — entities that can be *independently influenced* should be considered objects (cf. Sec. 2.4.2).

Another direction is to model *the observer* as an explicit entity in addition to the objects, requiring disentangling the camera from object motion.[5] Doing so could be a first step towards exciting new capabilities, e.g. inferring an object's 3D properties purely from RGB video, or for simultaneous localization and mapping (SLAM) of agent and objects. Such a model could form the basis for an integrated, end-to-end approach to robotic perception, manipulation, and navigation.

Finally, a long-term vision for object-centric representation learning is to build general and flexible *structured foundation models* that can be applied in a zero-shot fashion in any kind of environment (see also discussion in Sec. 8.2.3). One essential ingredient to do so will be to train on large-scale video collections. VideoSAUR, with its focus on real-world data and its use of pre-trained self-supervised representations, is a first step in this direction.

[5] The author contributed first steps in this direction in a non-object-centric framework, see Seitzer et al. (2024).

Part IV

On the Future of Structured Object Representations

Chapter 8

Discussion

I conclude this thesis by presenting my perspective on the future of structured object representations, and my research's role in it. First, Sec. 8.1 summarizes the contributions of this thesis. Then, Sec. 8.2 discusses the implications of the key result of this thesis: real-world object-centric representations. Finally, Sec. 8.3 provides a critical discussion and speculates on the long-term prospects of structure in AI.

8.1 Summary of Contributions

This thesis is motivated by the shortcomings that contemporary AI systems exhibit compared to human cognition. My core hypothesis is that these limitations stem from their inability to learn and maintain structured object representations of the world, where I view objects as composable, modular units of information (Chap. 2). The overarching research question is how such representations can be integrated with neural networks in a way that preserves the advantages of both. This is known as the binding problem of neural networks (Greff et al., 2020), and the field that attempts to solve it is known as object-centric representation learning (Chap. 3). The research contributions of this thesis can be organized in terms of two aspects of the binding problem: segregation and composition.

8.1.1 Composing Object Representations for Autonomous Agents

In Part II, I investigated the aspect of composition. Broadly, the research question I aimed to answer was how structured object representations can allow agents to (1) learn autonomously and (2) learn efficiently. Regarding (1), I introduced the Smorl agent (Chap. 4), a self-supervised visual RL algorithm solving environments with a complex goal structure. By representing the environment in terms of objects, Smorl is able to perform relational reasoning, explore efficiently, and decompose complex goals into simpler sub-goals, all without being provided with explicit reward information. Regarding (2), I developed cai (Chap. 5), a measure of an agent's causal influence on objects in the environment.

By integrating this measure into RL algorithms as a structural inductive bias that guides learning and exploration, sample efficiency is drastically improved.

Both works constitute contributions to the field of reinforcement learning, specifically SMORL to the area of *self-supervised RL* (Colas et al., 2020) and CAI to the area of *causal RL* (Zeng et al., 2024). SMORL solved a challenging setting for the first time, namely multi-goal RL with multiple objects, purely from images and unsupervised. The key insight behind CAI was that the sparsity of agent-environment interaction is a significant source of inefficiency; CAI showed the effectiveness of a *causal perspective* to remedy this problem. Both projects also sparked several follow-up works, e.g. Zadaianchuk et al. (2022) and Haramati et al. (2024) for SMORL, and Zizhao Wang et al. (2023) and Urpí et al. (2024) for CAI.

From the point of view of object-centric representation learning, the contribution is twofold.

(I) *The technical contribution* is showing how object representations can be effectively composed for agent learning and discovering causal influence. Specifically, for SMORL, I designed an architecture enabling decomposed, goal-based exploration and relational reasoning between objects. Then, for CAI, I demonstrated how higher-order structures, namely the causal relationship between objects, can be learned in a model-based manner from object representations.

(II) *The experimental contribution* is verifying the presumed benefits of structured object representations in the practical setting of robot learning (cf. Sec. 2.2.1). In particular, I found evidence that such representations (1) constitute a useful inductive bias for downstream tasks; (2) lead to improved data efficiency; (3) enable a degree of out-of-distribution generalization; and (4) allow the design of informed algorithms through their interpretability. This adds to the growing body of work demonstrating the benefits of structured object representation in various settings.[1]

8.1.2 Real-World Object-Centric Representation Learning

In Part III, I investigated the aspect of segregation, that is, the discovery of object representations in visual data. Specifically, the research question I posed was whether and how object-centric representations can be learned on real-world data in an unsupervised manner. From my personal perspective, this was motivated by the results obtained in Part II, which were encouraging but limited by the scope of data that object-centric methods could be applied to. From the perspective of the research field, this question was timely as well: object-centric learning was still restricted to simplistic, synthetic datasets, and there was a clear gap to the real world settings that other research communities had long contended with.

This gap was bridged by the DINOSAUR model (Chap. 6), which is perhaps the most important contribution of this thesis. Based on the

[1] See e.g. Dittadi et al. (2022), Heravi et al. (2022), Driess et al. (2023), Kim, Kim, Lan, et al. (2023), Stanić, Tang, et al. (2023), Yoon et al. (2023), Mamaghan et al. (2024), and Jiaqi Xu et al. (2024).

analysis that previous methods failed on complex data due to a bias towards surface-level image statistics, I proposed to add a *semantic bias* into object-centric models. Furthermore, I demonstrated that this kind of bias can be injected by predicting pre-trained semantic features, for example the representations learned by modern self-supervised learning methods such as DINO. The resulting DINOSAUR model constituted a major step forward in the complexity of data that object-centric methods, both unsupervised and (weakly) supervised, could handle. Not only did it significantly improve the state-of-the-art on existing benchmark datasets; it was also the *first model* that successfully learned object-centric representations on unconstrained real-world datasets such as PASCAL VOC and COCO.

In Chap. 7, I continued this line of research by introducing the VIDEOSAUR model. This model can be seen as an extension of DINOSAUR to real-world videos — it similarly relied on self-supervised pre-trained features. Additionally, VIDEOSAUR exploits the temporal signal in videos to integrate a bias towards grouping parts with consistent motion together. This was implemented by a novel *temporal similarity loss* based on predicting feature motion. In this way, VIDEOSAUR was the first model[2] that successfully learned object-centric representations on videos of YouTube-style complexity.

Together, these works started what I call "*real-world object-centric representation learning*": a growing set of methods and applications targeting the "outside the lab" setting.[3] In contrast to previous work, this means leaving the confines of simulated, synthetic datasets — exchanging them for a natural, open-world, in-the-wild setting. Naturally, this raises a set of challenges that are not present in controlled environments; but it also affords several opportunities, as it connects object-centric representation learning to the wider machine learning landscape. I will analyze the wider implications of this in Sec. 8.2. To conclude, the final contribution of this thesis is thus *opening up the real-world setting for the field of object-centric representation learning*.

8.1.3 Limitations, Extensions, and Perspectives

As with any research, the work presented in this thesis has limitations. I already discussed the specific constraints of each proposed method in their respective chapters. In the remainder, I broaden the focus to analyze the future prospects of structured object representations. Starting with the short term, the next section discusses the real-world setting enabled by the contributions of this thesis, highlighting both the arising challenges and opportunities. I will also outline our follow-up work[4] that complements and extends my research presented before. In the last section, I first adopt a critical perspective on object-centric representation learning and examine some fundamental issues that may hinder its progress in the medium term, and then speculate on the role structured object representations might play in the longer-term evolution of AI.

[2] With concurrent work from Aydemir et al. (2023).

[3] See e.g. Fan et al. (2023), Kim, Kim, Lan, et al. (2023), Wu, Hu, et al. (2023), Kakogeorgiou et al. (2024), Mamaghan et al. (2024), and Jiaqi Xu et al. (2024).

[4] Didolkar, Zadaianchuk, et al. (2024). ""Zero-Shot Object-Centric Representation Learning"". Under review.

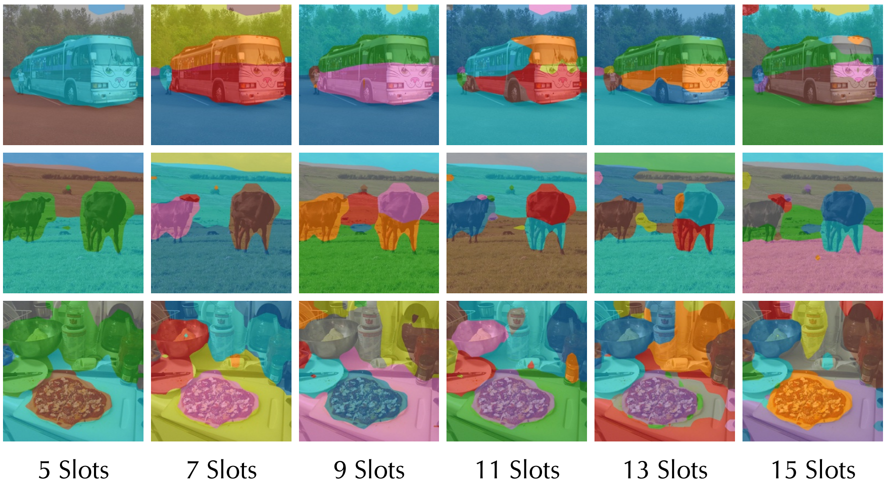


Figure 8.1: Ambiguities of real-world object discovery. Object decomposition of scenes with low, medium, and high complexity using DINOSAUR. Natural scenes admit several valid groupings; the model's grouping depends on the number of slots, and the complexity of the scene. Figure adapted from Seitzer et al. (2023).

8.2 New Frontiers for Object-Centric Learning

By becoming real-world viable, object-centric representation learning has reached a new stage of maturity. This progress introduces several challenges, such as the inherent *ambiguities* present in real-world data (Sec. 8.2.1), and the question of how to *scale* these methods effectively (Sec. 8.2.2). But addressing these challenges also presents exciting new opportunities, such as the development of "*object-centric foundation models*" that can be applied in a zero-shot manner across various settings (Sec. 8.2.3). Additionally, it opens the door for *practical applications* in areas like visual question answering or robotics, for which object-centric representations must also integrate with *new domains* such as natural language, or 3D data (Sec. 8.2.4). Finally, it challenges the research field to benchmark itself with alternative approaches, providing an opportunity to demonstrate the potential of object-centric representations (Sec. 8.2.5).

8.2.1 Dealing with Variations & Ambiguities

Natural images present several challenges that arise from the large variations and ambiguities encountered in the real world, including (1) a varying *number of objects* per scene; (2) a varying *complexity* per object; and (3) *ambiguities* from multiple plausible groupings. Synthetic data is constructed in a way that these challenges mostly do not occur; as such, prior methods developed with such data are ill-equipped to handle them.

Let us discuss the Slot Attention model as an example.[5] First, Slot Attention uses a *fixed number of slots* during training, unsuitable for modeling a varying number of objects.[6] However, even if the number of slots would match the number of objects, there remains a second issue: all slots have *equal capacity*, which is suboptimal for modeling scenes with objects of different complexities.[7] Consequently, the suitability of the discovered grouping strongly depends on the complexity of the scene and the number of slots (see Fig. 8.1, also Zimmermann et al., 2023).

[5] Methods derived from Slot Attention, such as DINOSAUR and VIDEOSAUR, inherit these limitations.

[6] The number of slots also cannot be arbitrarily increased, as this would clash with the *compression principle* driving object discovery (see Sec. 2.4.2).

[7] Complex objects tend to be split into multiple slots; several simple objects tend to be grouped into one slot.

Finally, for natural scenes, multiple groupings are often plausible, e.g. at different levels of the object hierarchy (cf. Sec. 2.4.1). The grouping to select depends on the intended task, and can even change dynamically[8]; however, Slot Attention lacks control over which grouping to select.

Two strategies could help address these issues: *regularization* and *conditioning*. Regularization involves penalizing the capacity of the slots so that the model can dynamically adapt to the complexity of the scene[9]; this could partially mitigate the issues related to a fixed number of slots and slot capacity. Conditioning involves inputting a parameter of interest into the model (e.g. number of slots, or the position of a slot), and *sampling* that parameter during training instead of fixing it. This way, the model learns to interpret scenes in multiple ways, and its grouping can be *controlled* at test time.[10]

8.2.2 Unlocking the Benefits of Scale

Being able to model real-world data unlocks vast quantities of data for training, with open datasets containing up to 5 billion images (Schuhmann et al., 2022). In contrast, synthetic datasets for object-centric learning typically contain 10k–100k data points; tiny by modern standards. The major advantage of unsupervised learning is that it is in principle possible to train on such large data collections — an opportunity to *scale* both data and models, following the playbook of today's foundation models.

However, initial experiments (Didolkar, Zadaianchuk, et al., 2024) indicated that the data scaling behavior of Dinosaur is suboptimal (see Fig. 8.2). The performance in terms of object discovery saturates quickly when increasing the size of the training dataset (already around 10k samples). This negative result raises the question of what prevents current object-centric models from scaling. One hypothesis is that the model scale is insufficient; however, while we found that increasing the model size had a moderately positive impact, it did not fundamentally change the observed trend. Another hypothesis is that the *representational bottleneck* of current architectures restricts the models' ability to fit the data.[11,12] If true, this problem is not easily addressed, as current object-centric models fundamentally *rely* on compression to discover objects (cf. Sec. 2.4.2).

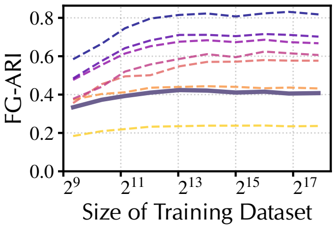


Figure 8.2: Data scaling on COCO. Dinosaur is trained on different numbers of samples and evaluated in-distribution (solid line) and various OOD datasets (dashed lines). Performance plateaus rapidly. Adapted from Didolkar, Zadaianchuk, et al. (2024).

Taking a step back, we must also examine the evaluated task, namely object discovery. First, it is possible that object discovery is close to saturation regarding the enhancements achievable through scaling. The main bottleneck to further improve on this task might instead lie in the limitations of object-centric models in handling variations and ambiguities, as discussed in Sec. 8.2.1. Second, it is conceivable that scaling does indeed improve the *representations*, but that this improvement may not be evident when evaluating *object discovery*. To reveal the benefits of scaling, object-centric models may need to be tested more comprehensively, for instance in terms of representational content, downstream applications, and generalization (see also the discussion of evaluation in Sec. 3.4).

[8] For example, imagine moving a plate of cookies to a table: the plate needs to be grouped as one object. Then, to eat the cookies, they need to be grouped separately. From Sancaktar et al. (2023).

[9] VAE-style regularization seems suitable to implement this (Kingma and Welling, 2014).

[10] However, this strategy does not provide a way to *automatically* set that parameter per data point.

[11] It is common deep learning wisdom that bottlenecks contradict scalability.

[12] I found Dinosaur to mostly operate in an underfitting regime. This could be due to limited model capacity, but also due to optimization difficulties stemming from the bottleneck.

It is likely that all of these factors contribute to the observed inability to scale.[13] Thus, to overcome the obstacles preventing scaling, it is essential to investigate these aspects further. I believe a *key goal* of real-world object-centric representation learning is to train large-scale models on large-scale datasets — realizing the full potential of unsupervised learning. One aspirational outcome of such a research program is to develop an "*object-centric foundation model*", as we will discuss next.

8.2.3 Zero-Shot Object-Centric Representations

In my discussion of Dinosaur (Chap. 6), I claimed that it was the first model for *unconstrained* real-world datasets. This is true in the sense that Dinosaur works on considerably less constrained data than previous methods; but ultimately, the datasets on which it was evaluated are still closed-world[14] and limited in scope (e.g. COCO). More important for future applications is the *open-world setting*, where the test data can differ from the training data in any way — a more realistic simulation of deployment conditions. To support open-world conditions, a model must be capable of transferring its knowledge in a *zero-shot manner* to novel situations.

Object-centric models, with their presumed generalization abilities, seem particularly suited for this task. However, so far, object-centric models have not been evaluated in such a zero-shot setting. To alleviate this, we recently proposed a zero-shot benchmark[15] and evaluated several Dinosaur-based methods on it (Didolkar, Zadaianchuk, et al., 2024). Interestingly, we found that all tested object-centric models exhibit considerable zero-shot transfer to unobserved domains. Our proposed model, FT-Dinosaur,[16] reaches zero-shot performance comparable to training *directly on the test distribution*, and is able to discover objects fully outside the training distribution (see Fig. 8.3).

These results suggest the exciting possibility of an "object-centric foundation model", a model that produces object representations across a variety of domains, facilitating a variety of tasks. A further insight is that the zero-shot robustness required for such a model could be attainable by integrating object-centric components with existing foundation models. For instance, in the case of FT-Dinosaur, pre-trained features from DINOv2 (Oquab et al., 2023) were fine-tuned for the task of object discovery within the Dinosaur model. Despite the remaining challenges (Secs. 8.2.1 and 8.2.2), I speculate that object-centric "backbones" capable of supporting real-world applications are thus within reach.

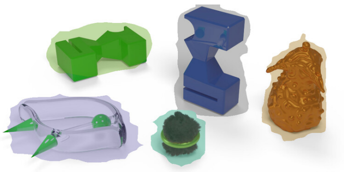


Figure 8.3: Zero-shot discovery. FT-Dinosaur (Didolkar, Zadaianchuk, et al., 2024) segregates "*A visual scene composed of various unfamiliar objects.*" (Greff et al., 2020). Image licensed under CC-BY 4.0.

8.2.4 New Applications, New Domains

Object-centric representation learning has always suffered from a lack of realistic applications. With the advent of real-world capable object-centric models, this situation could now change. In the following, I outline two areas where object-centric modeling could have an impact in the future: *visual question answering* (VQA) and *robotics*.[17] These two areas

[13] Particularities of the Dinosaur model, e.g. the fixed pre-trained targets, might also play a role.

[14] A closed-world dataset is constructed such that the data falls into known categories.

[15] Consisting of 7 datasets previously used for object-centric learning and the open-world EntitySeg dataset (Qi et al., 2023).

[16] FT-Dinosaur ≙ **F**ine**t**uned Dinosaur

[17] There are also more emerging applications, e.g. compositional generation (Jabri et al., 2024; Z. Wu et al., 2024) or world modeling (Wu, Dvornik, et al., 2023), which we are not going to cover here.

also benefit from integrating *new domains* into object-centric approaches, respectively *language*, and *3D*.

Visual Question Answering In the VQA task, the model is provided with an image or video and a textual question, and needs to either select from a set of pre-defined answer (closed world, e.g. Mamaghan et al., 2024) or respond in free-form language (open world, e.g. Driess et al., 2023; Jiaqi Xu et al., 2024). To perform well on this task, object comprehension and relational reasoning are crucial. Thus, object-centric representations appear naturally suited — their modular nature could provide a significant advantage over other kinds of representations (Mondal et al., 2023; Webb et al., 2023). Furthermore, the combinatorial nature of object relationships results in a vast number of possible questions; to manage this complexity, the compositional generalization that object representations could provide might be significant. Finally, specifically for *video* VQA, object representations could also offer a computationally manageable yet expressive abstraction (Jiaqi Xu et al., 2024).

From the point of view of object-centric representation learning, it would be particularly interesting to target the zero-shot, open-world setting, where the data is not restricted to pre-defined categories. This is because this setting encompasses many other tasks such as image classification, attribute prediction, object localization, object counting, or character recognition. Thus, the scope of object-centric models could be significantly increased by integrating them into such a VQA pipeline. As a less flexible but perhaps simpler alternative to open-world VQA, an open-vocabulary CLIP-style evaluation[18] could be considered as well (Jiarui Xu et al., 2022; Fan et al., 2023).

Language Grounding Both the VQA task and CLIP-style evaluation require an integration of the object representations with natural language. Depending on the setting, different implementations have been proposed; they all involve training to align the object representations with pre-trained language models:

- Closed-world VQA: object and text representations can be jointly processed by a "reasoning head" trained from scratch (D. Ding et al., 2021; Mamaghan et al., 2024).
- Open-world VQA: object representations can be learned to be mapped into the textual representation space of the language model, and then jointly processed with the query (Driess et al., 2023; Jiaqi Xu et al., 2024).
- CLIP-style: object representations can be aligned with pre-trained textual representations through contrastive learning (Jiarui Xu et al., 2022; Fan et al., 2023; Kim, Kim, Lan, et al., 2023).

For all settings, a dataset of images and textual descriptions is required; the quality of the language grounding depends on whether the descriptions capture the occurring objects. An idea yet unrealized is

[18] For lack of a better name; this involves comparing vision representations to a set of embeddings of textual queries, with the best match considered as the answer (Radford et al., 2021).

the *dynamic* integration of language into the object representation, for instance by *prompting* the model with textual descriptions of the objects to be captured — similar to SAM (Kirillov et al., 2023). Finally, it is interesting to note that this form of textual supervision can also be used to improve the quality of the object-centric representations themselves (cf. Sec. 2.4.2).

Robotics Object-centric representations are particularly well-suited for robotics for several reasons. First, many robotics tasks require fine-grained object understanding, which includes basic properties like position and shape, but also a comprehension of object affordances, dynamics, and inter-object relations. Second, robotics can benefit from the systematic generalization abilities that object-centric representations could offer. Finally, despite ongoing efforts to gather large-scale robotics datasets (Open X-Embodiment Collaboration et al., 2023), there is still a need for data effiency; data collection for both online learning and learning from demonstrations is labor-intensive and costly.

To integrate object representations into robotics, let us first consider a tabula-rasa approach where the robot learns from scratch on each environment with RL (similar to Part II of this thesis). In principle, algorithms like Smorl (Chap. 4) could be scaled-up using modern object-centric representations obtained from models like Dinosaur or VideoSAUR. While such an approach could achieve some success (Heravi et al., 2022; Haramati et al., 2024), it would likely be limited in scope and inefficient due to learning control from scratch.[19]

A strategy that has shown promise recently is to use demonstration data with *imitation learning* to train a policy instead of RL. Such demonstrations are typically sparse; thus, to be as robust as possible, approaches extensively use *pre-trained representations*, often trained on large-scale, out-of-domain datasets (S. Nair et al., 2022; Radosavovic et al., 2022; Xiao et al., 2022; Ma et al., 2023). For example, R3M (S. Nair et al., 2022) trains a vision encoder on the Ego4D dataset, which contains 3 670 hours of video footage depicting human daily life activities (Grauman et al., 2022). In these setups, the representations can be easily substituted with object-centric ones. Recent research has begun to explore this direction, with promising results (Ferraro et al., 2023; J. Qian et al., 2024; Shi et al., 2024). However, their object representations have been manually constructed by chaining foundation models. For example, Shi et al. (2024) combine SAM (Kirillov et al., 2023) for localization with LIV (Ma et al., 2023) for representation. It would be intriguing to investigate the use of representations from an end-to-end model like VideoSAUR in this context; an advantage of employing end-to-end representations could be the potential for task- and domain-specific fine-tuning.

3D-Aware Representations How can object-centric representations be further enhanced for robotics applications? Most pre-trained representations for robotics are image-based; hence, video-based representations

[19] Although object-centric exploration strategies like CAI (Chap. 5) and others (Blaes et al., 2019; Sancaktar et al., 2022, 2023) could alleviate this to some degree.

could already offer improvements in terms of handling consistency and dynamics. An additional critical advantage could be derived from *integrating 3D information into the representations*. There is a line of research focused on such 3D-aware object-centric representations (Stelzner et al., 2021; Sajjadi et al., 2022; Yu et al., 2022), which are trained on multi-view data via *novel view synthesis*. It has been shown that such representations are indeed beneficial for robotic manipulation (Driess et al., 2023). However, 3D object-centric modeling has not yet been scaled to real-world data — integrating Dinosaur with such approaches is a promising direction for future work. Another interesting direction could involve modeling the camera as an explicit entity (Seitzer et al., 2024), thereby disentangling observer motion from object motion.

8.2.5 From Niche to Mainstream?

I have discussed a multitude of avenues for future research that have emerged with the transition to the real world. Importantly, this shift also puts object-centric representation learning in context with the broader machine learning landscape. As a result, for the first time, it becomes possible to directly compare object-centric approaches to more mainstream methodologies using established benchmarks. For the task of object discovery, this includes computer vision methods for unsupervised instance segmentation,[20] which have recently seen enormous progress.[21] In terms of representations, this includes well-established self-supervised learning methods.[22]

Ultimately, object-centric representation learning aspires to support all kinds of applications and tasks. To showcase its relevance, it will eventually become necessary to compare with the top-performing approaches in each particular setting, whether supervised or unsupervised. This also includes comparisons with foundation models at the largest scales.[23,24]

Competing with such methods is a daunting prospect. However, real-world object-centric representation learning is largely uncharted territory, and its practical advantages have only recently begun to emerge. To further advance the field, it will be important to develop testbeds where the beneficial properties of object representations can be fully demonstrated. These could include applications involving the physical world, or those requiring systematic generalization, robustness, or efficient resource utilization, both in terms of data and compute. *Robotics* appears to be a natural candidate where many of those aspects converge. In conclusion, I believe that the full potential of object-centric representations has not yet been realized — realizing this potential requires overcoming the aforementioned challenges, but would let the field move beyond its current niche status.

[20] E.g. Xinlong Wang et al. (2022) and Xudong Wang et al. (2023, 2024).

[21] We already compared to such methods with both Dinosaur and VideoSAUR, with competitive results.

[22] E.g. Caron et al. (2021), He et al. (2022), and Oquab et al. (2023).

[23] E.g. LLMs (OpenAI, 2024), VLMs (Radford et al., 2021), or pure vision models (Kirillov et al., 2023; Oquab et al., 2023).

[24] In Didolkar, Zadaianchuk, et al. (2024), we started to compare object-centric models to the supervised large-scale SAM model (Kirillov et al., 2023).

8.3 Structured Object Representations: An Outlook

This thesis was about structured object representations. In the last chapter, we discussed the great potential of its current instantiation, object-centric representation learning. But we also saw obstacles arising, which raises the question of whether object-centric representation learning has fundamental problems that simply cannot be overcome — requiring a new set of solutions to integrate structure into contemporary AI.

The Limits of the Slot Paradigm In Sec. 8.2.1, we examined the difficulties of object-centric models face in dealing with variations and ambiguities. Although some of these issues may be addressable, it could be that the slot paradigm — representing the scene as a flat set of fixed size vectors — is ultimately too limiting to model the real world. For instance, it does not allow for modeling hierarchical structures at different leves of granularity. Additionally, the assumptions behind slots — a fixed description length and common representational format for each object — may be overly simplistic. Finally, the static nature of these models forces them to always infer a singular, complete interpretation of the scene. This is caused by the reconstruction objective in conjunction with the compression principle, which drives the full scene to be represented in the bottleneck. In contrast, Greff et al.'s (2020) vision to approach the binding problem suggests that object grouping should be *dynamic* and driven by *task-specific context*. This aspect is fully missing from current object-centric models. All in all, this suggests an eventual paradigm shift away from slot-based models.

The Bitter Lesson: Redux An even more fundamental limitation may arise from what we discussed in Sec. 8.2.2: the fact that current object-centric models face challenges when scaling to larger models and datasets. Reflecting on this, these challenges may suggest that we have once again *fallen prey to the bitter lesson*. In particular, by (over-)optimizing for the task of object discovery, that is, segmenting objects exactly as prescribed by human annotations, we engineered numerous biases into the models. But these biases ultimately prevent scaling, which is what matters most in the long run. Richard Sutton describes this pitfall clearly (emphasis mine):

> [We] should stop trying to find simple ways [. . .] to think about space, *objects*, multiple agents, or symmetries. [. . .] They are not what should be built in, as their complexity is endless; instead we should build in only the meta-methods that can find and capture this arbitrary complexity. [. . .] the search for them should be by our methods, not by us. *We want AI agents that can discover like we can, not which contain what we have discovered*. Building in our discoveries only makes it harder to see how the discovering process can be done. Sutton (2019), *The Bitter Lesson*

Applied to our context, it could be said that it was not the *models* that discovered objects, but rather *us*, forcing the models to discover objects.

Is Scale All We Need? Following Sutton's argument, we can ask whether the current scaling paradigm already comprises "*AI agents that can discover like we can*". For example, can a Transformer already discover all needed structure if it is just scaled-up enough? The conjecture behind this is that structural properties like compositionality or modularity could *emerge* at certain scales; essentially, if a structure is *useful* for the task the model is trying to solve (e.g. next token prediction), the model will learn to discover and utilize it. If this were true, we would not need to explicitly encode structure into the model, but simply combine scaling with hard-enough tasks that necessitate the discovery of structure.

On the one hand, there is evidence of such emerging structure in Transformers, e.g. the tracking of objects, even if they are not explicitly trained for this purpose (Caron et al., 2021; Sun et al., 2023; Kowal et al., 2024). There are also more basic results that show that neural networks can, in principle, learn modular structures from data (Lepori et al., 2023). On the other hand, there is clear evidence that current large models have problems with compositionality (Yuksekgonul et al., 2023; Kobayashi et al., 2024), especially for difficult tasks (Dziri et al., 2023; Z. Xu et al., 2024). These problems arise even though Transformers *do already possess* a structured representation[25] — conceivably giving it the computational substrate to learn to perform modular operations.

Scalable Modular Representations Do large-scale models require more structure? While there is no conclusive evidence in either direction at this point, we can still speculate how the principles underlying structured object representations could be applied to large models. We previously discussed how using bottlenecks to discover structure may contradict scalability. Essentially, bottlenecks constrain learning too much; instead, we should use soft mechanisms that encourage the formation of structure without enforcing it. Central to this endeavor are mechanisms that *learn better structures as more data and compute becomes available*. The mutual information-based principles of *independence* and *predictiveness* (see Sec. 2.4.2) might be well-suited for the design of such mechanisms; their success hinges on whether the estimation of mutual information can be transformed into productive learning principles.[26] As the outcome, I envision "*scalable modular representations*" — representations that capture modular structure implicitly, exchanging explicit grounding for the ability to scale.[27]

Conclusion I have discussed my perspective on the *future of structured object representations*. To summarize: in the short term, the field of object-centric representation learning is full of potential. In the medium term, however, its more fundamental limitations may require a change of paradigm. In the long term, I see three possible futures unfolding, defined by their use of *explicit*, *implicit*, or *emergent* structured object representations; in any case, I believe the principles of structure will play a crucial role. Hopefully, this thesis can contribute a minor step in this direction.

[25] That is, the set of tokens they operate on.

[26] Indeed, maximizing mutual information already plays a key role in self-supervised representation learning (Aaron van den Oord et al., 2019; Wang and Isola, 2022).

[27] Such representations can therefore no longer be called "object-centric". The resulting models may not resemble today's object-centric learning models much, but ultimately carry on the flag of structured object representations.

Bibliography

Agrawal, Raj, Chandler Squires, Karren Yang, Karthik Shanmugam, and Caroline Uhler (2019). "ABCD-Strategy: Budgeted Experimental Design for Targeted Causal Structure Discovery". In: *Artificial Intelligence and Statistics Conference (AISTATS)* (cit. on p. 58).

Ahn, Michael et al. (2022). "Do As I Can and Not As I Say: Grounding Language in Robotic Affordances". In: *Conference on Robot Learning (CoRL)* (cit. on p. 48).

Andrychowicz, Marcin, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, Pieter Abbeel, and Wojciech Zaremba (2017). "Hindsight Experience Replay". In: *Conference on Neural Information Processing Systems (NIPS)* (cit. on pp. 40, 44, 56).

Arbeláez, Pablo, Michael Maire, Charless Fowlkes, and Jitendra Malik (2011). "Contour Detection and Hierarchical Image Segmentation". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 33.5, pp. 898–916 (cit. on p. 32).

Arkoudas, Konstantine (2023). "GPT-4 Can't Reason". In: *arXiv:2308.03762* (cit. on p. 9).

Assran, Mahmoud, Mathilde Caron, Ishan Misra, Piotr Bojanowski, Florian Bordes, Pascal Vincent, Armand Joulin, Michael G. Rabbat, and Nicolas Ballas (2022). "Masked Siamese Networks for Label-Efficient Learning". In: *European Conference on Computer Vision (ECCV)* (cit. on pp. 65, 66, 68).

Attneave, Fred (1971). "Multistability in perception". In: *Scientific American* 225.6, pp. 62–71. ISSN: 0036-8733 (cit. on p. 24).

Aydemir, Görkay, Weidi Xie, and Fatma Güney (2023). "Self-supervised Object-Centric Learning for Videos". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on pp. 69, 76, 83).

Ba, Jimmy Lei, Jamie Ryan Kiros, and Geoffrey E. Hinton (2016). "Layer Normalization". In: *NIPS 2016 Deep Learning Symposium* (cit. on p. 27).

Bagon, Shai, Oren Boiman, and Michal Irani (2008). "What Is a Good Image Segment? A Unified Approach to Segment Extraction". In: *European Conference on Computer Vision (ECCV)* (cit. on p. 17).

Bahdanau, Dzmitry, Shikhar Murty, Michael Noukhovitch, Thien Huu Nguyen, Harm de Vries, and Aaron Courville (2019). "Systematic generalization: what is required and can it be learned?" In: *International Conference on Learning Representations (ICLR)* (cit. on p. 10).

Baldassarre, Federico and Hossein Azizpour (2022). "Towards Self-Supervised Learning of Global and Object-Centric Representations". In: *ICLR Workshop on the Elements of Reasoning: Objects, Structure and Causality* (cit. on pp. 16, 25).

Balestriero, Randall, Mark Ibrahim, Vlad Sobal, Ari Morcos, Shashank Shekhar, Tom Goldstein, Florian Bordes, Adrien Bardes, Gregoire Mialon, Yuandong Tian, Avi Schwarzschild, Andrew Gordon Wilson, Jonas Geiping, Quentin Garrido, Pierre Fernandez, Amir Bar, Hamed Pirsiavash, Yann LeCun, and Micah Goldblum (2023). "A Cookbook of Self-Supervised Learning". In: *arXiv:2304.12210* (cit. on pp. 63, 65).

Bao, Zhipeng, Pavel Tokmakov, Allan Jabri, Yu-Xiong Wang, Adrien Gaidon, and Martial Hebert (2022). "Discovering Objects that Can Move". In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on pp. 16, 24, 64, 72).

Bao, Zhipeng, Pavel Tokmakov, Yu-Xiong Wang, Adrien Gaidon, and Martial Hebert (2023). "Object Discovery from Motion-Guided Tokens". In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on pp. 12, 65).

Bauer, Jakob et al. (2023). "Human-Timescale Adaptation in an Open-Ended Task Space". In: *International Conference on Machine Learning (ICML)* (cit. on p. 48).

Bengio, Yoshua, Aaron Courville, and Pascal Vincent (2013). "Representation Learning: A Review and New Perspectives". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* (cit. on pp. 1, 7, 8, 13).

Betker, James, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, Wesam Manassra, Prafulla Dhariwal, Casey Chu, Yunxin Jiao, and Aditya Ramesh (2023). *Improving Image Generation with Better Captions*. OpenAI Blog (cit. on p. 9).

Biza, Ondrej, Sjoerd van Steenkiste, Mehdi S. M. Sajjadi, Gamaleldin F. Elsayed, Aravindh Mahendran, and Thomas Kipf (2023). "Invariant Slot Attention: Object Discovery with Slot-Centric Reference Frames". In: *International Conference on Machine Learning (ICML)* (cit. on pp. 21, 22).

Blaes, Sebastian, Marin Vlastelica Pogančić, Jiajie Zhu, and Georg Martius (2019). "Control What You Can: Intrinsically Motivated Task-Planning Agent". In: *Conference on Neural Information Processing Systems (NeurIPS)*. Vol. 32, pp. 12541–12552 (cit. on pp. 47, 88).

Brady, Jack, Roland S. Zimmermann, Yash Sharma, Bernhard Schölkopf, Julius von Kügelgen, and Wieland Brendel (2023). "Provably Learning Object-Centric Representations". In: *International Conference on Machine Learning (ICML)* (cit. on pp. 16, 28).

Brendel, Wieland and Matthias Bethge (2019). "Approximating CNNs with Bag-of-local-Features models works surprisingly well on ImageNet". In: *International Conference on Learning Representations (ICLR)* (cit. on p. 11).

Brown, Tom, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. (2020). "Language models are few-shot learners". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on pp. 2, 9, 64).

Bubeck, Sébastien, Varun Chandrasekaran, Ronen Eldan, John A. Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuan-Fang Li, Scott M. Lundberg, Harsha Nori, Hamid Palangi, Marco Tulio Ribeiro, and Yi Zhang (2023). "Sparks of Artificial General Intelligence: Early experiments with GPT-4". In: *arXiv:2303.12712* (cit. on p. 9).

Burgess, Christopher P., Loic Matthey, Nicholas Watters, Rishabh Kabra, Irina Higgins, Matt Botvinick, and Alexander Lerchner (2019). "MONet: Unsupervised Scene Decomposition and Representation". In: *arXiv:1901.11390* (cit. on pp. 2, 12, 20, 23).

Caron, Mathilde, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin (2020). "Unsupervised Learning of Visual Features by Contrasting Cluster Assignments". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on p. 65).

Caron, Mathilde, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin (2021). "Emerging Properties in Self-Supervised Vision Transformers". In: *IEEE International Conference on Computer Vision (ICCV)* (cit. on pp. 4, 63, 65, 66, 68, 89, 91).

Chakravarthy, Ayush, Trang Nguyen, Anirudh Goyal, Yoshua Bengio, and Michael C. Mozer (2023). "Spotlight Attention: Robust Object-Centric Learning With a Spatial Locality Prior". In: *arXiv:2305.19550* (cit. on pp. 17, 24, 68).

Chang, Michael, Thomas L. Griffiths, and Sergey Levine (2022). "Object Representations as Fixed Points: Training Iterative Refinement Algorithms with Implicit Differentiation". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on pp. 22, 27).

Chater, Nick (1996). "Reconciling Simplicity and Likelihood Principles in Perceptual Organization". In: *Psychological review* 103, pp. 566–81 (cit. on p. 14).

Chen, Ricky T. Q., Xuechen Li, Roger Grosse, and David Duvenaud (2018). "Isolating Sources of Disentanglement in Variational Autoencoders". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on p. 34).

Chen, Ting, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton (2020). "A Simple Framework for Contrastive Learning of Visual Representations". In: *International Conference on Machine Learning (ICML)* (cit. on p. 65).

Chen, Xinlei, Saining Xie, and Kaiming He (2021). "An Empirical Study of Training Self-Supervised Vision Transformers". In: *IEEE International Conference on Computer Vision (ICCV)* (cit. on pp. 65, 68).

Chiang, Wei-Lin, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing (2023). *Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90% ChatGPT Quality*. (Cit. on p. 69).

Cho, Kyunghyun, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio (2014). "Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation". In: *Conference on Empirical Methods in Natural Language Processing (EMNLP)* (cit. on p. 27).

Chomsky, Noam (2002). *Syntactic Structures*. A Mouton classic. Mouton de Gruyter. ISBN: 9783110172799 (cit. on p. 1).

Colas, Cédric, Tristan Karch, Olivier Sigaud, and Pierre-Yves Oudeyer (2020). "Autotelic Agents with Intrinsically Motivated Goal-Conditioned Reinforcement Learning: A Short Survey". In: *Journal of Artificial Intelligence Research (JAIR)* 74, pp. 1159–1199 (cit. on pp. 4, 82).

Cover, Thomas M. and Joy A. Thomas (2006). *Elements of Information Theory*. 2nd ed. Wiley Series in Telecommunications and Signal Processing. Wiley-Interscience (cit. on pp. 14, 53).

Crawford, Eric and Joelle Pineau (2020a). "Exploiting Spatial Invariance for Scalable Unsupervised Object Tracking". In: *AAAI Conference on Artificial Intelligence* (cit. on p. 23).

Crawford, Eric and Joelle Pineau (2020b). "Spatially Invariant Unsupervised Object Detection with Convolutional Neural Networks". In: *AAAI Conference on Artificial Intelligence* (cit. on pp. 20, 21).

Dang-Nhu, Raphaël (2022). "Evaluating Disentanglement of Structured Representations". In: *International Conference on Learning Representations (ICLR)* (cit. on p. 34).

Deng, Zhiwei, Ting Chen, and Yang Li (2024). "Perceptual Group Tokenizer: Building Perception with Iterative Grouping". In: *International Conference on Learning Representations (ICLR)* (cit. on p. 26).

Didolkar, Aniket, Anirudh Goyal, and Yoshua Bengio (2024). "Cycle Consistency Driven Object Discovery". In: *International Conference on Learning Representations (ICLR)* (cit. on pp. 24, 68).

Didolkar, Aniket, Andrii Zadaianchuk, Anirudh Goyal, Mike Mozer, Yoshua Bengio, Georg Martius, and Maximilian Seitzer (2024). "Zero-Shot Object-Centric Representation Learning". In: *arXiv:2408.09162* (cit. on pp. 35, 67, 70, 83, 85, 86, 89).

Ding, David, Felix Hill, Adam Santoro, Malcolm Reynolds, and Matt Botvinick (2021). "Attention over Learned Object Embeddings Enables Complex Visual Reasoning". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on pp. 3, 87).

Ding, Wenhao, Haohong Lin, Bo Li, and Ding Zhao (2022). "Generalizing Goal-Conditioned Reinforcement Learning with Variational Causal Reasoning". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on p. 59).

Ding, Wenhao, Laixi Shi, Yuejie Chi, and Ding Zhao (2023). "Seeing is not Believing: Robust Reinforcement Learning against Spurious Correlation". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on p. 59).

Dittadi, Andrea, Samuele Papa, Michele De Vita, Bernhard Schölkopf, Ole Winther, and Francesco Locatello (2022). "Generalization and Robustness Implications in Object-Centric Learning". In: *International Conference on Machine Learning (ICML)* (cit. on pp. 2, 10, 31, 32, 34, 82).

Dittadi, Andrea, Frederik Träuble, Francesco Locatello, Manuel Wuthrich, Vaibhav Agrawal, Ole Winther, Stefan Bauer, and Bernhard Schölkopf (2021). "On the Transfer of Disentangled Representations in Realistic Settings". In: *International Conference on Learning Representations (ICLR)* (cit. on p. 34).

Doerr, Andreas, Christian Daniel, Martin Schiegg, Duy Nguyen-Tuong, Stefan Schaal, Marc Toussaint, and Sebastian Trimpe (2018). "Probabilistic Recurrent State-Space Models". In: *International Conference on Machine Learning (ICML)* (cit. on p. 77).

Dosovitskiy, Alexey, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby (2021). "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale". In: *International Conference on Learning Representations (ICLR)* (cit. on p. 64).

Driess, Danny et al. (2023). "PaLM-E: An Embodied Multimodal Language Model". In: *International Conference on Machine Learning (ICML)* (cit. on pp. 2, 3, 10, 11, 48, 82, 87, 89).

Dziri, Nouha, Ximing Lu, Melanie Sclar, Xiang Lorraine Li, Liwei Jiang, Bill Yuchen Lin, Sean Welleck, Peter West, Chandra Bhagavatula, Ronan Le Bras, Jena D. Hwang, Soumya Sanyal, Xiang Ren, Allyson Ettinger, Zaid Harchaoui, and Yejin Choi (2023). "Faith and Fate: Limits of Transformers on Compositionality". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on p. 91).

Eastwood, Cian and Christopher K. I. Williams (2018). "A framework for the quantitative evaluation of disentangled representations". In: *International Conference on Learning Representations (ICLR)* (cit. on p. 34).

Eichler, Michael (2012). *Causal Inference in Time Series Analysis*. Wiley. ISBN: 9781119945710 (cit. on p. 58).

Elhage, Nelson et al. (2021). "A Mathematical Framework for Transformer Circuits". In: *Transformer Circuits Thread* (cit. on p. 11).

Elich, Cathrin, Martin R. Oswald, Marc Pollefeys, and Jörg Stueckler (2022). "Weakly supervised learning of multi-object 3D scene decompositions using deep shape priors". In: *Computer Vision and Image Understanding* 220. ISSN: 1077-3142 (cit. on p. 17).

Elsayed, Gamaleldin Fathy, Aravindh Mahendran, Sjoerd van Steenkiste, Klaus Greff, Michael Curtis Mozer, and Thomas Kipf (2022). "SAVi++: Towards End-to-End Object-Centric Learning from Real-World Videos". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on pp. 25, 64, 65, 72).

Engelcke, Martin, Oiwi Parker Jones, and Ingmar Posner (2020). "Reconstruction Bottlenecks in Object-Centric Generative Models". In: *ICML 2020 Workshop on Object-Oriented Learning* (cit. on pp. 14, 15).

Engelcke, Martin, Oiwi Parker Jones, and Ingmar Posner (2021). "GENESIS-V2: Inferring Unordered Object Representations without Iterative Refinement". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on pp. 20, 23).

Engelcke, Martin, Adam R. Kosiorek, Oiwi Parker Jones, and Ingmar Posner (2020). "GENESIS: Generative Scene Inference and Sampling with Object-Centric Latent Representations". In: *International Conference on Learning Representations (ICLR)* (cit. on pp. 2, 12, 20, 22, 32, 33).

Eslami, S. M. Ali, Nicolas Heess, Theophane Weber, Yuval Tassa, David Szepesvari, Koray Kavukcuoglu, and Geoffrey E. Hinton (2016). "Attend, Infer, Repeat: Fast Scene Understanding with Generative Models". In: *Conference on Neural Information Processing Systems (NIPS)* (cit. on pp. 2, 20, 21).

Everingham, Mark, Luc Van Gool, Christopher K. I. Williams, John M. Winn, and Andrew Zisserman (2010). "The Pascal Visual Object Classes (VOC) Challenge". In: *International Journal of Computer Vision* (cit. on pp. 4, 63).

Fan, Ke, Zechen Bai, Tianjun Xiao, Tong He, Max Horn, Yanwei Fu, Francesco Locatello, and Zheng Zhang (2024). "Adaptive Slot Attention: Object Discovery with Dynamic Slot Number". In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on pp. 24, 68).

Fan, Ke, Zechen Bai, Tianjun Xiao, Dominik Zietlow, Max Horn, Zixu Zhao, Carl-Johann Simon-Gabriel, Mike Zheng Shou, Francesco Locatello, Bernt Schiele, Thomas Brox, Zheng Zhang, Yanwei Fu, and Tong He (2023). "Unsupervised Open-Vocabulary Object Localization in Videos". In: *IEEE International Conference on Computer Vision (ICCV)* (cit. on pp. 69, 83, 87).

Ferraro, Stefano, Pietro Mazzaglia, Tim Verbelen, and Bart Dhoedt (2023). "FOCUS: Object-Centric World Models for Robotic Manipulation". In: *NeurIPS 2023 Intrinsically-Motivated and Open-Ended Learning Workshop (IMOL)* (cit. on p. 88).

Fodor, Jerry A. (1975). *The Language of Thought*. Language and thought series. Harvard University Press. ISBN: 9780674510302 (cit. on p. 1).

Foo, Alex, Wynne Hsu, and Mong-Li Lee (2023). "Multi-Object Representation Learning via Feature Connectivity and Object-Centric Regularization". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on pp. 17, 25).

Fowlkes, E. B. and C. L. Mallows (1983). "A Method for Comparing Two Hierarchical Clusterings". In: *Journal of the American Statistical Association* 78.383, pp. 553–569. ISSN: 01621459 (cit. on p. 33).

Gärdenfors, Peter (2014). *The Geometry of Meaning: Semantics Based on Conceptual Spaces*. The MIT Press. ISBN: 9780262319584 (cit. on p. 17).

Geirhos, Robert, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard S. Zemel, Wieland Brendel, Matthias Bethge, and Felix Wichmann (2020). "Shortcut learning in deep neural networks". In: *Nature Machine Intelligence* 2, pp. 665–673 (cit. on pp. 2, 8, 11).

Geirhos, Robert, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix Wichmann, and Wieland Brendel (2019). "ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness". In: *International Conference on Learning Representations (ICLR)* (cit. on p. 11).

Ghosh, Dibya, Abhishek Gupta, and Sergey Levine (2019). "Learning Actionable Representations with Goal-Conditioned Policies". In: *International Conference on Learning Representations (ICLR)* (cit. on p. 40).

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville (2016). *Deep Learning*. MIT Press (cit. on pp. 1, 8).

Goyal, Anirudh and Yoshua Bengio (2022). "Inductive biases for deep learning of higher-level cognition". In: *Proceedings of the Royal Society* 478.2266 (cit. on p. 2).

Goyal, Anirudh, Alex Lamb, Jordan Hoffmann, Shagun Sodhani, Sergey Levine, Yoshua Bengio, and Bernhard Schölkopf (2021). "Recurrent Independent Mechanisms". In: *International Conference on Learning Representations (ICLR)* (cit. on p. 16).

Grauman, Kristen et al. (2022). "Ego4D: Around the World in 3,000 Hours of Egocentric Video". In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on p. 88).

Greff, Klaus et al. (2022). "Kubric: A Scalable Dataset Generator". In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on pp. 25, 64, 74).

Greff, Klaus, Raphaël Lopez Kaufman, Rishabh Kabra, Nick Watters, Christopher Burgess, Daniel Zoran, Loic Matthey, Matthew Botvinick, and Alexander Lerchner (2019). "Multi-Object Representation Learning with Iterative Variational Inference". In: *International Conference on Machine Learning (ICML)* (cit. on pp. 2, 12, 20, 22, 23).

Greff, Klaus, Antti Rasmus, Mathias Berglund, Tele Hao, Harri Valpola, and Jürgen Schmidhuber (2016). "Tagger: Deep Unsupervised Perceptual Grouping". In: *Conference on Neural Information Processing Systems (NIPS)* (cit. on p. 2).

Greff, Klaus, Sjoerd van Steenkiste, and Jürgen Schmidhuber (2017). "Neural Expectation Maximization". In: *Conference on Neural Information Processing Systems (NIPS)* (cit. on pp. 2, 12, 20, 22, 23, 25).

Greff, Klaus, Sjoerd Van Steenkiste, and Jürgen Schmidhuber (2020). "On the Binding Problem in Artificial Neural Networks". In: *arXiv:2012.05208* (cit. on pp. 2, 9, 10, 12, 13, 15, 19, 22, 24, 39–41, 81, 86, 90).

Grill, Jean-Bastien, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Remi Munos, and Michal Valko (2020). "Bootstrap Your Own Latent - A New Approach to Self-Supervised Learning". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on pp. 65, 76).

Gu, Qiao, Ali Kuwajerwala, Sacha Morin, Krishna Murthy Jatavallabhula, Bipasha Sen, Aditya Agarwal, Corban Rivera, William Paul, Kirsty Ellis, Ramalingam Chellappa, Chuang Gan, Celso Miguel de Melo, Joshua B Tenenbaum, Antonio Torralba, Florian Shkurti, and Liam Paull (2024). "ConceptGraphs: Open-Vocabulary 3D Scene Graphs for Perception and Planning". In: (cit. on p. 11).

Ha, David and Jürgen Schmidhuber (2018). "Recurrent World Models Facilitate Policy Evolution". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on pp. 42, 77).

Haarnoja, T., Aurick Zhou, P. Abbeel, and S. Levine (2018). "Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor". In: *International Conference on Machine Learning (ICML)* (cit. on p. 44).

Halpern, Joseph Y. (2016). *Actual Causality*. The MIT Press. ISBN: 9780262035026 (cit. on p. 52).

Hamdan, Shadi and Fatma Guney (2024). "CarFormer: Self-Driving with Learned Object-Centric Representations". In: *European Conference on Computer Vision (ECCV)* (cit. on p. 11).

Hamilton, Mark, Zhoutong Zhang, Bharath Hariharan, Noah Snavely, and William T. Freeman (2022). "Unsupervised Semantic Segmentation by Distilling Feature Correspondences". In: *International Conference on Learning Representations (ICLR)* (cit. on p. 67).

Haramati, Dan, Tal Daniel, and Aviv Tamar (2024). "Entity-Centric Reinforcement Learning for Object Manipulation from Pixels". In: *International Conference on Learning Representations (ICLR)* (cit. on pp. 47, 82, 88).

Haugeland, John (1985). *Artificial Intelligence: The Very Idea*. MIT Press. ISBN: 9780262081535 (cit. on p. 1).

He, Kaiming, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick (2022). "Masked Autoencoders are Scalable Vision Learners". In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on pp. 63, 66, 68, 89).

He, Kaiming, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick (2019). "Momentum Contrast for Unsupervised Visual Representation Learning". In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on p. 65).

He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun (2015). "Deep Residual Learning for Image Recognition". In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on p. 64).

Helmholtz, Hermann von (1962). *Helmholtz's Treatise on Physiological Optics*. Helmholtz's Treatise on Physiological Optics Bd. 3. Dover Publications. ISBN: 9780486600161 (cit. on p. 14).

Hendrycks, Dan and Thomas Dietterich (2019). "Benchmarking Neural Network Robustness to Common Corruptions and Perturbations". In: *International Conference on Learning Representations (ICLR)* (cit. on p. 11).

Hendrycks, Dan, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Xiaodong Song (2021). "Natural Adversarial Examples". In: (cit. on p. 11).

Heravi, Negin, Ayzaan Wahid, Corey Lynch, Peter R. Florence, Travis Armstrong, Jonathan Tompson, Pierre Sermanet, Jeannette Bohg, and Debidatta Dwibedi (2022). "Visuomotor Control in Multi-Object Scenes Using Object-Aware Representations". In: (cit. on pp. 3, 10, 11, 82, 88).

Higgins, Irina, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner (2017). "beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework". In: *International Conference on Learning Representations (ICLR)* (cit. on pp. 13, 34, 44).

Higgins, Irina, Arka Pal, Andrei A. Rusu, Loic Matthey, Christopher P Burgess, Alexander Pritzel, Matthew Botvinick, Charles Blundell, and Alexander Lerchner (2017). "DARLA: Improving Zero-Shot Transfer in Reinforcement Learning". In: *International Conference on Machine Learning (ICML)* (cit. on p. 46).

Hinton, Geoffrey (2023). "How to Represent Part-Whole Hierarchies in a Neural Network". In: *Neural Computation* 35.3, pp. 413–452. ISSN: 0899-7667 (cit. on p. 2).

Hinton, Geoffrey, Alex Krizhevsky, and Sida D. Wang (2011). "Transforming Auto-Encoders". In: *International Conference on Artificial Neural Networks (ICANN)* (cit. on p. 15).

Houthooft, Rein, Xi Chen, Xi Chen, Yan Duan, John Schulman, Filip De Turck, and Pieter Abbeel (2016). "VIME: Variational Information Maximizing Exploration". In: *Conference on Neural Information Processing Systems (NIPS)* (cit. on p. 57).

Huang, Xiao Shi, Felipe Perez, Jimmy Ba, and Maksims Volkovs (2020). "Improving Transformer Optimization Through Better Initialization". In: *International Conference on Machine Learning (ICML)* (cit. on p. 48).

Huber, William A. (2015). *Notation for possible values of a random variable*. Version from 2017-04-13. (Cit. on p. 50).

Hubert, Lawrence J. and Phipps Arabie (1985). "Comparing partitions". In: *Journal of Classification* 2, pp. 193–218 (cit. on p. 32).

Hupkes, Dieuwke, Verna Dankers, Mathijs Mul, and Elia Bruni (2020). "Compositionality Decomposed: How do Neural Networks Generalise?" In: *Journal of Artificial Intelligence Research (JAIR)* 67, pp. 757–795 (cit. on p. 10).

Hwang, Inwoo, Yunhyeok Kwak, Suhyung Choi, Byoung-Tak Zhang, and Sanghack Lee (2024). "Fine-Grained Causal Dynamics Learning with Quantization for Improving Robustness in Reinforcement Learning". In: *International Conference on Machine Learning (ICML)* (cit. on pp. 58, 59).

Hwang, Inwoo, Yunhyeok Kwak, Yeon-Ji Song, Byoung-Tak Zhang, and Sanghack Lee (2023). "On Discovery of Local Independence over Continuous Variables via Neural Contextual Decomposition". In: *Conference on Causal Learning and Reasoning (CLeaR)* (cit. on pp. 58, 59).

Isola, Phillip, Daniel Zoran, Dilip Krishnan, and Edward H. Adelson (2014). "Crisp Boundary Detection Using Pointwise Mutual Information". In: *European Conference on Computer Vision (ECCV)* (cit. on p. 16).

Jabri, Allan, Sjoerd van Steenkiste, Emiel Hoogeboom, Mehdi S. M. Sajjadi, and Thomas Kipf (2024). "DORSal: Diffusion for Object-centric Representations of Scenes et al". In: *International Conference on Learning Representations (ICLR)* (cit. on pp. 3, 86).

Jaderberg, Max, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu (2015). "Spatial Transformer Networks". In: *Conference on Neural Information Processing Systems (NIPS)* (cit. on pp. 21, 30).

Ji, Xu, João F. Henriques, and Andrea Vedaldi (2019). "Invariant Information Clustering for Unsupervised Image Classification and Segmentation". In: *IEEE International Conference on Computer Vision (ICCV)* (cit. on p. 16).

Jiang, Jindong, Fei Deng, Gautam Singh, and Sungjin Ahn (2023). "Object-Centric Slot Diffusion". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on pp. 3, 24, 35, 69).

Jiang, Jindong, Sepehr Janghorbani, Gerard de Melo, and Sungjin Ahn (2020). "Scalable Object-Oriented Sequential Generative Models". In: *International Conference on Learning Representations (ICLR)* (cit. on pp. 3, 17, 19–21, 23, 25, 29, 30, 41, 72).

Johnson-Laird, Philip N. (2010). "Mental models and human reasoning". In: *Proceedings of the National Academy of Sciences* 107.43, pp. 18243–18250 (cit. on p. 1).

Jung, Whie, Jaehoon Yoo, Sungjin Ahn, and Seunghoon Hong (2024). "Learning to Compose: Improving Object Centric Learning by Injecting Compositionality". In: *International Conference on Learning Representations (ICLR)* (cit. on p. 25).

Kabra, Rishabh, Daniel Zoran, Goker Erdogan, Loic Matthey, Antonia Creswell, Matthew Botvinick, Alexander Lerchner, and Christopher P. Burgess (2021). "SIMONe: View-Invariant, Temporally-Abstracted Object Representations via Unsupervised Video Decomposition". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on p. 23).

Kahneman, Daniel and Dale T. Miller (1986). "Norm theory: Comparing reality to its alternatives". In: *Psychological Review* 93, pp. 136–153 (cit. on p. 52).

Kakogeorgiou, Ioannis, Spyros Gidaris, Konstantinos Karantzalos, and Nikos Komodakis (2024). "SPOT: Self-Training with Patch-Order Permutation for Object-Centric Learning with Autoregressive Transformers". In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on pp. 68, 83).

Karazija, Laurynas, Iro Laina, and Christian Rupprecht (2021). "ClevrTex: A Texture-Rich Benchmark for Unsupervised Multi-Object Segmentation". In: *NeurIPS Track on Datasets and Benchmarks* (cit. on pp. 32–34).

Karl, Maximilian, Maximilian Soelch, Justin Bayer, and Patrick van der Smagt (2017). "Deep Variational Bayes Filters: Unsupervised Learning of State Space Models from Raw Data". In: *International Conference on Learning Representations (ICLR)* (cit. on p. 77).

Kim, Dongwon, Namyup Kim, and Suha Kwak (2023). "Improving Cross-Modal Retrieval with Set of Diverse Embeddings". In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on p. 3).

Kim, Dongwon, Namyup Kim, Cuiling Lan, and Suha Kwak (2023). "Shatter and Gather: Learning Referring Image Segmentation with Text Supervision". In: *IEEE International Conference on Computer Vision (ICCV)* (cit. on pp. 11, 16, 69, 82, 83, 87).

Kim, Jinwoo, Janghyuk Choi, Ho-Jin Choi, and Seon Joo Kim (2023). "Shepherding Slots to Objects: Towards Stable and Robust Object-Centric Learning". In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on pp. 16, 24).

Kingma, Diederik P and Max Welling (2014). "Auto-encoding Variational Bayes". In: *International Conference on Learning Representations (ICLR)* (cit. on pp. 24, 41, 85).

Kipf, Thomas, Gamaleldin F Elsayed, Aravindh Mahendran, Austin Stone, Sara Sabour, Georg Heigold, Rico Jonschkowski, Alexey Dosovitskiy, and Klaus Greff (2022). "Conditional Object-Centric Learning from Video". In: *International Conference on Learning Representations (ICLR)* (cit. on pp. 2, 3, 16, 17, 19, 20, 23, 25, 28, 33, 64, 65, 71–74).

Kipf, Thomas, Elise van der Pol, and Max Welling (2020). "Contrastive Learning of Structured World Models". In: *International Conference on Learning Representations* (cit. on pp. 3, 16, 25, 77).

Kirillov, Alexander, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick (2023). "Segment Anything". In: *IEEE International Conference on Computer Vision (ICCV)* (cit. on pp. 88, 89).

Kobayashi, Seijin, Simon Schug, Yassir Akram, Florian Redhardt, Johannes von Oswald, Razvan Pascanu, Guillaume Lajoie, and João Sacramento (2024). "When can transformers compositionally generalize in-context?" In: *ICML 2024 Workshop on Next Generation of Sequence Modeling Architectures* (cit. on pp. 9, 10, 91).

Koffka, Kurt (2013). *Principles of Gestalt Psychology*. Cognitive psychology. Routledge. ISBN: 9780415209625 (cit. on p. 21).

Koller, Daphne and Nir Friedman (2009). *Probabilistic Graphical Models: Principles and Techniques - Adaptive Computation and Machine Learning*. The MIT Press. ISBN: 0262013193 (cit. on p. 8).

Kori, Avinash, Francesco Locatello, Fabio De Sousa Ribeiro, Francesca Toni, and Ben Glocker (2024). "Grounded Object-Centric Learning". In: *International Conference on Learning Representations (ICLR)* (cit. on p. 15).

Kosiorek, Adam R., Hyunjik Kim, Yee Whye Teh, and Ingmar Posner (2018). "Sequential Attend, Infer, Repeat: Generative Modelling of Moving Objects". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on pp. 20, 21, 23, 29, 72).

Kossen, Jannik, Karl Stelzner, Marcel Hussing, Claas Voelcker, and Kristian Kersting (2020). "Structured Object-Aware Physics Prediction for Video Modeling and Planning". In: *International Conference on Learning Representations (ICLR)* (cit. on p. 3).

Kowal, Matthew, Achal Dave, Rares Ambrus, Adrien Gaidon, Konstantinos G Derpanis, and Pavel Tokmakov (2024). "Understanding Video Transformers via Universal Concept Discovery". In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on p. 91).

Kuhn, Harold W. (1955). "The Hungarian Method for the Assignment Problem". In: *Naval Research Logistics Quarterly* (cit. on p. 32).

Lachapelle, Sébastien, Pau Rodríguez López, Yash Sharma, Katie Everett, Rémi Le Priol, Alexandre Lacoste, and Simon Lacoste-Julien (2022). "Disentanglement via Mechanism Sparsity Regularization: A New Principle for Nonlinear ICA". In: *Conference on Causal Learning and Reasoning (CLeaR)* (cit. on pp. 58, 59).

Lake, Brenden M., Tomer D. Ullman, Joshua B. Tenenbaum, and Samuel J. Gershman (2017). "Building machines that learn and think like people". In: *Behavioral and Brain Sciences* 40 (cit. on p. 2).

Laversanne-Finot, Adrien, Alexandre Pere, and Pierre-Yves Oudeyer (2018). "Curiosity Driven Exploration of Learned Disentangled Goal Spaces". In: *Conference on Robot Learning (CoRL)* (cit. on p. 40).

LeCun, Yann (2022). *A Path Towards Autonomous Machine Intelligence* (cit. on p. 9).

Lepori, Michael A., Thomas Serre, and Ellie Pavlick (2023). "Break It Down: Evidence for Structural Compositionality in Neural Networks". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on p. 91).

Li, Richard, Allan Jabri, Trevor Darrell, and Pulkit Agrawal (2019). "Towards Practical Multi-Object Manipulation using Relational Reinforcement Learning". In: *IEEE International Conference on Robotics and Automation (ICRA)* (cit. on p. 47).

Li, Xiaodan, Yuefeng Chen, Yao Zhu, Shuhui Wang, Rong Zhang, and Hui Xue (2023). "ImageNet-E: Benchmarking Neural Network Robustness via Attribute Editing". In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on p. 11).

Lillicrap, Timothy P., Jonathan J. Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra (2016). "Continuous control with deep reinforcement learning". In: *International Conference on Learning Representations (ICLR)* (cit. on p. 56).

Lin, Tsung, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick (2014). "Microsoft COCO: Common Objects in Context". In: *European Conference on Computer Vision (ECCV)* (cit. on pp. 4, 63).

Lin, Zhixuan, Yi-Wu Fu, Skand Vishwanath Peri, Weihao Sun, Gautam Singh, Fei Deng, Jindong Jiang, and Sungjin Ahn (2020). "SPACE: Unsupervised Object-Oriented Scene Representation via Spatial Attention and Decomposition". In: *International Conference on Learning Representations (ICLR)* (cit. on pp. 2, 20, 22).

Lindley, David (1956). "On a Measure of the Information Provided by an Experiment". In: *Annals of Mathematical Statistics* 27, pp. 986–1005 (cit. on p. 58).

Lippe, Phillip, Sara Magliacane, Sindy Löwe, Yuki M. Asano, Taco Cohen, and Efstratios Gavves (2022). "CITRIS: Causal Identifiability from Temporal Intervened Sequences". In: *International Conference on Machine Learning (ICML)* (cit. on pp. 58, 59).

Lippe, Phillip, Sara Magliacane, Sindy Löwe, Yuki M. Asano, Taco Cohen, and Efstratios Gavves (2023). "Causal Representation Learning for Instantaneous and Temporal Effects in Interactive Systems". In: *International Conference on Learning Representations (ICLR)* (cit. on pp. 58, 59).

Locatello, Francesco, Stefan Bauer, Mario Lucic, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem (2018). "Challenging Common Assumptions in the Unsupervised Learning of Disentangled Representations". In: *International Conference on Machine Learning (ICML)* (cit. on p. 34).

Locatello, Francesco, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf (2020). "Object-Centric Learning with Slot Attention". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on pp. 2, 3, 12, 19, 20, 22–27, 63, 64, 66, 70).

Löwe, Sindy, Phillip Lippe, Francesco Locatello, and Max Welling (2023). "Rotating Features for Object Discovery". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on pp. 19, 69).

Löwe, Sindy, Phillip Lippe, Maja R. Rudolph, and Max Welling (2022). "Complex-Valued Autoencoders for Object Discovery". In: *Transactions on Machine Learning Research (TMLR)* (cit. on p. 19).

Ma, Yecheng Jason, William Liang, Vaidehi Som, Vikash Kumar, Amy Zhang, Osbert Bastani, and Dinesh Jayaraman (2023). "LIV: Language-Image Representations and Rewards for Robotic Control". In: *International Conference on Machine Learning (ICML)* (cit. on p. 88).

Majumdar, Arjun et al. (2024). "OpenEQA: Embodied Question Answering in the Era of Foundation Models". In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on p. 9).

Mamaghan, Amir Mohammad Karimi, Samuele Papa, Karl Henrik Johansson, Stefan Bauer, and Andrea Dittadi (2024). "Exploring the Effectiveness of Object-Centric Representations in Visual Question Answering: Comparative Insights with Foundation Models". In: *arXiv:2407.15589* (cit. on pp. 3, 11, 35, 69, 82, 83, 87).

Mambelli, Davide, Frederik Träuble, Stefan Bauer, Bernhard Scholkopf, and Francesco Locatello (2022). "Compositional Multi-Object Reinforcement Learning with Linear Relation Networks". In: *ICLR 2022 Workshop on Objects, Structure and Causality* (cit. on p. 48).

Minsky, Marvin (1961). "Steps toward Artificial Intelligence". In: *Proceedings of the IRE* 49.1, pp. 8–30 (cit. on p. 59).

Mitchell, Tom M. (1980). "The Need for Biases in Learning Generalizations". In: *Readings in Machine Learning*. Ed. by Jude W. Shavlik and Thomas G. Dietterich. Morgan Kauffman, pp. 184–191 (cit. on pp. 2, 8).

Mnih, Volodymyr, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller (2013). "Playing Atari with Deep Reinforcement Learning". In: *NIPS Deep Learning Workshop* (cit. on p. 56).

Mondal, Shanka Subhra, Taylor W. Webb, and Jonathan Cohen (2023). "Learning to reason over visual objects". In: *International Conference on Learning Representations (ICLR)* (cit. on pp. 3, 87).

Nair, Ashvin, Shikhar Bahl, Alexander Khazatsky, Vitchyr Pong, G. Berseth, and S. Levine (2019). "Contextual Imagined Goals for Self-Supervised Robotic Learning". In: *Conference on Robot Learning (CoRL)* (cit. on p. 40).

Nair, Ashvin, Vitchyr Pong, Murtaza Dalal, Shikhar Bahl, Steven Lin, and Sergey Levine (2018). "Visual reinforcement learning with imagined goals". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on pp. 40, 43, 44, 46).

Nair, Suraj, Aravind Rajeswaran, Vikash Kumar, Chelsea Finn, and Abhinav Gupta (2022). "R3M: A Universal Visual Representation for Robot Manipulation". In: *Conference on Robot Learning (CoRL)* (cit. on pp. 48, 88).

Nanbo, Li and Robert B. Fisher (2021). "Duplicate Latent Representation Suppression for Multi-object Variational Autoencoders". In: *British Machine Vision Conference (BMVC)* (cit. on p. 24).

Newell, Allen and Herbert A. Simon (1976). "Computer science as empirical inquiry: symbols and search". In: *Communications of the ACM* 19.3, pp. 113–126. ISSN: 0001-0782 (cit. on p. 1).

Olah, Chris, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter (2020). "Zoom In: An Introduction to Circuits". In: *Distill* (cit. on p. 11).

Oord, Aaron van den, Yazhe Li, and Oriol Vinyals (2019). "Representation Learning with Contrastive Predictive Coding". In: *arXiv:1807.03748* (cit. on pp. 16, 91).

Oord, Aäron van den, Oriol Vinyals, and Koray Kavukcuoglu (2017). "Neural Discrete Representation Learning". In: *Conference on Neural Information Processing Systems (NIPS)* (cit. on p. 65).

Open X-Embodiment Collaboration et al. (2023). "Open X-Embodiment: Robotic Learning Datasets and RT-X Models". In: *arXiv:2310.08864* (cit. on p. 88).

OpenAI (2024). "GPT-4 Technical Report". In: *arXiv:2303.08774* (cit. on pp. 9, 89).

OpenAI, Matthias Plappert, Raul Sampedro, Tao Xu, Ilge Akkaya, Vineet Kosaraju, Peter Welinder, Ruben D'Sa, Arthur Petron, Henrique P. d. O. Pinto, Alex Paino, Hyeonwoo Noh, Lilian Weng, Qiming Yuan, Casey Chu, and Wojciech Zaremba (2021). "Asymmetric self-play for automatic goal discovery in robotic manipulation". In: *arXiv:2101.04882* (cit. on p. 47).

Oquab, Maxime et al. (2023). "DINOv2: Learning Robust Visual Features without Supervision". In: *Transactions on Machine Learning Research (TMLR)* (cit. on pp. 64, 69, 86, 89).

Papa, Samuele, Ole Winther, and Andrea Dittadi (2022). "Inductive Biases for Object-Centric Representations in the Presence of Complex Textures". In: *UAI 2022: Workshop on Causal Representation Learning* (cit. on p. 15).

Pathak, Deepak, Dhiraj Gandhi, and Abhinav Gupta (2019). "Self-Supervised Exploration via Disagreement". In: *International Conference on Machine Learning (ICML)* (cit. on p. 57).

Patterson, Genevieve and James Hays (2016). "COCO Attributes: Attributes for People, Animals, and Objects". In: *European Conference on Computer Vision (ECCV)* (cit. on p. 34).

Pearl, Judea (2009). *Causality: Models, Reasoning and Inference*. 2nd. USA: Cambridge University Press. ISBN: 9780511803161 (cit. on pp. 11, 51, 58).

Peters, Jonas, Dominik Janzing, and Bernhard Schölkopf (2017). *Elements of Causal Inference: Foundations and Learning Algorithms*. Cambridge, MA, USA: MIT Press. ISBN: 978-0-262-03731-0 (cit. on pp. 11, 51).

Pitis, Silviu, Elliot Creager, and Animesh Garg (2020). "Counterfactual Data Augmentation using Locally Factored Dynamics". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on pp. 58, 59).

Pitis, Silviu, Elliot Creager, Ajay Mandlekar, and Animesh Garg (2022). "MoCoDA: Model-based Counterfactual Data Augmentation". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on p. 58).

Plappert, Matthias, Marcin Andrychowicz, Alex Ray, Bob McGrew, Bowen Baker, Glenn Powell, J. Schneider, Joshua Tobin, Maciek Chociej, P. Welinder, V. Kumar, and

W. Zaremba (2018). "Multi-Goal Reinforcement Learning: Challenging Robotics Environments and Request for Research". In: *arXiv:1802.09464* (cit. on pp. 54, 57).

Pomerantz, James and Michael Kubovy (1986). "Theoretical approaches to perceptual organization: Simplicity and likelihood principles". In: John Wiley & Sons (cit. on p. 14).

Pong, Vitchyr, Murtaza Dalal, Steven Lin, Ashvin Nair, Shikhar Bahl, and Sergey Levine (2020). "Skew-fit: State-covering self-supervised reinforcement learning". In: *International Conference on Machine Learning (ICML)* (cit. on pp. 40, 43, 46).

Pont-Tuset, Jordi, Pablo Arbeláez, Jonathan T. Barron, Ferran Marques, and Jitendra Malik (2017). "Multiscale Combinatorial Grouping for Image Segmentation and Object Proposal Generation". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 39.1 (cit. on p. 32).

Pont-Tuset, Jordi, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alexander Sorkine-Hornung, and Luc Van Gool (2017). "The 2017 DAVIS Challenge on Video Object Segmentation". In: *arXiv:1704.00675* (cit. on pp. 4, 74).

Poole, Ben, Sherjil Ozair, Aaron Van Den Oord, Alex Alemi, and George Tucker (2019). "On Variational Bounds of Mutual Information". In: *International Conference on Machine Learning (ICML)* (cit. on p. 54).

Porter, Thomas and Tom Duff (1984). "Compositing digital images". In: *SIGGRAPH Computer Graphics* 18.3, pp. 253–259. ISSN: 0097-8930 (cit. on p. 27).

Qi, Lu, Jason Kuen, Tiancheng Shen, Jiuxiang Gu, Weidong Guo, Jiaya Jia, Zhe Lin, and Ming-Hsuan Yang (2023). "High-Quality Entity Segmentation". In: *IEEE International Conference on Computer Vision (ICCV)* (cit. on p. 86).

Qian, Jianing, Anastasios Panagopoulos, and Dinesh Jayaraman (2024). "Recasting Generic Pretrained Vision Transformers As Object-Centric Scene Encoders For Manipulation Policies". In: *IEEE International Conference on Robotics and Automation (ICRA)* (cit. on p. 88).

Qian, Rui, Shuangrui Ding, Xian Liu, and Dahua Lin (2023). "Semantics Meets Temporal Correspondence: Self-supervised Object-centric Learning in Videos". In: (cit. on pp. 25, 76).

Radford, Alec, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever (2021). "Learning Transferable Visual Models From Natural Language Supervision". In: *International Conference on Machine Learning (ICML)* (cit. on pp. 69, 87, 89).

Radosavovic, Ilija, Tete Xiao, Stephen James, P. Abbeel, Jitendra Malik, and Trevor Darrell (2022). "Real-World Robot Learning with Masked Visual Pre-training". In: *Conference on Robot Learning (CoRL)* (cit. on p. 88).

Rand, William M. (1971). "Objective Criteria for the Evaluation of Clustering Methods". In: *Journal of the American Statistical Association* 66.336, pp. 846–850 (cit. on p. 32).

Reddy, Pradyumna, Scott Wisdom, Klaus Greff, John R. Hershey, and Thomas Kipf (2023). "AudioSlots: A slot-centric generative model for audio separation". In: *Self-supervision in Audio, Speech and Beyond (SASB) Workshop at ICASSP 2023* (cit. on p. 26).

Romijnders, Rob, Aravindh Mahendran, Michael Tschannen, Josip Djolonga, Marvin Ritter, Neil Houlsby, and Mario Lucic (2021). "Representation learning from videos in-the-wild: An object-centric approach". In: *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)* (cit. on p. 10).

Royer, Loic A., David L. Richmond, Carsten Rother, Bjoern Andres, and Dagmar Kainmueller (2016). "Convexity Shape Constraints for Image Segmentation". In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on p. 17).

Russell, Stuart and Peter Norvig (2020). *Artificial Intelligence: A Modern Approach*. Pearson. ISBN: 9780134610993 (cit. on p. 1).

Sajjadi, Mehdi SM, Daniel Duckworth, Aravindh Mahendran, Sjoerd van Steenkiste, Filip Pavetic, Mario Lucic, Leonidas J Guibas, Klaus Greff, and Thomas Kipf (2022). "Object Scene Representation Transformer". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on pp. 72, 89).

Salge, Christoph, Cornelius Glackin, and Daniel Polani (2014). "Empowerment–An Introduction". In: *Guided Self-Organization: Inception*. Springer, Berlin, Heidelberg. ISBN: 978-3-642-53734-9 (cit. on p. 56).

Sancaktar, Cansu, Sebastian Blaes, and Georg Martius (2022). "Curious Exploration via Structured World Models Yields Zero-Shot Object Manipulation". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on pp. 47, 88).

Sancaktar, Cansu, Justus Piater, and Georg Martius (2023). "Regularity as Intrinsic Reward for Free Play". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on pp. 85, 88).

Santoro, Adam, Felix Hill, David G. T. Barrett, Ari S. Morcos, and Timothy P. Lillicrap (2018). "Measuring abstract reasoning in neural networks". In: *International Conference on Machine Learning (ICML)* (cit. on p. 10).

Schaul, Tom, John Quan, Ioannis Antonoglou, and David Silver (2016). "Prioritized Experience Replay". In: *International Conference on Learning Representations (ICLR)* (cit. on pp. 56, 57).

Schmidhuber, Jürgen (1991). "A Possibility for Implementing Curiosity and Boredom in Model-Building Neural Controllers". In: *From Animals to Animats: Proceedings of the First International Conference on Simulation of Adaptive Behavior* (cit. on p. 55).

Schmidhuber, Jürgen (2010). "Formal Theory of Creativity, Fun, and Intrinsic Motivation (1990–2010)". In: *IEEE Transactions on Autonomous Mental Development* (cit. on p. 55).

Schölkopf, Bernhard (2022). "Causality for Machine Learning". In: *Probabilistic and Causal Inference: The Works of Judea Pearl*. 1st ed. New York, NY, USA: Association for Computing Machinery. ISBN: 9781450395861 (cit. on pp. 11, 50).

Schölkopf, Bernhard, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio (2021). "Towards Causal Representation Learning". In: *Proceedings of the IEEE* 109.5 (cit. on pp. 2, 9, 11).

Schuhmann, Christoph, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev (2022). "LAION-5B: An open large-scale dataset for training next generation image-text models". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on p. 85).

Schwarzer, Max, Ankesh Anand, Rishab Goel, R Devon Hjelm, Aaron Courville, and Philip Bachman (2021). "Data-Efficient Reinforcement Learning with Self-Predictive Representations". In: *International Conference on Learning Representations (ICLR)* (cit. on p. 48).

Schwarzer, Max, Johan Samir Obando Ceron, Aaron Courville, Marc G Bellemare, Rishabh Agarwal, and Pablo Samuel Castro (2023). "Bigger, Better, Faster: Human-level Atari with human-level efficiency". In: *International Conference on Machine Learning (ICML)* (cit. on p. 48).

Seitzer, Maximilian, Max Horn, Andrii Zadaianchuk, Dominik Zietlow, Tianjun Xiao, Carl-Johann Simon-Gabriel, Tong He, Zheng Zhang, Bernhard Schölkopf, Thomas Brox, and Francesco Locatello (2023). "Bridging the Gap to Real-World Object-Centric Learning". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on pp. 4, 20, 33, 64, 66, 67, 69, 70, 84, 145).

Seitzer, Maximilian, Bernhard Schölkopf, and Georg Martius (2021). "Causal Influence Detection for Improving Efficiency in Reinforcement Learning". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on pp. 4, 50, 52–55, 57, 129).

Seitzer, Maximilian, Sjoerd van Steenkiste, Thomas Kipf, Klaus Greff, and Mehdi S. M. Sajjadi (2024). "DyST: Towards Dynamic Neural Scene Representations on Real-World Videos". In: *International Conference on Learning Representations (ICLR)* (cit. on pp. 77, 89).

Shi, Junyao, Jianing Qian, Yecheng Jason Ma, and Dinesh Jayaraman (2024). "Composing Pre-Trained Object-Centric Representations for Robotics From "What" and "Where" Foundation Models". In: *IEEE International Conference on Robotics and Automation (ICRA)* (cit. on p. 88).

Shwartz Ziv, Ravid and Yann LeCun (2024). "To Compress or Not to Compress - Self-Supervised Learning and Information Theory: A Review". In: *Entropy* 26.3. ISSN: 1099-4300 (cit. on p. 16).

Singh, Gautam, Fei Deng, and Sungjin Ahn (2022). "Illiterate DALL-E Learns to Compose". In: *International Conference on Learning Representations (ICLR)* (cit. on pp. 2, 3, 20, 25, 64, 65, 67, 68).

Singh, Gautam, Yi-Fu Wu, and Sungjin Ahn (2022). "Simple Unsupervised Object-Centric Learning for Complex and Naturalistic Videos". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on pp. 20, 72, 74).

Smolensky, Paul, Richard McCoy, Roland Fernandez, Matthew Goldrick, and Jianfeng Gao (2022). "Neurocompositional Computing: From the Central Paradox of Cognition to a New Generation of AI Systems". In: *AI Magazine* 43.3, pp. 308–322 (cit. on p. 2).

Spelke, Elizabeth S. and Katherine D. Kinzler (2007). "Core knowledge". In: *Developmental science* 10.1, pp. 89–96 (cit. on p. 1).

Stanić, Aleksandar, Anand Gopalakrishnan, Kazuki Irie, and Jürgen Schmidhuber (2023). "Contrastive Training of Complex-Valued Autoencoders for Object Discovery". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on p. 19).

Stanić, Aleksandar, Yujin Tang, David Ha, and Jürgen Schmidhuber (2023). "Learning to Generalize with Object-centric Agents in the Open World Survival Game Crafter". In: *IEEE Transactions on Games* (cit. on pp. 10, 82).

Steenkiste, Sjoerd van, Michael Chang, Klaus Greff, and Jürgen Schmidhuber (2018). "Relational Neural Expectation Maximization: Unsupervised Discovery of Objects and their Interactions". In: *International Conference on Learning Representations (ICLR)* (cit. on pp. 2, 25).

Stelzner, Karl, Kristian Kersting, and Adam R. Kosiorek (2021). "Decomposing 3D Scenes into Objects via Unsupervised Volume Segmentation". In: *arXiv:2104.01148* (cit. on p. 89).

Stone, Austin, Daniel Maurer, Alper Ayvaci, Anelia Angelova, and Rico Jonschkowski (2021). "SMURF: Self-Teaching Multi-Frame Unsupervised RAFT with Full-Image Warping". In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on p. 74).

Sun, Chen, Calvin Luo, Xingyi Zhou, Anurag Arnab, and Cordelia Schmid (2023). "Does Visual Pretraining Help End-to-End Reasoning?" In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on p. 91).

Sutton, Richard S. (1991). "Dyna, an integrated architecture for learning, planning, and reacting". In: *ACM SIGART Bulletin* (cit. on p. 77).

Sutton, Richard S. (2019). *The Bitter Lesson*. (Cit. on pp. 9, 90).

Sutton, Richard S. and Andrew G. Barto (2018). *Reinforcement Learning: An Introduction*. 2nd ed. Cambridge, MA, USA: MIT Press. ISBN: 9780262039246 (cit. on pp. 39, 51, 57, 59).

Tangemann, Matthias, Steffen Schneider, Julius von Kügelgen, Francesco Locatello, Peter Gehler V., Thomas Brox, Matthias Kümmerer, Matthias Bethge, and Bernhard Schölkopf (2023). "Unsupervised Object Learning via Common Fate". In: *Conference on Causal Learning and Reasoning (CLeaR)* (cit. on pp. 17, 64, 73).

Tigas, Panagiotis, Yashas Annadani, Andrew Jesson, Bernhard Schölkopf, Yarin Gal, and Stefan Bauer (2022). "Interventions, Where and How? Experimental Design for Causal Models at Scale". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on p. 58).

Todorov, E., T. Erez, and Y. Tassa (2012). "MuJoCo: A physics engine for model-based control". In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (cit. on p. 44).

Tong, Shengbang, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie (2024). "Eyes Wide Shut? Exploring the Visual Shortcomings of Multimodal LLMs". In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on p. 9).

Touchette, Hugo and Seth Lloyd (2004). "Information-theoretic approach to the study of control systems". In: *Physica A: Statistical Mechanics and its Applications* 331.1, pp. 140–172. ISSN: 0378-4371 (cit. on p. 53).

Traub, Manuel, Frederic Becker, Adrian Sauter, Sebastian Otte, and Martin V. Butz (2024). "Loci-Segmented: Improving Scene Segmentation Learning". In: *International Conference on Artificial Neural Networks (ICANN)* (cit. on pp. 22, 65).

Traub, Manuel, Sebastian Otte, Tobias Menge, Matthias Karlbauer, Jannik Thuemmel, and Martin V. Butz (2023). "Learning What and Where: Disentangling Location and Identity Tracking Without Supervision". In: *International Conference on Learning Representations (ICLR)* (cit. on pp. 16, 20–22, 72).

Tsividis, Pedro, Thomas Pouncy, Jaqueline L. Xu, Joshua B. Tenenbaum, and Samuel J. Gershman (2017). "Human Learning in Atari". In: *AAAI Spring Symposia* (cit. on p. 10).

Tung, Yi-Shiuan, Himanshu Gupta, Wei Jiang, Bradley Hayes, and Alessandro Roncone (2024). "Causal Influence Detection for Human Robot Interaction". In: *HRI'24: Workshop on Causal Learning for Human-Robot Interaction (Causal-HRI)* (cit. on p. 58).

Udandarao, Vishaal, Ameya Prabhu, Adhiraj Ghosh, Yash Sharma, Philip H. S. Torr, Adel Bibi, Samuel Albanie, and Matthias Bethge (2024). "No "Zero-Shot" Without Exponential Data: Pretraining Concept Frequency Determines Multimodal Model Performance". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on p. 10).

Uijlings, J.R.R., K.E.A. van de Sande, T. Gevers, and A. W. M. Smeulders (2013). "Selective Search for Object Recognition". In: *International Journal of Computer Vision* 104, pp. 154–171 (cit. on p. 32).

Urpí, Núria Armengol, Marco Bagatella, Marin Vlastelica, and Georg Martius (2024). "Causal Action Influence Aware Counterfactual Data Augmentation". In: *International Conference on Machine Learning (ICML)* (cit. on pp. 58, 82).

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin (2017). "Attention is All you Need". In: *Conference on Neural Information Processing Systems (NIPS)* (cit. on pp. 9, 21, 26, 28, 42).

Veerapaneni, Rishi, John D. Co-Reyes, Michael Chang, Michael Janner, Chelsea Finn, Jiajun Wu, Joshua B. Tenenbaum, and Sergey Levine (2019). "Entity Abstraction in Visual Model-Based Reinforcement Learning". In: *Conference on Robot Learning (CoRL)* (cit. on pp. 3, 20, 23, 35).

Vinh, Nguyen Xuan, Julien Epps, and James Bailey (2010). "Information Theoretic Measures for Clusterings Comparison: Variants, Properties, Normalization and Correction for Chance". In: *Journal of Machine Learning Research (JMLR)* 11.95, pp. 2837–2854 (cit. on p. 32).

Vowels, Matthew J., Necati Cihan Camgoz, and Richard Bowden (2022). "D'ya Like DAGs? A Survey on Structure Learning and Causal Discovery". In: *ACM Computing Surveys* 55.4. ISSN: 0360-0300 (cit. on p. 58).

Wang, Shunxin, Raymond N. J. Veldhuis, Christoph Brune, and Nicola Strisciuglio (2023). "A Survey on the Robustness of Computer Vision Models against Common Corruptions". In: *arXiv:2305.06024* (cit. on p. 11).

Wang, Tongzhou and Phillip Isola (2022). "Understanding Contrastive Representation Learning through Alignment and Uniformity on the Hypersphere". In: *International Conference on Machine Learning (ICML)* (cit. on p. 91).

Wang, Xinlong, Zhiding Yu, Shalini De Mello, Jan Kautz, Anima Anandkumar, Chunhua Shen, and Jose M. Alvarez (2022). "FreeSOLO: Learning to Segment Objects without Annotations". In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on p. 89).

Wang, Xudong, Rohit Girdhar, Stella X Yu, and Ishan Misra (2023). "Cut and Learn for Unsupervised Object Detection and Instance Segmentation". In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on p. 89).

Wang, Xudong, Jingfeng Yang, and Trevor Darrell (2024). "Segment Anything without Supervision". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on p. 89).

Wang, Yanbo, Letao Liu, and Justin Dauwels (2023). "Slot-VAE: Object-Centric Scene Generation with Slot Attention". In: *International Conference on Machine Learning (ICML)* (cit. on pp. 24, 26, 35).

Wang, Yangtao, Xi Shen, Shell Xu Hu, Yuan Yuan, James L. Crowley, and Dominique Vaufreydaz (2022). "Self-Supervised Transformers for Unsupervised Object Discovery Using Normalized Cut". In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on p. 67).

Wang, Ziyu, Mike Zheng Shou, and Mengmi Zhang (2023). "Object-centric Learning with Cyclic Walks between Parts and Whole". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on pp. 25, 69).

Wang, Zizhao, Jiaheng Hu, Peter Stone, and Roberto Martín-Martín (2023). "ELDEN: Exploration via Local Dependencies". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on pp. 58, 59, 82).

Wang, Zizhao, Xuesu Xiao, Zifan Xu, Yuke Zhu, and Peter Stone (2022). "Causal Dynamics Learning for Task-Independent State Abstraction". In: *International Conference on Machine Learning (ICML)* (cit. on p. 59).

Warde-Farley, David, Tom Van de Wiele, Tejas Kulkarni, Catalin Ionescu, Steven Hansen, and Volodymyr Mnih (2019). "Unsupervised Control Through Non-Parametric Discriminative Rewards". In: *International Conference on Learning Representations (ICLR)* (cit. on p. 40).

Watters, Nick, Loic Matthey, Chris P. Burgess, and Alexander Lerchner (2019). "Spatial Broadcast Decoder: A Simple Architecture for Disentangled Representations in VAEs". In: *ICLR Learning from Limited Labeled Data Workshop* (cit. on pp. 27, 68).

Webb, Taylor W., Shanka Subhra Mondal, and Jonathan D. Cohen (2023). "Systematic Visual Reasoning through Object-Centric Relational Abstraction". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on pp. 3, 87).

Weinzaepfel, Philippe, Thomas Lucas, Diane Larlus, and Yannis Kalantidis (2022). "Learning Super-Features for Image Retrieval". In: *International Conference on Learning Representations (ICLR)* (cit. on p. 3).

Weis, Marissa A, Kashyap Chitta, Yash Sharma, Wieland Brendel, Matthias Bethge, Andreas Geiger, and Alexander S. Ecker (2021). "Benchmarking Unsupervised Object Representations for Video Sequences". In: *Journal of Machine Learning Research (JMLR)* (cit. on pp. 20, 23, 72).

Wen, Xin, Bingchen Zhao, Anlin Zheng, Xiangyu Zhang, and Xiaojuan Qi (2022). "Self-Supervised Visual Representation Learning with Semantic Grouping". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on pp. 15, 16).

Wertheimer, Max (2012). *On Perceived Motion and Figural Organization*. New English translation of two articles from 1912/1923 by Lothar Spillmann. The MIT Press. ISBN: 9780262305686 (cit. on pp. 14, 17, 73).

Wiedemer, Thaddäus, Jack Brady, Alexander Panfilov, Attila Juhos, Matthias Bethge, and Wieland Brendel (2024). "Provable Compositional Generalization for Object-Centric Learning". In: *International Conference on Learning Representations (ICLR)* (cit. on pp. 2, 10, 16, 25, 28, 35).

Wu, Yi-Fu, Klaus Greff, Gamaleldin Fathy Elsayed, Michael Curtis Mozer, Thomas Kipf, and Sjoerd van Steenkiste (2023). "Inverted-Attention Transformers can Learn Object Representations: Insights from Slot Attention". In: *NeurIPS 2023 Workshop on Causal Representation Learning* (cit. on pp. 27, 68).

Wu, Ziyi, Nikita Dvornik, Klaus Greff, Thomas Kipf, and Animesh Garg (2023). "Slot-Former: Unsupervised Visual Dynamics Simulation with Object-Centric Models". In: *International Conference on Learning Representations (ICLR)* (cit. on pp. 3, 23, 86).

Wu, Ziyi, Jingyu Hu, Wuyue Lu, Igor Gilitschenski, and Animesh Garg (2023). "SlotDiffusion: Object-Centric Generative Modeling with Diffusion Models". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on pp. 24, 33, 35, 69, 83).

Wu, Ziyi, Yulia Rubanova, Rishabh Kabra, Drew A. Hudson, Igor Gilitschenski, Yusuf Aytar, Sjoerd van Steenkiste, Kelsey R. Allen, and Thomas Kipf (2024). "Neural Assets: 3D-Aware Multi-Object Scene Synthesis with Image Diffusion Models". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on pp. 11, 86).

Xiao, Tete, Ilija Radosavovic, Trevor Darrell, and Jitendra Malik (2022). "Masked Visual Pre-training for Motor Control". In: (cit. on pp. 48, 88).

Xu, Jiaqi, Cuiling Lan, Wenxuan Xie, Xuejin Chen, and Yan Lu (2024). "Slot-VLM: Object-Event Slots for Video-Language Modeling". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on pp. 2, 3, 10, 11, 26, 69, 82, 83, 87).

Xu, Jiarui, Shalini De Mello, Sifei Liu, Wonmin Byeon, Thomas Breuel, Jan Kautz, and Xiaolong Wang (2022). "GroupViT: Semantic Segmentation Emerges from Text Supervision". In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on pp. 3, 16, 26, 64, 87).

Xu, Zhuoyan, Zhenmei Shi, and Yingyu Liang (2024). "Do Large Language Models Have Compositional Ability? An Investigation into Limitations and Scalability". In: (cit. on pp. 9, 10, 91).

Yang, Linjie, Yuchen Fan, Yang Fu, and Ning Xu (2021). *The 3rd Large-scale Video Object Segmentation Challenge — Video Instance Segmentation Track*. (Cit. on pp. 4, 71, 74).

Yang, Yanchao, Yutong Chen, and Stefano Soatto (2020). "Learning to Manipulate Individual Objects in an Image". In: (cit. on p. 16).

Yoon, Jaesik, Yi-Fu Wu, Heechul Bae, and Sungjin Ahn (2023). "An Investigation into Pre-Training Object-Centric Representations for Reinforcement Learning". In: *International Conference on Machine Learning (ICML)* (cit. on pp. 2, 3, 10, 34, 35, 82).

Yu, Hong-Xing, Leonidas Guibas, and Jiajun Wu (2022). "Unsupervised Discovery of Object Radiance Fields". In: *International Conference on Learning Representations (ICLR)* (cit. on p. 89).

Yuan, Jinyang, Tonglin Chen, Bin Li, and Xiangyang Xue (2023). "Compositional Scene Representation Learning via Reconstruction: A Survey". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* (cit. on pp. 3, 12, 20).

Yuksekgonul, Mert, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Y. Zou (2023). "When and why vision-language models behave like bags-of-words, and what to do about it?" In: *International Conference on Learning Representations (ICLR)* (cit. on pp. 10, 91).

Zadaianchuk, Andrii, Matthaeus Kleindessner, Yi Zhu, Francesco Locatello, and Thomas Brox (2023). "Unsupervised Semantic Segmentation with Self-supervised Object-centric Representations". In: *International Conference on Learning Representations (ICLR)* (cit. on pp. 15, 67).

Zadaianchuk, Andrii, Georg Martius, and Fanny Yang (2022). "Self-supervised Reinforcement Learning with Independently Controllable Subgoals". In: *Conference on Robot Learning (CoRL)* (cit. on pp. 47, 82).

Zadaianchuk, Andrii, Maximilian Seitzer, and Georg Martius (2021). "Self-supervised Visual Reinforcement Learning with Object-centric Representations". In: *International Conference on Learning Representations (ICLR)* (cit. on pp. 4, 41, 43–45, 115).

Zadaianchuk, Andrii, Maximilian Seitzer, and Georg Martius (2023). "Object-Centric Learning for Real-World Videos by Predicting Temporal Feature Similarities". In: *Conference on Neural Information Processing Systems (NeurIPS)* (cit. on pp. 4, 20, 69, 74–76, 161).

Zaheer, Manzil, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Russ R Salakhutdinov, and Alexander J Smola (2017). "Deep Sets". In: *Conference on Neural Information Processing Systems (NIPS)*. Vol. 30 (cit. on p. 42).

Zeng, Yan, Ruichu Cai, Fuchun Sun, Libo Huang, and Zhifeng Hao (2024). "A Survey on Causal Reinforcement Learning". In: *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–21 (cit. on pp. 4, 82).

Zhao, Rui, Yang Gao, Pieter Abbeel, Volker Tresp, and Wei Xu (2021). "Mutual Information State Intrinsic Control". In: *International Conference on Learning Representations (ICLR)* (cit. on p. 58).

Zhao, Rui and Volker Tresp (2018). "Energy-Based Hindsight Experience Prioritization". In: *Conference on Robot Learning (CoRL)* (cit. on p. 57).

Zhao, Zixu, Jiaze Wang, Max Horn, Yizhuo Ding, Tong He, Zechen Bai, Dominik Zietlow, Carl-Johann Simon-Gabriel, Bing Shuai, Zhuowen Tu, Thomas Brox, Bernt Schiele, Yanwei Fu, Francesco Locatello, Zheng Zhang, and Tianjun Xiao (2023). "Object-Centric Multiple Object Tracking". In: *IEEE International Conference on Computer Vision (ICCV)* (cit. on p. 69).

Zhou, Allan, Vikash Kumar, Chelsea Finn, and Aravind Rajeswaran (2022). "Policy Architectures for Compositional Generalization in Control". In: *NeurIPS 2022 Deep Reinforcement Learning Workshop* (cit. on p. 48).

Zhou, Yi, Hui Zhang, Hana Lee, Shuyang Sun, Pingjun Li, Yangguang Zhu, ByungIn Yoo, Xiaojuan Qi, and Jae-Joon Han (2021). "Slot-VPS: Object-centric Representation Learning for Video Panoptic Segmentation". In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on p. 3).

Zimmermann, Roland S., Sjoerd van Steenkiste, Mehdi S. M. Sajjadi, Thomas Kipf, and Klaus Greff (2023). "Sensitivity of Slot-Based Object-Centric Models to their Number of Slots". In: *arXiv:2305.18890* (cit. on p. 84).

Zoran, Daniel, Rishabh Kabra, Alexander Lerchner, and Danilo J. Rezende (2021). "PARTS: Unsupervised segmentation with slots, attention and independence maximization". In: *IEEE International Conference on Computer Vision (ICCV)* (cit. on p. 16).

Appendices

Appendix A

Self-Supervised Visual Reinforcement Learning with Object-Centric Representations

The following 13 pages have been published as:

Andrii Zadaianchuk*, **Maximilian Seitzer***, and Georg Martius (2021). "Self-Supervised Visual Reinforcement Learning with Object-Centric Representations". In: *International Conference on Learning Representations (ICLR)*. *equal contribution

A summary of this publication can be found in Chap. 4. The supplementary material to this publication is available in the online version.

SELF-SUPERVISED VISUAL REINFORCEMENT LEARNING WITH OBJECT-CENTRIC REPRESENTATIONS

Andrii Zadaianchuk[1,2]*, **Maximilian Seitzer**[1]*, **Georg Martius**[1]
[1] Max Planck Institute for Intelligent Systems, Tübingen, Germany
[2] Department of Computer Science, ETH Zurich
andrii.zadaianchuk@tuebingen.mpg.de

ABSTRACT

Autonomous agents need large repertoires of skills to act reasonably on new tasks that they have not seen before. However, acquiring these skills using only a stream of high-dimensional, unstructured, and unlabeled observations is a tricky challenge for any autonomous agent. Previous methods have used variational autoencoders to encode a scene into a low-dimensional vector that can be used as a goal for an agent to discover new skills. Nevertheless, in compositional/multi-object environments it is difficult to disentangle all the factors of variation into such a fixed-length representation of the whole scene. We propose to use *object-centric representations* as a modular and structured observation space, which is learned with a compositional generative world model. We show that the structure in the representations in combination with *goal-conditioned attention policies* helps the autonomous agent to discover and learn useful skills. These skills can be further combined to address compositional tasks like the manipulation of several different objects.
https://martius-lab.github.io/SMORL

1 INTRODUCTION

Reinforcement learning (RL) includes a promising class of algorithms that have shown capability to solve challenging tasks when those tasks are well specified by suitable reward functions. However, in the real world, people are rarely given a well-defined reward function. Indeed, humans are excellent at setting their own abstract goals and achieving them. Agents that exist persistently in the world should likewise prepare themselves to solve diverse tasks by first constructing plausible goal spaces, setting their own goals within these spaces, and then trying to achieve them. In this way, they can learn about the world around them.

In principle, the goal space for an autonomous agent could be any arbitrary function of the state space. However, when the state space is high-dimensional and unstructured, such as only images, it is desirable to have goal spaces which allow efficient exploration and learning, where the factors of variation in the environment are well disentangled. Recently, unsupervised representation learning has been proposed to learn such goal spaces (Nair et al., 2018; 2019; Pong et al., 2020). All existing methods in this context use variational autoencoders (VAEs) to map observations into a low-dimensional latent space that can later be used for sampling goals and reward shaping.

However, for complex compositional scenes consisting of multiple objects, the inductive bias of VAEs could be harmful. In contrast, representing perceptual observations in terms of entities has been shown to improve data efficiency and transfer performance on a wide range of tasks (Burgess et al., 2019). Recent research has proposed a range of methods for unsupervised scene and video decomposition (Greff et al., 2017; Kosiorek et al., 2018; Burgess et al., 2019; Greff et al., 2019; Jiang et al., 2019; Weis et al., 2020; Locatello et al., 2020). These methods learn object representations and scene decomposition jointly. The majority of them are in part motivated by the fact that the learned representations are useful for downstream tasks such as image classification, object detection, or semantic segmentation. In this work, we show that such learned representations are also beneficial for autonomous control and reinforcement learning.

[*] equal contribution

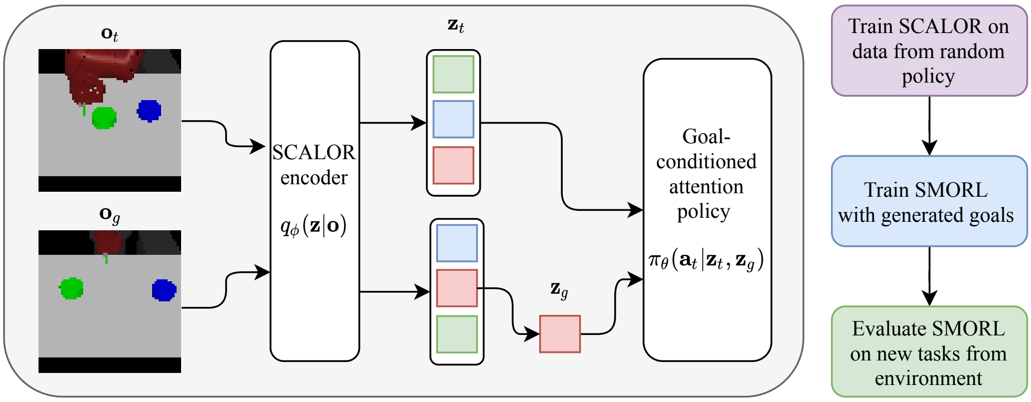

$\mathbf{o}_t$
$\mathbf{z}_t$
SCALOR encoder
$q_\phi(\mathbf{z}|\mathbf{o})$
$\mathbf{o}_g$
$\mathbf{z}_g$
Goal-conditioned attention policy
$\pi_\theta(\mathbf{a}_t|\mathbf{z}_t, \mathbf{z}_g)$
Train SCALOR on data from random policy
Train SMORL with generated goals
Evaluate SMORL on new tasks from environment

Figure 1: Our proposed SMORL architecture. Representations $\mathbf{z}_t$ are obtained from observations $\mathbf{o}_t$ through the object-centric SCALOR encoder q_ϕ, and processed by the goal-conditional attention policy $\pi_\theta(\mathbf{a}_t|\mathbf{z}_t, \mathbf{z}_g)$. During training, representations of goals are sampled conditionally on the representations of the first observation $\mathbf{z}_1$. At test time, the agent is provided with an external goal image $\mathbf{o}_g$ that is processed with the same SCALOR encoder to a set of potential goals $\{\mathbf{z}_n\}_{n=1}^{N}$. After this, the goal $\mathbf{z}_g$ is sequentially chosen from this set. This way, the agent attempts to solve all the discovered sub-tasks one-by-one, not simultaneously.

We propose to combine these *object-centric unsupervised representation* methods that represent the scene as a set of potentially structured vectors with goal-conditional visual RL. In our method (illustrated in Figure 1), dubbed SMORL (for self-supervised multi-object RL), a representation of raw sensory inputs is learned by a compositional latent variable model based on the SCALOR architecture (Jiang et al., 2019). We show that using object-centric representations simplifies the goal space learning. Autonomous agents can use those representations to learn how to achieve different goals with a reward function that utilizes the structure of the learned goal space. Our main contributions are as follows:

- We show that structured object-centric representations learned with generative world models can significantly improve the performance of the self-supervised visual RL agent.
- We develop SMORL, an algorithm that uses learned representations to autonomously discover and learn useful skills in compositional environments with several objects using only images as inputs.
- We show that even with fully disentangled ground-truth representation there is a large benefit from using SMORL in environments with complex compositional tasks such as rearranging many objects.

2 Related work

Our work lies in the intersection of several actively evolving topics: visual reinforcement learning for control and robotics, and self-supervised learning. *Vision-based RL* for robotics is able to efficiently learn a variety of behaviors such as grasping, pushing and navigation (Levine et al., 2016; Pathak et al., 2018; Levine et al., 2018; Kalashnikov et al., 2018) using only images and rewards as input signals. *Self-supervised learning* is a form of unsupervised learning where the data provides the supervision. It was successfully used to learn powerful representations for downstream tasks in natural language processing (Devlin et al., 2018; Radford et al., 2019) and computer vision (He et al., 2019; Chen et al., 2020). In the context of RL, self-supervision refers to the agent constructing its own reward signal and using it to solve self-proposed goals (Baranes & Oudeyer, 2013; Nair et al., 2018; Péré et al., 2018; Hausman et al., 2018; Lynch et al., 2019). This is especially relevant for visual RL, where a reward signal is usually not naturally available. These methods can potentially acquire a diverse repertoire of general-purpose robotic skills that can be reused and combined during test time. Such self-supervised approaches are crucial for scaling learning from narrow single-task learning to more general agents that explore the environment on their own to prepare for solving many different tasks in the future. Next, we will cover the two most related lines of research in more detail.

Self-supervised visual RL (Nair et al., 2018; 2019; Pong et al., 2020; Ghosh et al., 2019; Warde-Farley et al., 2019; Laversanne-Finot et al., 2018) tackles multi-task RL problems from images without any external reward signal. However, all previous methods assume that the environment observation can be encoded into a single vector, e.g. using VAE representations. With multiple objects being present, this assumption may result in object encodings overlapping in the representation, which is known as the binding problem (Greff et al., 2016; 2020). In addition, as the reward is also constructed based on this vector, the agent is incentivized to solve tasks that are incompatible, for instance simultaneously moving all objects to goal positions. In contrast, we suggest to learn object-centric representations and use them for reward shaping. This way, the agent can learn to solve tasks independently and then combine these skills during evaluation.

Learning object-centric representations in RL (Watters et al., 2019; van Steenkiste et al., 2019; Veerapaneni et al., 2020; Kipf et al., 2020) has been suggested to approach tasks with combinatorial and compositional elements such as the manipulation of multiple objects. However, the previous work has assumed a fixed, single task and a given reward signal, whereas we are using the learned object-representations to construct a reward signal that helps to learn useful skills that can be used to solve multiple tasks. In addition, these methods use scene-mixture models such as MONET (Burgess et al., 2019) and IODINE (Greff et al., 2019), which do not explicitly contain features like position and scale. These features can be used by the agent for more efficient sampling from the goal space and thus the explicit modeling of these features helps to create additional biases useful for manipulation tasks. However, we expect that other object-centric representations could also be successfully applied as suitable representations for RL tasks.

3 BACKGROUND

Our method combines goal-conditional RL with unsupervised object-oriented representation learning for multi-object environments. Before we describe each technique in detail, we briefly state some RL preliminaries. We consider a Markov decision process defined by $(\mathcal{S}, \mathcal{A}, p, r)$, where $\mathcal{S}$ and $\mathcal{A}$ are the continuous state and action spaces, $p\colon \mathcal{S} \times \mathcal{S} \times \mathcal{A} \mapsto [0, \infty)$ is an unknown probability density representing the probability of transitioning to state $\mathbf{s}_{t+1} \in \mathcal{S}$ from state $\mathbf{s}_t \in \mathcal{S}$ given action $\mathbf{a}_t \in \mathcal{A}$, and $r\colon \mathcal{S} \mapsto \mathbb{R}$ is a function computing the reward for reaching state $\mathbf{s}_{t+1}$. The agent's objective is to maximize the expected return $R = \sum_{t=1}^{T} \mathbb{E}_{\mathbf{s}_t \sim \rho_\pi, \mathbf{a}_t \sim \pi, \mathbf{s}_{t+1} \sim p} \left[r(\mathbf{s}_{t+1}) \right]$ over the horizon T, where $\rho_\pi(\mathbf{s}_t)$ is the state marginal distribution induced by the agent's policy $\pi(\mathbf{a}_t|\mathbf{s}_t)$.

3.1 GOAL-CONDITIONAL REINFORCEMENT LEARNING

In the standard RL setting described before, the agent only learns to solve a single task, specified by the reward function. If we are interested in an agent that can solve multiple tasks (each with a different reward function) in an environment, we can train the agent on those tasks by telling the agent which distinct task to solve at each time step. But how can we describe a task to the agent? A simple, yet not too restrictive way is to let each task correspond to an environment state the agent has to reach, denoted as the goal state g. The task is then given to the agent by conditioning its policy $\pi(a_t \mid s_t, g)$ on the goal g, and the agent's objective turns to maximize the expected goal-conditional return:

$$\mathbb{E}_{\mathbf{g} \sim G} \left[\sum_{t=1}^{T} \mathbb{E}_{\mathbf{s}_t \sim \rho_\pi, \mathbf{a}_t \sim \pi, \mathbf{s}_{t+1} \sim p} \left[r_{\mathbf{g}}(\mathbf{s}_{t+1}) \right] \right] \tag{1}$$

where G is some distribution over the space of goals $\mathcal{G} \subseteq \mathcal{S}$ the agent receives for training. The reward function can, for example, be the negative distance of the current state to the goal: $r_{\mathbf{g}}(\mathbf{s}) = -\|\mathbf{s} - \mathbf{g}\|$. Often, we are only interested in reaching a partial state configuration, e.g. moving an object to a target position, and want to avoid using the full environment state as the goal. In this case, we have to provide a mapping $m\colon \mathcal{S} \mapsto \mathcal{G}$ of states to the desired goal space; the mapping is then used to compute the reward function, i.e. $r_{\mathbf{g}}(\mathbf{s}) = -\|m(\mathbf{s}) - \mathbf{g}\|$.

As the reward is computed within the goal space, it is clear that the choice of goal space plays a crucial role in determining the difficulty of the learning task. If the goal space is low-dimensional and structured, e.g. in terms of ground truth positions of objects, rewards provide a meaningful signal towards reaching goals. However, if we only have access to high-dimensional, unstructured observations, e.g. camera images, and we naively choose this space as the goal space, optimization becomes hard as there is little correspondence between the reward and the distance of the underlying world states (Nair et al., 2018).

One option to deal with such difficult observation spaces is to *learn a goal space* in which the RL task becomes easier. For instance, we can try to find a low-dimensional latent space $\mathcal{Z}$ and use it both as the input space to our policy and the space in which we specify goals. If the environment is composed of independent parts that we intend to control separately, intuitively, learning to control is easiest if the latent space is also structured in terms of those independent components. Previous research (Nair et al., 2018; Pong et al., 2020) relied on the disentangling properties of representation learning models such as the β-VAE (Higgins et al., 2017) for this purpose. However, these models become insufficient when faced with multi-object scenarios due to the increasing combinatorial complexity of the scene, as we show in Sec. 5.2 and in App. A.2. Instead, we use a model explicitly geared towards inferring object-structured representations, which we introduce in the next section.

3.2 Structured representation learning with SCALOR

SCALOR (Jiang et al., 2019) is a probabilistic generative world model for learning object-oriented representations of a video or stream of high-dimensional environment observations. SCALOR assumes that the environment observation $\mathbf{o}_t$ at step t is generated by the background latent variable $\mathbf{z}_t^{\text{bg}}$ and the foreground latent variable $\mathbf{z}_t^{\text{fg}}$. The foreground is further factorized into a set of object representations $\mathbf{z}_t^{\text{fg}} = \{\mathbf{z}_{t,n}\}_{n \in \mathcal{O}_t}$, where $\mathcal{O}_t$ is the set of recognised object indices. To combine the information from previous time steps, a propagation-discovery model is used (Kosiorek et al., 2018). In SCALOR, an object is represented by $\mathbf{z}_{t,n} = \left(z_{t,n}^{\text{pres}}, \mathbf{z}_{t,n}^{\text{where}}, \mathbf{z}_{t,n}^{\text{what}}\right)$. The scalar $z_{t,n}^{\text{pres}}$ defines if the object is present in the scene, whereas the vector $\mathbf{z}_{t,n}^{\text{what}}$ encodes object appearance. The component $\mathbf{z}_{t,n}^{\text{where}}$ is further decomposed into the object's center position $\mathbf{z}_{t,n}^{\text{pos}}$, scale $\mathbf{z}_{t,n}^{\text{scale}}$, and depth $z_{t,n}^{\text{depth}}$. With this, the generative process of SCALOR can be written as:

$$p(\mathbf{o}_{1:T}, \mathbf{z}_{1:T}) = p(\mathbf{z}_1^{\mathcal{D}})p(\mathbf{z}_1^{\text{bg}}) \prod_{t=2}^{T} \underbrace{p(\mathbf{o}_t \mid \mathbf{z}_t)}_{\text{rendering}} \underbrace{p(\mathbf{z}_t^{\text{bg}} \mid \mathbf{z}_{<t}^{\text{bg}}, \mathbf{z}_t^{\text{fg}})}_{\text{background transition}} \underbrace{p(\mathbf{z}_t^{\mathcal{D}} \mid \mathbf{z}_t^{\mathcal{P}})}_{\text{discovery}} \underbrace{p(\mathbf{z}_t^{\mathcal{P}} \mid \mathbf{z}_{<t})}_{\text{propagation}}, \tag{2}$$

where $\mathbf{z}_t = (\mathbf{z}_t^{\text{bg}}, \mathbf{z}_t^{\text{fg}})$, $\mathbf{z}_t^{\mathcal{D}}$ contains latent variables of objects discovered in the present step, and $\mathbf{z}_t^{\mathcal{P}}$ contains latent variables of objects propagated from the previous step. Due to the intractability of the true posterior distribution $p(\mathbf{z}_{1:T}|\mathbf{o}_{1:T})$, SCALOR is trained using variational inference with the following posterior approximation:

$$q(\mathbf{z}_{1:T} \mid \mathbf{o}_{1:T}) = \prod_{t=1}^{T} q(\mathbf{z}_t \mid \mathbf{z}_{<t}, \mathbf{o}_{\leq t}) = \prod_{t=1}^{T} q(\mathbf{z}_t^{\text{bg}} \mid \mathbf{z}_t^{\text{fg}}, \mathbf{o}_t)\, q(\mathbf{z}_t^{\mathcal{D}} \mid \mathbf{z}_t^{\mathcal{P}}, \mathbf{o}_{\leq t})\, q(\mathbf{z}_t^{\mathcal{P}} \mid \mathbf{z}_{<t}, \mathbf{o}_{\leq t}), \tag{3}$$

by maximizing the following evidence lower bound $\mathcal{L}(\theta, \phi) =$

$$\sum_{t=1}^{T} \mathbb{E}_{q_\phi(\mathbf{z}_{<t}|\mathbf{o}_{<t})}\Big[\mathbb{E}_{q_\phi(\mathbf{z}_t|\mathbf{z}_{<t},\mathbf{o}_{\leq t})}\left[\log p_\theta(\mathbf{o}_t \mid \mathbf{z}_t)\right] - D_{\text{KL}}\left[q_\phi(\mathbf{z}_t \mid \mathbf{z}_{<t}, \mathbf{o}_{\leq t}) \parallel p_\theta(\mathbf{z}_t \mid \mathbf{z}_{<t})\right]\Big], \tag{4}$$

where D_{KL} denotes the Kullback-Leibler divergence. As we are using SCALOR in an active setting, we additionally condition the next step posterior predictions on the actions $\mathbf{a}_t$ taken by the agent. For more details and hyperparameters used to train SCALOR, we refer to App. D.3. In the next section, we describe how the structured representations learned by SCALOR can be used in downstream RL tasks such as goal-conditional visual RL.

4 Self-Supervised Multi-Object Reinforcement Learning

Learning from flexible representations obtained from unsupervised scene decomposition methods such as SCALOR creates several challenges for RL agents. In particular, these representations consist of sets of vectors, whereas standard policy architectures assume fixed-length state vectors as input. We propose to use a *goal-conditioned attention policy* that can handle sets as inputs and flexibly learns to attend to those parts of the representation needed to achieve the goal at hand.

In the setting we consider, the agent is not given *any reward signal or goals from the environment* at the training stage. Thus, to discover useful skills that can be used during evaluation tasks, the agent needs to rely on *self-supervision* in the form of an internally constructed reward signal and self-proposed goals. Previous VAE-based methods used latent distances to the goal state as the reward signal. However, for compositional goals, this means that the agent needs to master the simultaneous manipulation of all objects. In our experiments in Sec. 5.1, we show that even with fully disentangled, ground-truth representations of the scene, this is a challenging setting for state-of-the-art model-free RL agents. Instead, we propose to use the discovered structure of the learned goal and state spaces twofold: the structure within each representation, namely object position and appearance, to construct a reward signal, and the set-based structure between representations to construct sub-goals that correspond to manipulating individual objects.

4.1 Policy with Goal-conditioned Attention

We use the multi-head attention mechanism (Vaswani et al., 2017) as the first stage of our policy π_θ to deal with the challenge of set-based input representations. As the policy needs to flexibly vary its behavior based on the goal at hand, it appears sensible to steer the attention using a goal-dependent query $Q(\mathbf{z}_g) = \mathbf{z}_g W^q$. Each object is allowed to match with the query via an object-dependent key $K(\mathbf{z}_t) = \mathbf{z}_t W^k$ and contribute to the attention's output through the value $V(\mathbf{z}_t) = \mathbf{z}_t W^v$, which is weighted by the similarity between $Q(\mathbf{z}_g)$ and $K(\mathbf{z}_t)$. As inputs, we concatenate the representations for object n to vectors $\mathbf{z}_{t,n} = [\mathbf{z}_{t,n}^{\text{what}}; \mathbf{z}_{t,n}^{\text{where}}; z_{t,n}^{\text{depth}}]$, and similarly the goal representation to $\mathbf{z}_g = [\mathbf{z}_g^{\text{what}}; \mathbf{z}_g^{\text{where}}; z_g^{\text{depth}}]$. The attention head A_k is computed as

$$A_k = \text{softmax}\left(\frac{\mathbf{z}_g W^q (Z_t W^k)^T}{\sqrt{d_e}}\right) Z_t W^v, \tag{5}$$

where Z_t is a packed matrix of all $\mathbf{z}_{t,n}$'s, W^q, W^k, W^v constitute learned linear transformations and d_e is the common key, value and query dimensionality. The final attention output A is a concatenation of all the attention heads $A = [A_1; \ldots; A_K]$. In general, we expect it to be beneficial for the policy to not only attend to entities conditional on the goal; we thus let some heads attend based on a set of input independent, learned queries, which are not conditioned on the goal. We go into more details about the attention mechanism in App. D.1 and ablate the impact of different choices in App. B.

The second stage of our policy is a fully-connected neural network f that takes as inputs A and the goal representation $\mathbf{z}_g$ and outputs an action $\mathbf{a}_t$. The full policy π_θ can thus be described by

$$\pi_\theta\left(\{\mathbf{z}_{t,n}\}_{n \in \mathcal{O}_t}, \mathbf{z}_g\right) = f(A, \mathbf{z}_g). \tag{6}$$

4.2 Self-Supervised Training

In principle, our policy can be trained with any goal-conditional model-free RL algorithm. For our experiments, we picked soft-actor critic (SAC) (Haarnoja et al., 2018b) as a state-of-the-art method for continuous action spaces, using hindsight experience replay (HER) (Andrychowicz et al., 2017) as a standard way to improve sample-efficiency in the goal-conditional setting.

The training algorithm is summarized in Alg. 1. We first train SCALOR on data collected from a random policy and fit a distribution $p(\mathbf{z}^{\text{where}})$ to representations $\mathbf{z}^{\text{where}}$ of collected data. Each rollout, we generate a new goal for the agent by picking a random $\mathbf{z}^{\text{what}}$ from the initial observation $\mathbf{z}_1$ and sampling a new $\mathbf{z}^{\text{where}}$ from the fitted distribution $p(\mathbf{z}^{\text{where}})$. The policy is then rolled out using this goal. During off-policy training, we are relabeling goals with HER, and, similar to RIG (Nair et al., 2018), also with "imagined goals" produced in the same way as the rollout goals.

Algorithm 1 SMORL: Self-Supervised Multi-Object RL (Training)

```
Require: SCALOR encoder $q_\phi$, goal-conditional policy $\pi_\theta$, goal-conditional SAC trainer, number of training episodes $K$.
Train SCALOR on sequences uniformly sampled from $\mathcal{D}$ using loss described in Eq. 4.
Fit prior $p(\mathbf{z}^{\text{where}} \mid \mathbf{z}^{\text{what}})$ to the latent encodings of observations.
for $n = 1, ..., K$ episodes do
    Sample goal $\mathbf{z}_g = \left(\hat{\mathbf{z}}_g^{\text{where}}, \mathbf{z}_g^{\text{what}}\right)$.
    Collect episode data with policy $\pi_\theta(\mathbf{a}_t \mid \mathbf{z}_t, \mathbf{z}_g)$ and SCALOR representations of observations $q_\phi(\mathbf{z}_t \mid \mathbf{z}_{<t}, \mathbf{o}_{\leq t})$.
    Store transitions $(\mathbf{z}_t, \mathbf{a}_t, \mathbf{z}_{t+1}, \mathbf{z}_g)$ into replay buffer $\mathcal{R}$.
    Sample transitions from replay buffer $(\mathbf{z}, \mathbf{a}, \mathbf{z}', \mathbf{z}_g) \sim \mathcal{R}$.
    Relabel $\mathbf{z}_g^{\text{where}}$ goal components to a combination of future states and $p(\mathbf{z}^{\text{where}} \mid \mathbf{z}^{\text{what}})$.
    Compute matching reward signal $R = r(\mathbf{z}', \mathbf{z}_g)$.
    Update policy $\pi_\theta(\mathbf{a}_t \mid \mathbf{z}_t, \mathbf{z}_g)$ using $R$ with SAC trainer.
end for
```

We also refer to Alg. 2 in App. D.2 for a more detailed description of the algorithm.

A challenge with compositional representations is how to measure the progress of the agent towards achieving the chosen goal. As the goal always corresponds to a single object, we have to extract the state of this object in the current observation in order to compute a reward. One way is to rely on the tracking of objects, as was shown possible e.g. by SCALOR (Jiang et al., 2019). However, as the agent learns, we noticed that it would discover some flaws of the tracking and exploit them to get a maximal reward that is not connected with environment changes, but rather with internal vision and tracking flaws (details in App. E).

We follow an alternative approach, namely to use the $\mathbf{z}^{\text{what}}$ component of discovered objects and match them with the current goal representation $\mathbf{z}_g^{\text{what}}$. As the $\mathbf{z}^{\text{what}}$ space encodes the appearance of objects, two detections corresponding to the same object should be close in this space (we verify that this hypothesis holds in App. A.1). Thus, it is easy to find the object corresponding to the current goal object using the distance $\min_k ||\mathbf{z}_k^{\text{what}} - \mathbf{z}_g^{\text{what}}||$. In case of failure to discover a close representation, i.e. when all $\mathbf{z}_k^{\text{what}}$ have a distance larger than some threshold α to the goal representation $\mathbf{z}_g^{\text{what}}$, we use a fixed negative reward $r_{\text{no\_goal}}$ to incentivise the agent to avoid this situation.

Our reward signal is thus

$$r(\mathbf{z}, \mathbf{z}_g) = \begin{cases} -||\mathbf{z}_{\hat{k}}^{\text{where}} - \mathbf{z}_g^{\text{where}}|| & \text{if } \min_k ||\mathbf{z}_k^{\text{what}} - \mathbf{z}_g^{\text{what}}|| < \alpha, \\ r_{\text{no\_goal}} & \text{otherwise,} \end{cases} \tag{7}$$

where $\hat{k} = \arg\min_k ||\mathbf{z}_k^{\text{what}} - \mathbf{z}_g^{\text{what}}||$.

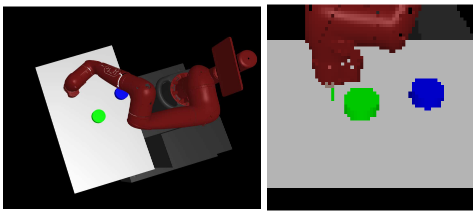
4.3 Composing Independent Sub-Goals during Evaluation

At evaluation time, the agent receives a goal image from the environment showing the state to achieve. The goal image is processed by SCALOR to yield a set of goal vectors. For our experiments, we assume that these sub-goals are independent of each other and that the agent can thus sequentially achieve them by cycling through them until all of them are solved. The evaluation algorithm is summarized in Alg. 3, with more details added in App. D.2.

5 Experiments

We have done computational experiments to address the following questions:

- How well does our method scale to challenging tasks with a large number of objects in case when ground-truth representations are provided?
- How does our method perform compared to prior visual goal-conditioned RL methods on image-based, multi-object continuous control tasks?
- How suitable are the representations learned by the compositional generative world model for discovering and solving RL tasks?



(a) View from top (b) Agent observation

Figure 2: Multi-Object Visual Push and Rearrange environments with 2 objects and a Sawyer robotic arm.

To answer these questions, we constructed the *Multi-Object Visual Push* and *Multi-Object Visual Rearrange* environments. Both environments are based on MuJoCo (Todorov et al., 2012) and the Multiworld package for image-based continuous control tasks introduced by Nair et al. (2018), and contain a 7-dof Sawyer arm where the agent needs to be controlled to manipulate a variable number of small picks on a table. In the first environment, the objects are located on fixed positions in front of the robot arm that the arm must push to random target positions. We included this environment as it largely corresponds to the *Visual Pusher* environments of Nair et al. (2018). In the second environment, the task is to rearrange the objects from random starting positions to random target positions. This task is more challenging for RL algorithms due to the randomness of initial object positions. For both environments, we measure the performance of the algorithms as the average distance of all pucks to their goal positions on the last step of the episode. Our code, as well as the multi-objects environments will be made public after the paper publication.

5.1 SMORL with ground-truth (GT) state representation

We first compared SMORL with ground-truth representation with Soft Actor-Critic (SAC) (Haarnoja et al., 2018a) with Hindsight Experience Replay (HER) relabeling (Andrychowicz et al., 2017) that takes an unstructured vector of all objects coordinates as input. We are using a one-hot encoding for object identities $\mathbf{z}^{\text{what}}$ and object and arm coordinates as $\mathbf{z}^{\text{where}}$ components. With such a representation, the matching task becomes trivial, so our main focus in this experiment is on the benefits of the goal-conditioned attention policy and the sequential solving of independent sub-tasks. We show the results in Fig. 3. While for 2 objects, SAC+HER is performing similarly, for 3 and 4 objects, SAC+HER fails to rearrange any of the objects. In contrast, SMORL equipped with ground-truth representation is still able to rearrange 3 and 4 objects, and it can solve the more simple sub-tasks of moving each object independently. This shows that provided with good representations, SMORL can use them for constructing useful sub-tasks and learn how to solve them.

5.2 Visual RL Methods Comparison

We compare the performance of our algorithm with two other self-supervised, multi-task visual RL algorithms on our two environments, with one and two objects. The first one, RIG (Nair et al., 2019), uses the VAE latent space to sample goals and to estimate the reward signal. The second one, Skew-Fit (Pong et al., 2020), also uses the VAE latent space, however, is additionally biased on rare observations that were not modeled well by the VAE on previously collected data. In terms of computational complexity, both our method and RIG need to train a generative model before RL training. We note that training SCALOR is more costly than training RIG's VAE due to the sequence processing utilized by SCALOR. However, once trained, SCALOR only adds little overhead compared to RIG's VAE during RL training, and compared to Skew-Fit, our method is still faster to train as Skew-Fit needs to continuously retrain its VAE.

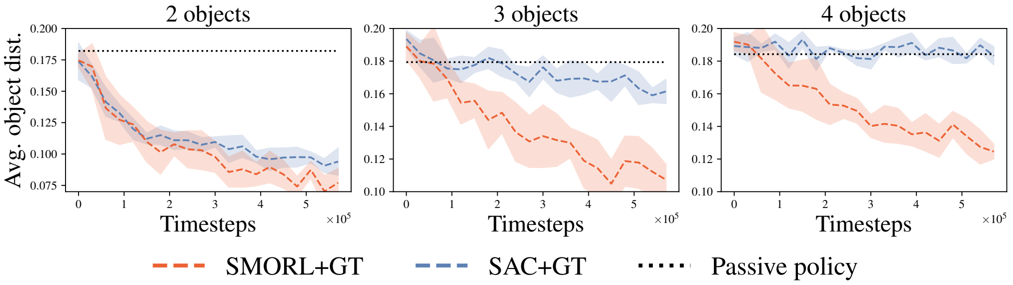


Figure 3: Average distance of objects to goal positions, comparing SMORL using ground truth representations to SAC with ground truth representations in the Rearrange environment with different number of objects. SAC struggles to improve performance when the combinatorial complexity of the scene rises. The dotted line indicates the performance of a *passive policy* that performs no movements. Results averaged over 5 random seeds, shaded regions indicate one standard deviation.

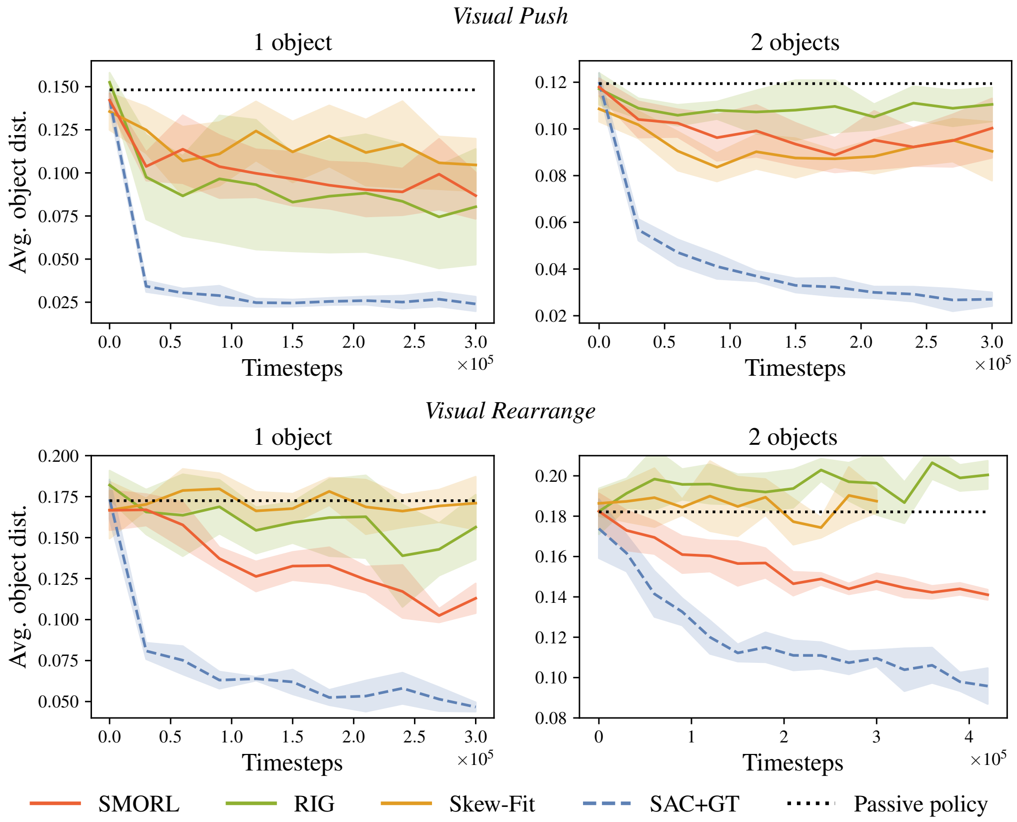


Figure 4: Average distance of objects to goal positions, comparing SMORL to Visual RL Baselines. In addition to the baselines, we show SAC's performance with ground truth representations. Results averaged over 5 random seeds, shaded regions indicate one standard deviation.

We show the results in Fig. 4. For the simpler *Multi-Object Visual Push* environment, the performance of SMORL is comparable to the best performing baseline, while for the more challenging *Multi-Object Visual Rearrange* environment, SMORL is significantly better then both RIG and Skew-Fit. This shows that learning of object-oriented representations brings benefits for goal sampling and self-supervised learning of useful skills. However, our method is still significantly worse than SAC with ground-truth representations. We hypothesize that one reason for this could be that SCALOR right now does not properly deal with occluded objects, which makes the environment partially observable from the point of view of the agent. On top of this, we suspect noise in the representations, misdetections and an imperfect matching signal to slow down training and ultimately hurt performance. Thus, we expect that adding recurrence to the policy or improving SCALOR itself could help close the gap to an agent with perfect information.

5.3 Out-of-Distribution Generalization for different number of objects

One important advantage of structured policies is that they could potentially still be applicable for observations that are from different, but related distributions. Standard visual RL algorithms were shown to be sensitive to small changes unrelated to the current task (Higgins et al., 2018). To see how our algorithm can generalize to a changing environment, we tested our SMORL agent trained on observations of the Rearrange environment with 2 objects on the environment with 1 object. As can be seen from Fig. 5, the performance of such an agent increases during training up to a performance comparable to a SMORL agent that was trained on the 1 object environment.

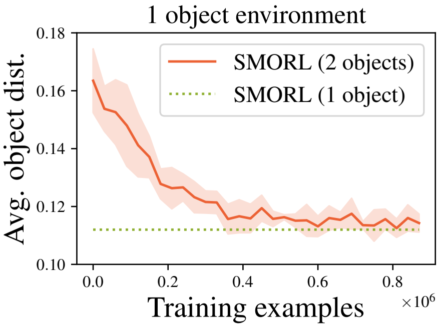


Figure 5: Out-of-distribution generalization of SMORL agent training on *Visual Rearrange* with two objects and being tested with one object. Green line shows final performance when training with one object.

6 Conclusion and Future Work

In this work, we have shown that discovering structure in the observations of the environment with a compositional generative world models and using it for controlling different parts of the environment is crucial for solving tasks in compositional environments. Learning to manipulate different parts of object-centric representations is a powerful way to acquire useful skills such as object manipulation. Our SMORL agent learns how to control different entities in the environment and can then combine the learned skills to achieve more complex compositional goals such as rearranging several objects using only the final image of the arrangement.

Given the results presented so far, there are a number of interesting directions to take this work. First, one can combine learned sub-tasks with a planning algorithm to achieve a particular goal. Currently, the agent is simply sequentially cycling through all discovered sub-tasks, so we expect that a more complex planning algorithm as e.g. described by Nasiriany et al. (2019) could allow solving more challenging tasks and improve the overall performance of the policy. To this end, considering interactions between objects in the manner of Fetaya et al. (2018) or Kipf et al. (2020) could help to lift the assumption of independence of sub-tasks. Second, prioritizing certain sub-tasks during learning, similar to Blaes et al. (2019), could accelerate the training of the agent. Finally, an active training of SCALOR to combine the object-oriented bias of SCALOR with a bias towards independently controllable objects (Thomas et al., 2018) is an interesting direction for future research.

Acknowledgements

The authors thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting Maximilian Seitzer. Andrii Zadaianchuk is supported by the Max Planck ETH Center for Learning Systems. We acknowledge the support from the German Federal Ministry of Education and Research (BMBF) through the Tübingen AI Center (FKZ: 01IS18039B).

References

Marcin Andrychowicz, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, OpenAI Pieter Abbeel, and Wojciech Zaremba. Hindsight experience replay. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30, pp. 5048–5058. Curran Associates, Inc., 2017. URL `https://proceedings.neurips.cc/paper/2017/file/453fadbd8a1a3af50a9df4df899537b5-Paper.pdf`.

Adrien Baranes and Pierre-Yves Oudeyer. Active learning of inverse models with intrinsically motivated goal exploration in robots. *Robotics Auton. Syst.*, 61:49–73, 2013.

Sebastian Blaes, Marin Vlastelica Pogančić, Jiajie Zhu, and Georg Martius. Control what you can: Intrinsically motivated task-planning agent. In H. Wallach, H. Larochelle, A. Beygelzimer, F. Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32, pp. 12541–12552. Curran Associates, Inc., 2019. URL `https://proceedings.neurips.cc/paper/2019/file/b6f97e6f0fd175613910d613d574d0cb-Paper.pdf`.

Christopher P. Burgess, Loic Matthey, Nicholas Watters, Rishabh Kabra, Irina Higgins, Matt Botvinick, and Alexander Lerchner. Monet: Unsupervised scene decomposition and representation, 2019.

Ricky TQ Chen, Xuechen Li, Roger B Grosse, and David K Duvenaud. Isolating sources of disentanglement in variational autoencoders. In *Advances in Neural Information Processing Systems*, pp. 2610–2620, 2018.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

T. Fetaya, E. Wang, K.-C. Welling, M. Zemel, Thomas Kipf, Ethan Fetaya, Kuan-Chieh Wang, M. Welling, and R. Zemel. Neural relational inference for interacting systems. In *ICML*, 2018.

D. Ghosh, A. Gupta, and S. Levine. Learning actionable representations with goal-conditioned policies. *ArXiv*, abs/1811.07819, 2019.

Klaus Greff, Rupesh Kumar Srivastava, and Jürgen Schmidhuber. Binding via reconstruction clustering, 2016.

Klaus Greff, Sjoerd Van Steenkiste, and Jürgen Schmidhuber. Neural expectation maximization. In *Advances in Neural Information Processing Systems*, pp. 6691–6701, 2017.

Klaus Greff, Raphaël Lopez Kaufman, Rishabh Kabra, Nick Watters, Chris Burgess, Daniel Zoran, Loic Matthey, Matthew Botvinick, and Alexander Lerchner. Multi-object representation learning with iterative variational inference. *Proceedings of the 36nd International Conference on Machine Learning*, 2019.

Klaus Greff, Sjoerd van Steenkiste, and Jürgen Schmidhuber. On the binding problem in artificial neural networks, 2020.

T. Haarnoja, Aurick Zhou, P. Abbeel, and S. Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *ICML*, 2018a.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. *arXiv preprint arXiv:1801.01290*, 2018b.

Karol Hausman, Jost Tobias Springenberg, Ziyu Wang, Nicolas Heess, and Martin Riedmiller. Learning an embedding space for transferable robot skills. In *International Conference on Learning Representations*, 2018. URL `https://openreview.net/forum?id=rk07ZXZRb`.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. *arXiv preprint arXiv:1911.05722*, 2019.

I. Higgins, Loïc Matthey, A. Pal, C. Burgess, Xavier Glorot, M. Botvinick, S. Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *ICLR*, 2017.

Irina Higgins, Arka Pal, Andrei A. Rusu, Loic Matthey, Christopher P Burgess, Alexander Pritzel, Matthew Botvinick, Charles Blundell, and Alexander Lerchner. Darla: Improving zero-shot transfer in reinforcement learning, 2018.

Jindong Jiang, Sepehr Janghorbani, Gerard de Melo, and Sungjin Ahn. Scalable object-oriented sequential generative models. *arXiv preprint arXiv:1910.02384*, 2019.

Dmitry Kalashnikov, Alex Irpan, Peter Pastor, Julian Ibarz, Alexander Herzog, Eric Jang, Deirdre Quillen, Ethan Holly, Mrinal Kalakrishnan, Vincent Vanhoucke, et al. Qt-opt: Scalable deep reinforcement learning for vision-based robotic manipulation. *arXiv preprint arXiv:1806.10293*, 2018.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2015.

Thomas Kipf, Elise van der Pol, and Max Welling. Contrastive learning of structured world models. In *International Conference on Learning Representations*, 2020. URL `https://openreview.net/forum?id=H1gax6VtDB`.

Adam Roman Kosiorek, Hyunjik Kim, Ingmar Posner, and Yee Whye Teh. Sequential attend, infer, repeat: Generative modelling of moving objects. In *Advances in Neural Information Processing Systems*, 2018. URL `https://arxiv.org/abs/1806.01794`.

Adrien Laversanne-Finot, Alexandre Pere, and Pierre-Yves Oudeyer. Curiosity driven exploration of learned disentangled goal spaces. In Aude Billard, Anca Dragan, Jan Peters, and Jun Morimoto (eds.), *Proceedings of The 2nd Conference on Robot Learning*, volume 87 of *Proceedings of Machine Learning Research*, pp. 487–504. PMLR, 29–31 Oct 2018. URL `http://proceedings.mlr.press/v87/laversanne-finot18a.html`.

Sergey Levine, Chelsea Finn, Trevor Darrell, and Pieter Abbeel. End-to-end training of deep visuomotor policies. *The Journal of Machine Learning Research*, 17(1):1334–1373, 2016.

Sergey Levine, Peter Pastor, Alex Krizhevsky, Julian Ibarz, and Deirdre Quillen. Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. *The International Journal of Robotics Research*, 37(4-5):421–436, 2018.

Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *International Conference on Machine Learning*, pp. 4114–4124, 2019.

Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention, 2020.

Corey Lynch, Mohi Khansari, Ted Xiao, Vikash Kumar, Jonathan Tompson, Sergey Levine, and Pierre Sermanet. Learning latent plans from play. *Conference on Robot Learning (CoRL)*, 2019. URL `https://arxiv.org/abs/1903.01973`.

Ashvin Nair, Shikhar Bahl, Alexander Khazatsky, Vitchyr H. Pong, G. Berseth, and S. Levine. Contextual imagined goals for self-supervised robotic learning. In *CoRL*, 2019.

Ashvin V Nair, Vitchyr Pong, Murtaza Dalal, Shikhar Bahl, Steven Lin, and Sergey Levine. Visual reinforcement learning with imagined goals. In *Advances in Neural Information Processing Systems*, pp. 9191–9200, 2018.

Soroush Nasiriany, Vitchyr Pong, Steven Lin, and Sergey Levine. Planning with goal-conditioned policies. In *Advances in Neural Information Processing Systems*, pp. 14843–14854, 2019.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32, pp. 8026–8037. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/bdbca288fee7f92f2bfa9f7012727740-Paper.pdf.

Deepak Pathak, Parsa Mahmoudieh, Guanghao Luo, Pulkit Agrawal, Dian Chen, Yide Shentu, Evan Shelhamer, Jitendra Malik, Alexei A Efros, and Trevor Darrell. Zero-shot visual imitation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 2050–2053, 2018.

Vitchyr H Pong, Murtaza Dalal, Steven Lin, Ashvin Nair, Shikhar Bahl, and Sergey Levine. Skew-fit: State-covering self-supervised reinforcement learning. In *Proceedings of the 37nd International Conference on Machine Learning*, volume 42 of *JMLR Workshop and Conference Proceedings*. JMLR, 2020.

Alexandre Péré, Sébastien Forestier, Olivier Sigaud, and Pierre-Yves Oudeyer. Unsupervised learning of goal spaces for intrinsically motivated goal exploration. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=S1DWPP1A-.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

Michal Rolinek, Dominik Zietlow, and Georg Martius. Variational autoencoders pursue pca directions (by accident). In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, June 2019. URL http://openaccess.thecvf.com/content_CVPR_2019/papers/Rolinek_Variational_Autoencoders_Pursue_PCA_Directions_by_Accident_CVPR_2019_paper.pdf.

Valentin Thomas, Emmanuel Bengio, William Fedus, Jules Pondard, Philippe Beaudoin, Hugo Larochelle, Joelle Pineau, Doina Precup, and Yoshua Bengio. Disentangling the independently controllable factors of variation by interacting with the world. *arXiv preprint arXiv:1802.09484*, 2018.

E. Todorov, T. Erez, and Y. Tassa. Mujoco: A physics engine for model-based control. *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5026–5033, 2012.

Sjoerd van Steenkiste, Klaus Greff, and Jürgen Schmidhuber. A perspective on objects and systematic generalization in model-based rl. *arXiv preprint arXiv:1906.01035*, 2019.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pp. 5998–6008, 2017.

Rishi Veerapaneni, John D. Co-Reyes, Michael Chang, Michael Janner, Chelsea Finn, Jiajun Wu, Joshua B. Tenenbaum, and Sergey Levine. Entity abstraction in visual model-based reinforcement learning, 2020.

David Warde-Farley, Tom Van de Wiele, Tejas Kulkarni, Catalin Ionescu, Steven Hansen, and Volodymyr Mnih. Unsupervised control through non-parametric discriminative rewards. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=r1eVMnA9K7.

Nicholas Watters, Loic Matthey, Matko Bosnjak, Christopher P. Burgess, and Alexander Lerchner. Cobra: Data-efficient model-based rl through unsupervised object discovery and curiosity-driven exploration, 2019.

Marissa A. Weis, Kashyap Chitta, Yash Sharma, Wieland Brendel, Matthias Bethge, Andreas Geiger, and Alexander S. Ecker. Unmasking the inductive biases of unsupervised object representations for video sequences, 2020.

Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Russ R Salakhutdinov, and Alexander J Smola. Deep sets. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30, pp. 3391–3401. Curran Associates, Inc., 2017. URL `https://proceedings.neurips.cc/paper/2017/file/f22e4747da1aa27e363d86d40ff442fe-Paper.pdf`.

Appendix B

Causal Influence Detection for Improving Efficiency in Reinforcement Learning

The following 14 pages have been published as:

Maximilian Seitzer, Bernhard Schölkopf, and Georg Martius (2021). "Causal Influence Detection for Improving Efficiency in Reinforcement Learning". In: *Conference on Neural Information Processing Systems (NeurIPS)*

A summary of this publication can be found in Chap. 5. The supplementary material to this publication is available in the online version.

Causal Influence Detection for Improving Efficiency in Reinforcement Learning

Maximilian Seitzer
MPI for Intelligent Systems
Tübingen, Germany
maximilian.seitzer@tue.mpg.de

Bernhard Schölkopf
MPI for Intelligent Systems
Tübingen, Germany
bs@tue.mpg.de

Georg Martius
MPI for Intelligent Systems
Tübingen, Germany
georg.martius@tue.mpg.de

Abstract

Many reinforcement learning (RL) environments consist of independent entities that interact sparsely. In such environments, RL agents have only limited influence over other entities in any particular situation. Our idea in this work is that learning can be efficiently guided by knowing when and what the agent can influence with its actions. To achieve this, we introduce a measure of *situation-dependent causal influence* based on conditional mutual information and show that it can reliably detect states of influence. We then propose several ways to integrate this measure into RL algorithms to improve exploration and off-policy learning. All modified algorithms show strong increases in data efficiency on robotic manipulation tasks.

1 Introduction

Reinforcement learning (RL) is a promising route towards versatile and dexterous artificial agents. Learning from interactions can lead to robust control strategies that can cope with all the intricacies of the real world that are hard to engineer correctly. Still, many relevant tasks such as object manipulation pose significant challenges for RL. Although impressive results have been achieved using simulation-to-real transfer [1] or heavy physical parallelization [2], training requires countless hours of interaction. Improving sample efficiency is thus a key concern in RL. In this paper, we approach this issue from a causal inference perspective.

When is an agent in control of its environment? An agent can only influence the environment by its actions. This seemingly trivial observation has the underappreciated aspect that the causal influence of actions is *situation dependent*. Consider the simple scenario of a robotic arm in front of an object on a table. Clearly, the object can only be moved when contact between the robot and object is made. Generally, there are situations where immediate causal influence is possible, while in others, none is. In this work, we formalize this situation-dependent nature of control and show how it can be exploited to improve the sample efficiency of RL agents. To this end, we derive a measure that captures the causal influence of actions on the environment and devise a practical method to compute it.

Knowing when the agent has control over an object of interest is important both from a learning and an exploration perspective. The learning algorithm should pay particular attention to these situations because (i) the robot is initially rarely in control of the object of interest, making training inefficient, (ii) physical contacts are hard to model, thus require more effort to learn and (iii) these states are enabling manipulation towards further goals. But for learning to take place, the algorithm first needs data that contains these relevant states. Thus, the agent has to take its causal influence into account already during exploration.

We propose several ways in which our measure of causal influence can be integrated into RL algorithms to address both the exploration, and the learning side. For exploration, agents can be rewarded with a bonus for visiting states of causal influence. We show that such a bonus leads the agent to quickly discover useful behavior even in the absence of task-specific rewards. Moreover,

35th Conference on Neural Information Processing Systems (NeurIPS 2021), virtual.

our approach allows to explicitly guide the exploration to favor actions with higher predicted causal impact. This works well as an alternative to ϵ-greedy exploration, as we demonstrate. Finally, for learning, we propose an off-policy prioritization scheme and show that it reliably improves data efficiency. Each of our investigations is backed by empirical evaluations in robotic manipulation environments and demonstrates a clear improvement of the state-of-the-art with the same generic influence measure.

2 Related Work

The idea underlying our work is that an agent can only sometimes influence its surroundings. This rests on two basic assumptions about the causal structure of the world. The first is that the world consists of independent entities, in accordance with the principle of independent causal mechanisms (ICM) [3], stating that the world's generative process consists of autonomous modules. The second assumption is that the potential influence that entities have over other entities is localized spatially and occurs sparsely in time. We can see this as explaining the sparse mechanism shift hypothesis, which states that naturally occurring distribution shifts will be due to local mechanism changes [4]. This is usually traced back to the ICM principle, i.e. that interventions on one mechanism will not affect other mechanisms [5]. But we argue that it is also due to the *limited interventional range* of agents (or, more generally, physical processes), which restricts the breadth and frequency of mechanism-changes in the real world. Previous work has used sparseness to learn disentangled representations [6, 7], causal models [8], or modular architectures [9]. In the present work, we show that taking the localized and sparse nature of influence into account can also strongly improve RL algorithms.

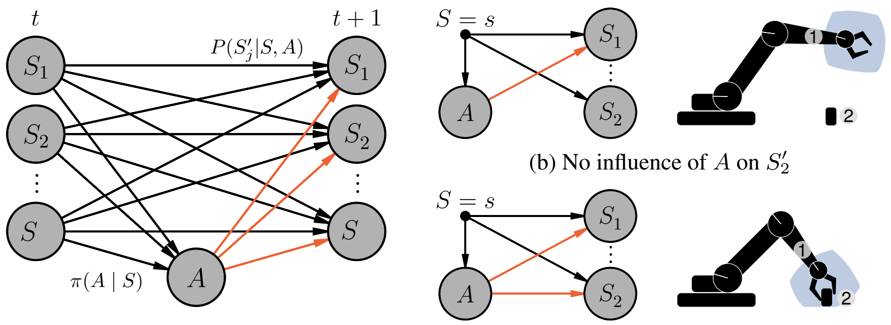
Detecting causal influence, informally, means deciding whether changing a causal variable would have an impact on another variable. This involves causal discovery, that is, finding the existence of arrows in the causal graph [10]. While the task of causal discovery is unidentifiable in general [11], there are assumptions which permit discovery [12], in particular in the time series setting we are concerned with [13]. Even if the existence of an arrow is established, the problem remains of quantifying its causal impact, for which various measures such as transfer entropy or information flow have been proposed [14–18]. We compare how our work relates to these measures in Sec. 4.1.

The intersection of RL and causality has been the subject of recent research [19–23]. Close to ours is the work of Pitis et al. [24], who also use influence detection, albeit to create counterfactual data that augments the training of RL agents. In Sec. 5, we find that our approach to action influence detection performs better than their heuristic approach. Additionally, we demonstrate that influence detection can also be used to help agents explore better. To this end, we use influence as a type of intrinsic motivation. For exploration, various signals have been proposed, e.g. model surprise [25–27], learning progress [27, 28], empowerment [29, 30], information gain [31–33], or predictive information [34, 35]. Inspired by causality, Sontakke et al. [36] introduce an exploration signal that leads agents to experiment with the environment to discover causal factors of variation. In concurrent work, Zhao et al. [37] propose to use mutual information between the agent and the environment state for exploration. As in our work, the agent is considered a separate entity from the environment. However, their approach does not discriminate between individual situations the agent is in. Causal influence is also related to the concept of contingency awareness from psychology [38], that is, the knowledge that one's actions can affect the environment. On Atari games, exploring through the lens of contingency awareness has led to state-of-the-art results [39, 40].

3 Background

We are concerned with a Markov decision process $\langle \mathcal{S}, \mathcal{A}, P, r, \gamma \rangle$ consisting of state and action space, transition distribution, reward function and discount factor.[1] Most real world environments consist of entities that behave mostly independently of each other. We model this by assuming a known state space factorization $\mathcal{S} = \mathcal{S}_1 \times \ldots \times \mathcal{S}_N$, where each $\mathcal{S}_i$ corresponds to the state of an entity.

[1] We use capital letters (e.g. X) to denote random variables, small letters to denote samples drawn from particular distributions (e.g. $x \sim P_X$), and caligraphy letters to denote graphs, sets and sample spaces (e.g. $x \in \mathcal{X}$). We denote distributions with P and their densitites with p.



(b) No influence of A on S_2'

(a) Causal Graph $\mathcal{G}$

(c) Influence of A on S_1' and S_2'

Figure 1: Causal graphical model capturing the environment transition from state S to S' by action A, factorized into state components. (a): Viewed globally over all time steps, all components of the state and the action can influence all state components at the next time step. (b, c): Given a situation $S = s$, some influences may or may not not hold in the *local causal graph* $\mathcal{G}_{S=s}$. In this paper, our aim is to detect which influence the action has on S', i.e. the presence of the red arrows.

3.1 Causal Graphical Models

We can model the one-step transition dynamics at time step t using a *causal graphical model* (CGM) [3, 10] over the set of random variables $\mathcal{V} = \{S_1, \dots, S_N, A, S_1', \dots, S_N'\}$, consisting of a directed graph $\mathcal{G}$ (see Fig. 1a) and a conditional distribution $P(V_i \mid \mathrm{Pa}(V_i))$ for each node $V_i \in \mathcal{V}$, where $\mathrm{Pa}(V_i)$ is the set of parents of V_i in the causal graph. We assume that the joint distribution $P_\mathcal{V}$ is Markovian with respect to the graph [3, Def. 6.21 (iii)], that is, its density exists and factorizes as

$$p(v_1, \dots, v_{|\mathcal{V}|}) = \prod_{i=1}^{|\mathcal{V}|} p(v_i \mid \mathrm{Pa}(V_i)). \tag{1}$$

In a CGM, we can model a (stochastic) intervention $\mathrm{do}(V_i := q(v_i \mid \mathrm{Pa}(V_i)))$ on variable V_i by replacing its conditional $p(v_i \mid \mathrm{Pa}(V_i))$ in Eq. 1 with the distribution $q(v_i \mid \mathrm{Pa}(V_i))$ [3]. Here, V_i could e.g. be a state component S_i, or the agent's action A. Thus, whereas a probabilistic graphical model represents a single distribution, a CGM represents a set of distributions [4].

The causal graph that we assume is shown in Fig. 1a. Within a time step, there are no edges, i.e. no instantaneous effects, except for the action which is computed by the policy $\pi(A \mid S)$. Between time steps, the graph is fully connected. The reason is that whenever an interaction between two components S_i and S_j, however unlikely, is possible, it is necessary to include an arrow $S_i \to S_j'$ (and vice versa). Nevertheless, during most concrete time steps, there should be *no* interaction between entities, reflecting the assumption that the state components represent independent entities in the world. In particular, the agent's "sphere of influence" (depicted in blue in Figs. 1b and 1c) is limited – its action A can only sparsely affect other entities. Thus, in this paper, we are interested in inferring the influence the action has in a specific state configuration $S = s$, that is, the *local causal model* in s.

Definition 1. Given a CGM with distribution $P_\mathcal{V}$ and graph $\mathcal{G}$, we define the *local CGM* induced by observing $X = x$ with $X \subset \mathcal{V}$ to be the CGM with joint distribution $P_{\mathcal{V}|X=x}$ and the graph $\mathcal{G}_{X=x}$ resulting from removing edges from $\mathcal{G}$ until $P_{\mathcal{V}|X=x}$ is causally minimal with respect to the graph.

Causal minimality tells us that each edge $X \to Y$ in the graph must be "active", in the sense that Y is conditionally dependent on X given all other parents of Y [3, Prop. 6.36].

3.2 The Cause of an Effect

When is an agent's action $A = a$ the cause of an outcome $B = b$? Answering this question precisely is surprisingly non-trivial and is studied under the name of *actual causation* [10, 41]. Humans would answer by contrasting the actual outcome to some normative world in which $A = a$ did not happen, i.e. they would ask the counterfactual question "What would have happened normally to B without $A = a$?" [41]. Algorithmitizing this approach poses certain problems. First, it requires a "normal" outcome which can be difficult to compute as it depends on the behavior of the different actors in the world. Second, it requires to actually observe the world's state without the agents interference. Such a "no influence" action may not be available for every agent. Instead, we are inspired by an alternative approach, the so-called *"but-for"* test: "$B = b$ would not have happened but for $A = a$." In other words, $A = a$ was a necessary condition for $B = b$ to occur, and under a different value for A, B would have had a different value as well. This matches well with an algorithmic view on causation: A is a cause of B if the value of A is required to determine the value of B [42].

The but-for test yields potentially counterintuitive assessments. Consider a robotic arm close to an object but performing an action that moves it away from the object. Then this action is considered a cause for the position of the object in this step, as an alternative action touching the object would have led to a different outcome. Algorithmically, knowing the action is required to determine what happens to the object – all actions are considered to be a cause in this situation. Importantly, this implies that we cannot differentiate whether individual actions are causes or not, but can only identify whether or not the agent has causal influence on other entities in the current state.

4 Causal Influence Detection

As the previous discussion showed, having causal influence is dependent on the situation the agent is in, rather than the chosen actions. We characterize this as the agent being *in control*, analogous to similar notions in control theory [43]. Formally, using the causal model introduced in Sec. 3, *we define the agent to be in control of S'_j in state $S = s$ if there is an edge $A \to S'_j$ in the local causal graph $\mathcal{G}_{S=s}$ under all interventions* $\mathrm{do}(A := \pi(a|s))$ *with π having full support.* The following proposition states when such an edge exists (proofs in Suppl. A.1).

Proposition 1. *Let $\mathcal{G}_{S=s}$ be the graph of the local CGM induced by $S = s$. There is an edge $A \to S'_j$ in $\mathcal{G}_{S=s}$ under the intervention* $\mathrm{do}(A := \pi(a|s))$ *if and only if $S'_j \not\perp\!\!\!\perp A \mid S = s$.*

To detect when the agent is in control, we can intervene with a policy. The following proposition gives conditions under which conclusions drawn from one policy generalize to many policies.

Proposition 2. *If there is an intervention* $\mathrm{do}(A := \pi(a|s))$ *under which $S'_j \not\perp\!\!\!\perp A \mid S = s$, this dependence holds under* all *interventions with full support, and the agent is in control of S'_j in s. If there is an intervention* $\mathrm{do}(A := \pi(a|s))$ *with π having full support under which $S'_j \perp\!\!\!\perp A \mid S = s$, this independence holds under* all *possible interventions and the agent is not in control of S'_j in s.*

4.1 Measuring Causal Action Influence

Our goal is to find a state-dependent quantity that measures whether the agent is in control of S'_j. As Prop. 1 tells us, control (or its absence) is linked to the independence $S'_j \perp\!\!\!\perp A \mid S = s$. A well-known measure of dependence is the conditional mutual information (CMI) [44] which is zero for independence. We thus propose to use (pointwise) CMI as a measure of *causal action influence* (CAI) that can be thresholded to get a classification of control (see Suppl. A.2 for a derivation):

$$C^j(s) := I(S'_j; A \mid S = s) = \mathbb{E}_{a\sim\pi}\Big[\mathrm{D_{KL}}\Big(P_{S'_j|s,a} \,\Big\|\, P_{S'_j|s}\Big)\Big]. \tag{2}$$

We want this measure to be independent of the particular policy used in the joint distribution $P(S, A, S')$. This is because we might not be able to sample from or evaluate this policy (e.g. in off-policy RL, the data stems from a mixture of different policies). Fortunately, Prop. 2 shows that to detect control, it is sufficient to demonstrate (in-)dependence for a single policy with full support. Thus, we can choose a uniform distribution over the action space as the policy: $\pi(A) := \mathcal{U}(\mathcal{A})$.

Let us discuss how CAI relates to previously suggested measures of (causal) influence. *Transfer entropy* [14] is a non-linear extension of Granger causality [45] quantifying causal influence in time series under certain conditions. CAI is similar to a one-step, local transfer entropy [17] with the difference that CAI conditions on the full state S. Janzing et al. [18] put forward a measure of *causal strength* fulfilling several natural criteria that other measures, including transfer entropy, fail to satisfy. In Suppl. A.3, we show that CAI is a pointwise version of Janzing et al.'s causal strength, for policies not conditional on the state S (adding further justification for the choice of a uniform random policy). Furthermore, we can relate CAI to notions of *controllability* [43]. Decomposing $C^j(s)$ as $H(S'_j \mid s) - H(S'_j \mid A, s)$, where H denotes the conditional entropy [44], we can interpret CAI as quantifiying the degree to which S'_j can be controlled in s, accounting for the system's intrinsic uncertainty that cannot be reduced by the action.

In the context of RL, *empowerment* [29, 30, 46] is a well-known quantity used for intrinsically-motivated exploration that leads agents to states of maximal influence over the environment. Empowerment, for a state s, is defined as the channel capacity between action and a future state, which coincides with $\max_\pi C(s)$ for one-step empowerment. CAI can thus be seen as a non-trivial lower bound of empowerment that is easier to compute. However, CAI differs from empowerment in that it does not treat the state space as monolithic and is specific to an entity. In Sec. 6.1, we demonstrate that an RL agent maximizing CAI quickly achieves control over its environment.

4.2 Learning to Detect Control

Estimating CMI is a hard problem on many levels: it involves computing high dimensional integrals, representing complicated distributions and having access to limited data; strictly speaking, each conditioning point s is seen only once in continuous spaces. In practice, one thus has to resort to an approximation. Non-parametric estimators based on nearest neighbors [47, 48] or kernels methods [49] are known to not scale well to higher dimensions [50]. Instead, we approach the problem by learning neural network models with suitable simplifying assumptions.

Expanding the KL divergence in Eq. 2, we can write CAI as

$$C^j(s) = I(S'_j; A \mid S = s) = \mathbb{E}_{A|s}\mathbb{E}_{S'_j|s,a}\left[\log \frac{p(s'_j \mid s, a)}{\int p(s'_j \mid s, a)\, \pi(a) \mathrm{d}a}\right] \tag{3}$$

To compute this term, we estimate the transition distribution $p(s'_j \mid s, a)$ from data. We then approximate the outer expectation and the transition marginal $p(s'_j \mid s)$ by sampling K actions from the policy π. This gives us the estimator

$$\hat{C^j}(s) = \frac{1}{K}\sum_{i=1}^{K}\left[\mathrm{D}_{\mathrm{KL}}\left(p(s'_j \mid s, a^{(i)}) \,\Big\|\, \frac{1}{K}\sum_{k=1}^{K} p(s'_j \mid s, a^{(k)})\right)\right], \tag{4}$$

with $\{a^{(1)}, \ldots, a^{(K)}\} \overset{\text{iid}}{\sim} \pi$. Here, we replaced the infinite mixture $p(s'_j \mid s)$ with a finite mixture, $p(s'_j \mid s) \approx \frac{1}{K}\sum_{i=1}^{K} p\big(s'_j \mid s, a^{(i)}\big)$, and used Monte-Carlo to approximate the expectation. Poole et al. [51] show that this estimator is a lower bound converging to the true mutual information $I(S'_j; A \mid S = s)$ as K increases (assuming, however, the true density $p(s'_j \mid s, a)$).

To compute the KL divergence itself, we make the simplifying assumption that the transition distribution $p(s'_j \mid s, a)$ is normally distributed given the action, which is reasonable in the robotics environment we are targeting. This allows us to estimate the KL without expensive MC sampling by using an approximation for mixtures of Gaussians from Durrieu et al. [52]. We detail the exact formula we use in Suppl. A.4.

With the normality assumption, the density itself can be learned using a probabilistic neural network and simple maximum likelihood estimation. That is, we parametrize $p(s'_j \mid s, a)$ as $\mathcal{N}(s'_j; \mu_\theta(s, a), \sigma^2_\theta(s, a))$, where $\mu_\theta, \sigma^2_\theta$ are the outputs of a neural network $f_\theta(s, a)$. We find the parameters θ by minimizing the negative log-likelihood over samples $\mathcal{D} = \{(s^{(i)}, a^{(i)}, s'^{(i)})\}_{i=1}^N$ collected by some policy (the univariate case shown here also extends to the multivariate case):

$$\theta^* = \arg\min_\theta \frac{1}{N}\sum_{i=1}^{N} \frac{\big(s'^{(i)}_j - \mu_\theta\big(s^{(i)}, a^{(i)}\big)\big)^2}{2\sigma^2_\theta\big(s^{(i)}, a^{(i)}\big)} + \frac{1}{2}\log \sigma^2_\theta\big(s^{(i)}, a^{(i)}\big). \tag{5}$$

There are some intricacies regarding the policy that collects the data for model training and the sampling policy π that is used to compute CAI. First of all, the two policies need to have overlapping support to avoid evaluating the model under actions never seen during training. Furthermore, if the data policy is different from the sampling policy π, the model is biased to some degree. This suggests to use π for collecting the data; however, as we use a random policy, this will not result in interesting data in most environments. The bias can be reduced by sampling actions from π during data collection with some probability and only train on those. In practice, however, we find to obtain better performing models by training on all data despite potentially being biased.

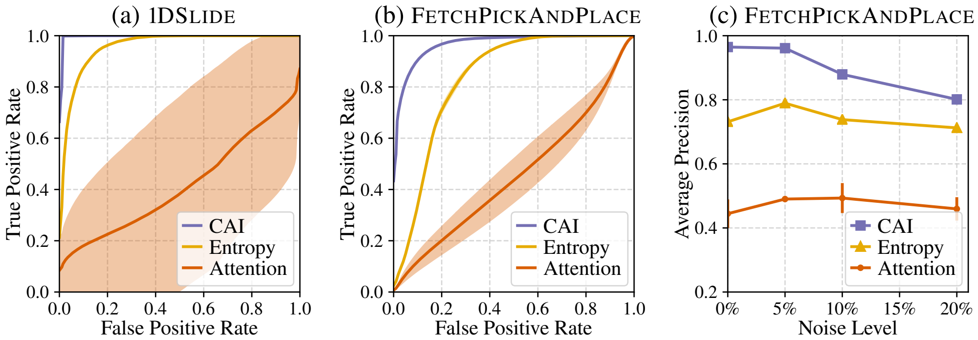


Figure 2: Causal influence detection performance. (a, b) ROC curves on 1DSLIDE and FETCHPICKANDPLACE environments. (c) Average precision for FETCHPICKANDPLACE depending on added state noise. Noise level is given as percentage of one standard deviation over the dataset.

Table 1: Results for evaluating causal influence detection on different environments. We measure area under the ROC curve (AUC), average precision (AP), and the best achievable F-score (F_1).

	1DSLIDE			**FETCHPICKANDPLACE**		
	AUC	AP	F_1	AUC	AP	F_1
CAI (ours)	1.00 ± 0.00	0.98 ± 0.00	0.95 ± 0.01	0.97 ± 0.01	0.96 ± 0.00	0.89 ± 0.00
Entropy	0.96 ± 0.00	0.47 ± 0.01	0.50 ± 0.01	0.84 ± 0.00	0.73 ± 0.00	0.78 ± 0.00
Attention [24]	0.42 ± 0.31	0.13 ± 0.14	0.18 ± 0.17	0.46 ± 0.06	0.44 ± 0.04	0.62 ± 0.00
Contacts	0.89	0.78	0.88	0.79	0.77	0.73

5 Empirical Evaluation of Causal Influence Detection

In this section, we evaluate the quality of our proposed causal influence detection approach in relevant environments. As a simple test case, we designed an environment (1DSLIDE) in which the agent must slide an object to a goal location by colliding with it. Furthermore, we test on the FETCHPICKANDPLACE environment from OpenAI Gym [53], in its original setting and when adding Gaussian noise to the observations to simulate more real-world conditions. In both environments, the target variables of interest are the coordinates of the object. Note that we need the true causal graph at each time step for the evaluation. For 1DSLIDE, we derive this information from the simulation. For the pick and place environment with its non-trivial dynamics, we resort to a heuristic of the possible movement range of the robotic arm in one step. Detailed information about the setup is provided in Suppls. B and E.

For our method, we use CAI estimated according to Eq. 4 (with $K = 64$) as a classification score that is thresholded to gain a binary decision. We compare with a recently proposed method [24] that uses the attention weights of a Transformer model [54] to model influence. Moreover, we compare with an *Entropy* baseline that uses $H(S'_j \mid s)$ as a score and a *Contact* baseline based on binary contact information from the simulator. We show the test results over 5 random seeds in Table 1 and Fig. 2. We observe that CAI is able to reliably detect causal influence and no other baseline is able to do so. When increasing the observation noise, the performance drops gracefully for CAI as shown in Fig. 2c. Suppl. C contains more experimental results, including a visualization of CAI's behavior.

6 Improving Efficiency in Reinforcement Learning

Having established the efficacy of our causal action influence (CAI) measure, we now develop several approaches to use it to improve RL algorithms.We will empirically verify the following claims in robotic manipulation environments: CAI improves sample efficiency and performance by (i) better state exploration through an exploration bonus, (ii) causal action exploration, and (iii) prioritizing experiences with causal influence during training.

We consider the environments FETCHPUSH, FETCHPICKANDPLACE from OpenAI Gym [55], and FETCHROTTABLE which is our modification containing a rotating table (explained in Suppl. B.3).

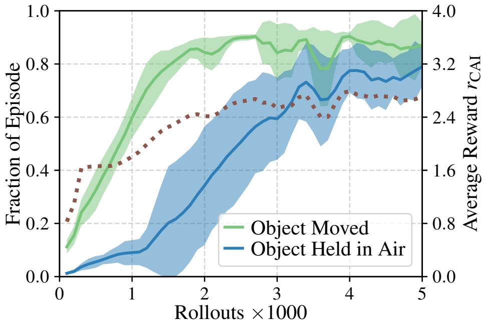


Figure 3: Intrinsically motivated learning on FETCHPICKANDPLACE. The reward is only r_{CAI} measured on the object coordinates.

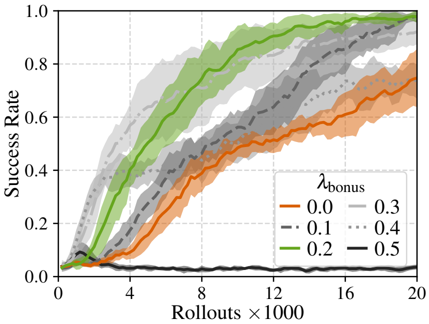


Figure 4: *Exploration bonus* improves performance in FETCHPICKANDPLACE. Sensitivity to the bonus reward scale λ_{bonus}.

These environments are goal-conditioned RL tasks with sparse rewards, meaning that each episode, a new goal is provided and the agent only receives a distinct reward upon reaching it. We use DDPG [56] with hindsight experience replay (HER) [57] as the base RL algorithm, a combination that achieves state-of-the-art results in these environment. The influence detection model is trained online on the data collected from an RL agent learning to solve its task. Since our measure C^j requires an entity of interest, we choose the coordinates of the object (as S_j). In all experiments, we report the mean success rate with standard deviation over 10 random seeds. More information about the experimental settings can be found in Suppl. F.

6.1 Intrinsic Motivation to Seek Influence

Causal Action Influence as Reward Bonus. We hypothesize that it is useful for an agent to be intrinsically motivated to gain control over its environment. We test this hypothesis by letting the agent maximize the causal influence it has over entities of interest. This can be achieved by using our influence measure as a reward signal. The reward signal can be used on its own, as an intrinsic motivation-type objective, or in conjunction with a task-specific reward as an exploration bonus. In the former case, we expect the agent to discover useful behaviors that can help it master task-oriented skills afterwards; in the latter case, we expect learning efficiency to improve, especially in sparse extrinsic reward scenarios. Concretely, for a state s, we define the bonus as $r_{\text{CAI}}(s) = C^j(s)$, and the total reward as $r(s) = r_{\text{task}}(s) + \lambda_{\text{bonus}}\, r_{\text{CAI}}(s)$, where $r_{\text{task}}(s)$ is the task reward, and λ_{bonus} is a hyperparameter.

Experiment on Intrinsically Motivated Learning. We first test the behavior of the agent in the absence of any task-specific reward on the FETCHPICKANDPLACE environment. Interestingly, the agent learned to grasp, lift, and hold the object in the air already after 2000 episodes, as shown in Fig. 3. The results demonstrate that encouraging causal control over the environment is well suited to prepare the agent for further tasks it might have to solve.

Impact of CAI Reward Bonus. Second, we are interested in the impact of adding an exploration bonus. In Fig. 4, we present results on the FETCHPICKANDPLACE environment when varying the reward scale λ_{bonus}. Naturally, the exploration bonus needs to be selected in the appropriate scale as a value too high will make it dominate the task reward. If selected correctly, the sample efficiency is improved drastically; for example, we find that the agent reaches a success rate of 60% four-times faster than the baseline (DDPG+HER) without any bonus ($\lambda_{\text{bonus}} = 0$).

6.2 Actively Exploring Actions with Causal Influence

Following Actions with the Most Causal Influence. Exploration via bonus rewards favors the re-visitation of already seen states. An alternative approach to exploration uses pro-active planning to choose exploratory actions. In our case, we can make use of our learned influence estimator to pick actions which we expect will have the largest causal effect on the agent's surroundings. From a causal viewpoint, the resulting agent can be seen as an experimenter that performs planned interventions in the environment to verify its beliefs. Should the actual outcome differ from the expected outcome, subsequent model updates can integrate the new data to self-correct the causal influence estimator.

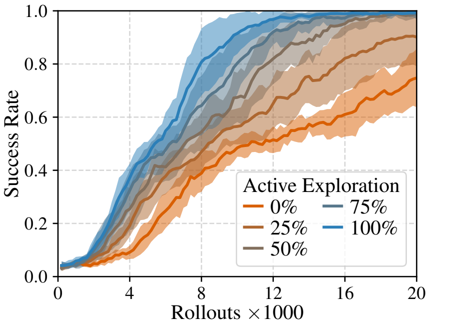


Figure 5: Performance of *active exploration* in FETCHPICKANDPLACE depending on the fraction of exploratory actions chosen actively (Eq. 6) from a total of 30% exploratory actions.

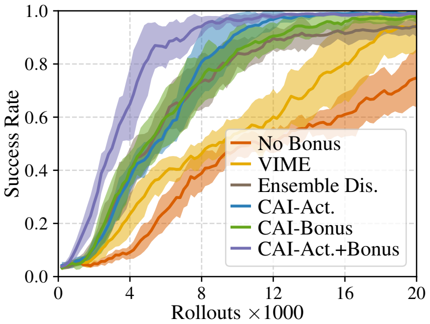


Figure 6: Experiment comparing exploration strategies on FETCHPICKANDPLACE. The combination of active exploration and reward bonus yields the largest sample efficiency.

Concretely, given the agent being in state s, we choose the action that has the largest contribution to the empirical mean in Eq. 4:

$$a^* = \underset{a \in \{a^{(1)}, \dots, a^{(K)}\}}{\arg\max} \ \mathrm{D}_{\mathrm{KL}}\left(p(s'_j \mid s, a) \;\middle\|\; \frac{1}{K}\sum_{k=1}^{K} p(s'_j \mid s, a^{(k)}) \right), \tag{6}$$

with $\{a^{(1)}, \dots, a^{(K)}\} \overset{\text{iid}}{\sim} \pi$. Intuitively, the selected action will be the one which results in maximal deviation from the expected outcome under all actions. For states s where the the agent is not in control, i.e. $C^j(s) \approx 0$, the action selection is uniform at random.

Active Exploration in Practice. Can active exploration replace ϵ-greedy exploration? To gain insights, we study the impact of the fraction of actively chosen exploration actions. For every exploratory action (ϵ is 30% in our experiments), we choose an action according to Eq. 6 the specified fraction of the time, and otherwise a random action. Figure 5 shows that any amount of active exploration improves over simple random exploration. Active causal action exploration can improve the sample efficiency roughly by a factor of two.

Combined CAI Exploration. We also present the combination of reward bonus and active exploration and compare our method with VIME, another exploration scheme based on information-theoretic measures [33]. In contrast to our method, VIME maximizes the information gain about the state transition dynamics. Further, we compare to ensemble disagreement [58], which in effect minimizes epistemic uncertainty about the transition dynamics. We compare different variants of VIME and ensemble disagreement in Suppl. D, and display only their best versions here. Figure 6 quantifies the superiority of all CAI variants (with ensemble disagreement as a viable alternative) and shows that combining the two exploration strategies compounds to increase sample efficiency even further. In the figure, CAI uses 100% active exploration and $\lambda_{\text{bonus}} = 0.2$ as the bonus reward scale.

6.3 Causal Influence-based Experience Replay

Prioritizing According to Causal Influence. We will now propose another method using CAI, namely to inform the choice of samples replayed to the agent during off-policy training. Typically, past states are sampled uniformly for learning. Intuitively, focusing on those states where the agent has control over the object of interest (as measured by CAI) should improve the sample efficiency. We can implement this idea using a prioritization scheme that samples past episodes in which the agent had more influence more frequently. Concretely, we define the probability $P^{(i)}$ of sampling any state from episode i (of M episodes) in the replay buffer as

$$P^{(i)} = \frac{p^{(i)}}{\sum_{i=1}^{M} p^{(i)}} \cdot \frac{1}{T}, \qquad \text{with} \qquad p^{(i)} = \left(M + 1 - \underset{\mathrm{i}}{\mathrm{rank}} \sum_{t=1}^{T} C^j\big(s^{(t)}\big) \right)^{-1}. \tag{7, 8}$$

where T is the episode length, and $p^{(i)}$ is the priority of episode i. The priority of an episode i is based on the (inverse) rank of the episode (rank_i) when sorting all M episodes according to their total influence (i.e. sum of state influences). We call this *causal action influence prioritization* (CAI-P).

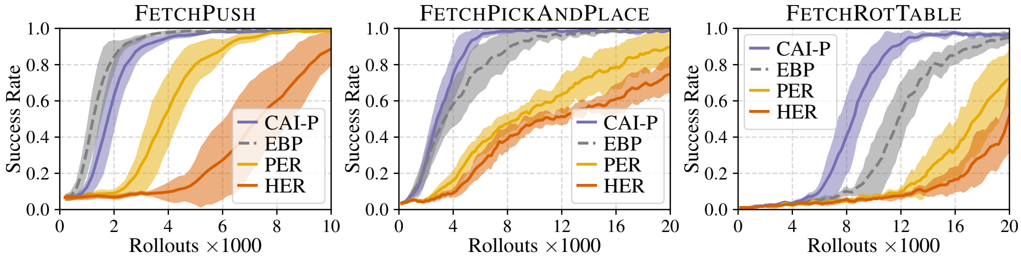


Figure 7: Prioritizing experience replay in different manipulation environments. Comparison of causal action influence prioritization (CAI-P) against baselines: the energy-based method (EBP) [60] with privileged information, prioritized experience replay (PER) [59], and HER without prioritization.

This scheme is similar to Prioritized Experience Replay [59], with two differences: instead of using the TD error for prioritization, we use the causal influence measure. Furthermore, instead of prioritizing individual states, we prioritize episodes and sample states uniformly within episodes. This is because the information about the return that can be achieved from an influence state still needs to be propagated back to non-influenced states by TD updates, which requires sampling them.

Influence-Based Prioritization in Manipulation Tasks. We compare our influence-based prioritization (CAI-P) against no prioritization in hindsight experience replay (HER) (a strong baseline for multi-goal RL), and two other prioritization schemes: prioritized experience replay (PER) [59] and energy-based prioritization (EBP) [60]. Especially EBP is a strong method for the environments we are considering as it uses privileged knowledge of the underlying physics to replay episodes based on the amount of energy that is transferred from agent to the object to manipulate. All prioritization variants are equipped with HER as well. The FetchRotTable environment, shown in Fig. 8, is an interesting test bed as the object can move through the table rotation without the control of the agent. The results, shown in Fig. 7, reveal that causal influence prioritization can speed up learning drastically. Our method is on par or better than the energy-based (oracle) method EBP and improves over PER by a factor of 1.5–2.5 in learning speed (at 60% success rate). Finally, in Suppl. D, we combine all our proposed improvements and show that FetchPickAndPlace can be solved up to 95% success rate in just 3000 episodes.

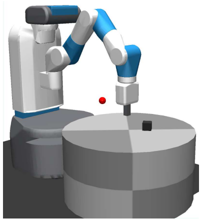


Figure 8: FetchRotTable. The table rotates periodically.

7 Discussion

In this work, we show how situation-dependent causal influence detection can help improve reinforcement learning agents. To this end, we derive a measure of local causal action influence (CAI) and introduce a data-driven approach based on neural network models to estimate it. We showcase using CAI as an exploration bonus, as a way to perform active action exploration, and to prioritize in experience replay. Each of our applications yields strong improvements in sample efficiency. We expect that there are further ways to use our causal measure in RL, e.g. for credit assignment.

Our work has several limitations. First, we assume full observability of the state, which simplifies the causal inference problem as there is no confounding between an agent's action and its effect. Under partial observability, our approach could still be applicable using latent variable models [61]. Second, we require an available factorization of the state into causal variables. The problem of automatically learning causal variables from high-dimensional data is open [4] and our method would likely benefit from advances in this field. Third, the accurate estimation of our measure relies on a correct model. We found that deep networks can struggle at times to pick up the causal relationship between actions and entities. How to design models with appropriate inductive biases for cause-effect inference is an open question [3, 4, 62].

An intriguing future direction is to extend our work to influence detection between entities, a prerequisite for identifying multi-step influences of the agent on the environment. Being able to model such indirect interventions would bring us closer to "artificial scientists" – agents that can perform planned experiments to reveal the latent causal structure of the world.

Acknowledgments and Disclosure of Funding

The authors thank Andrii Zadaianchuk and Dominik Zietlow for many helpful discussions and providing feedback on the text. Furthermore, the authors would also like to thank Sebastian Blaes for creating the FETCHROTTABLE environment. The authors thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting Maximilian Seitzer. GM and BS are members of the Machine Learning Cluster of Excellence, EXC number 2064/1 – Project number 390727645. We acknowledge the financial support from the German Federal Ministry of Education and Research (BMBF) through the Tübingen AI Center (FKZ: 01IS18039B).

References

[1] Marcin Andrychowicz, Bowen Baker, Maciek Chociej, Rafal Jozefowicz, Bob McGrew, Jakub Pachocki, Arthur Petron, Matthias Plappert, Glenn Powell, Alex Ray, et al. Learning dexterous in-hand manipulation. *The International Journal of Robotics Research*, 39(1):3–20, 2020.

[2] Dmitry Kalashnikov, Alex Irpan, Peter Pastor Sampedro, Julian Ibarz, Alexander Herzog, Eric Jang, Deirdre Quillen, Ethan Holly, Mrinal Kalakrishnan, Vincent Vanhoucke, and Sergey Levine. Qt-opt: Scalable deep reinforcement learning for vision-based robotic manipulation. In *Conference on Robot Learning 2018*, 2018. URL `https://arxiv.org/pdf/1806.10293`.

[3] Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of Causal Inference: Foundations and Learning Algorithms*. MIT Press, Cambridge, MA, USA, 2017.

[4] B. Schölkopf, F. Locatello, S. Bauer, R. Nan Ke, N. Kalchbrenner, A. Goyal, and Y. Bengio. Towards causal representation learning. *Proceedings of the IEEE*, 2021. doi: 10.1109/JPROC.2021.3058954.

[5] Giambattista Parascandolo, Niki Kilbertus, Mateo Rojas-Carulla, and Bernhard Schölkopf. Learning independent causal mechanisms. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 4036–4044. PMLR, 10–15 Jul 2018. URL `http://proceedings.mlr.press/v80/parascandolo18a.html`.

[6] Yoshua Bengio, Tristan Deleu, Nasim Rahaman, Nan Rosemary Ke, Sebastien Lachapelle, Olexa Bilaniuk, Anirudh Goyal, and Christopher Pal. A meta-transfer objective for learning to dis entangle causal mechanisms. In *8th International Conference on Learning Representations (ICLR)*, 2020. URL `https://openreview.net/forum?id=ryxWIgBFPS`.

[7] Francesco Locatello, Ben Poole, Gunnar Raetsch, Bernhard Schölkopf, Olivier Bachem, and Michael Tschannen. Weakly-supervised disentanglement without compromises. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 6348–6359. PMLR, 13–18 Jul 2020. URL `http://proceedings.mlr.press/v119/locatello20a.html`.

[8] Nan Rosemary Ke, Olexa Bilaniuk, Anirudh Goyal, Stefan Bauer, Hugo Larochelle, Bernhard Schölkopf, Michael C. Mozer, Chris Pal, and Yoshua Bengio. Learning neural causal models from unknown interventions. *ArXiv*, abs/1910.01075, 2019.

[9] A. Goyal, A. Lamb, J. Hoffmann, S. Sodhani, S. Levine, Y. Bengio, and B. Schölkopf. Recurrent independent mechanisms. In *9th International Conference on Learning Representations (ICLR)*, 2021.

[10] Judea Pearl. *Causality: Models, Reasoning and Inference*. Cambridge University Press, USA, 2nd edition, 2009.

[11] P. Spirtes, C. Glymour, and R. Scheines. *Causation, Prediction, and Search*. MIT Press, Cambridge, MA, 2nd edition, 2000.

[12] J. Peters, J. Mooij, D. Janzing, and B. Schölkopf. Identifiability of causal graphs using functional models. In F. G. Cozman and A. Pfeffer, editors, *27th Conference on Uncertainty in Artificial Intelligence*, pages 589–598, Corvallis, OR, 2011. AUAI Press.

[13] Michael Eichler. *Causal Inference in Time Series Analysis*. Wiley, 2012. ISBN 9781119945710. doi: 10.1002/9781119945710.ch22.

[14] Thomas Schreiber. Measuring information transfer. *Physical Review Letters*, 85 2:461–4, 2000.

[15] Nihat Ay and David C. Krakauer. Geometric robustness theory and biological networks. *Theory in Biosciences*, 125(2):93–121, 2007. ISSN 1431-7613. doi: https://doi.org/10.1016/j.thbio.2006.06.002. URL `https://www.sciencedirect.com/science/article/pii/S1431761306000255`.

[16] N. Ay and D. Polani. Information flows in causal networks. *Adv. Complex Syst.*, 11:17–41, 2008.

[17] Joseph T. Lizier. *The local information dynamics of distributed computation in complex systems*. PhD thesis, University of Sydney, 2013. URL `http://d-nb.info/1024629171`.

[18] D. Janzing, D. Balduzzi, M. Grosse-Wentrup, and B. Schölkopf. Quantifying causal influences. *Annals of Statistics*, 41(5):2324–2358, 2013. URL `http://projecteuclid.org/euclid.aos/1383661266`.

[19] E. Bareinboim, A. Forney, and J. Pearl. Bandits with unobserved confounders: A causal approach. In *Advances in Neural Information Processing*, volume 28. Curran Associates, Inc., 2015.

[20] Chaochao Lu, B. Schölkopf, and José Miguel Hernández-Lobato. Deconfounding reinforcement learning in observational settings. *ArXiv*, abs/1812.10576, 2018.

[21] Danilo Jimenez Rezende, Ivo Danihelka, George Papamakarios, N. Ke, Ray Jiang, T. Weber, K. Gregor, Hamza Merzic, Fabio Viola, J. Wang, Jovana Mitrovic, F. Besse, Ioannis Antonoglou, and Lars Buesing. Causally Correct Partial Models for Reinforcement Learning. *ArXiv*, abs/2002.02836, 2020.

[22] Lars Buesing, T. Weber, Yori Zwols, Sébastien Racanière, A. Guez, J. Lespiau, and N. Heess. Woulda, Coulda, Shoulda: Counterfactually-Guided Policy Search. In *7th International Conference on Learning Representations (ICLR)*, 2019.

[23] J. Zhang and E. Bareinboim. Near-optimal reinforcement learning in dynamic treatment regimes. In *Advances in Neural Information Processing Systems*, volume 33. Curran Associates, Inc., 2019.

[24] Silviu Pitis, Elliot Creager, and Animesh Garg. Counterfactual data augmentation using locally factored dynamics. In *Advances in Neural Information Processing Systems*, volume 33. Curran Associates, Inc., 2020.

[25] J. Schmidhuber. Curious model-building control systems. In *Proceedings IEEE International Joint Conference on Neural Networks*, pages 1458–1463 vol.2, 1991. doi: 10.1109/IJCNN.1991.170605.

[26] Deepak Pathak, Pulkit Agrawal, Alexei A. Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 488–489, 2017.

[27] Sebastian Blaes, Marin Vlastelica, Jia-Jie Zhu, and Georg Martius. Control What You Can: Intrinsically motivated task-planning agent. In *Advances in Neural Information Processing*, pages 12520–12531. Curran Associates, Inc., 2019. URL `http://papers.nips.cc/paper/9418-control-what-you-can-intrinsically-motivated-task-planning-agent.pdf`.

[28] Cédric Colas, Pierre-Yves Oudeyer, Olivier Sigaud, Pierre Fournier, and Mohamed Chetouani. CURIOUS: intrinsically motivated modular multi-goal reinforcement learning. In *International Conference on Machine Learning (ICML'19)*, pages 1331–1340, 2019.

[29] A.S. Klyubin, D. Polani, and C.L. Nehaniv. Empowerment: a universal agent-centric measure of control. In *2005 IEEE Congress on Evolutionary Computation*, volume 1, pages 128–135 Vol.1, 2005. doi: 10.1109/CEC.2005.1554676.

[30] Shakir Mohamed and Danilo Jimenez Rezende. Variational information maximisation for intrinsically motivated reinforcement learning. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015. URL `https://proceedings.neurips.cc/paper/2015/file/e00406144c1e7e35240afed70f34166a-Paper.pdf`.

[31] Jan Storck, Sepp Hochreiter, and Jürgen Schmidhuber. Reinforcement driven information acquisition in non-deterministic environments. In *ICANN'95*, pages 159–164, 1995.

[32] Daniel Little and Friedrich Sommer. Learning and exploration in action-perception loops. *Frontiers in Neural Circuits*, 7:37, 2013. ISSN 1662-5110. doi: 10.3389/fncir.2013.00037. URL `https://www.frontiersin.org/article/10.3389/fncir.2013.00037`.

[33] Rein Houthooft, Xi Chen, Xi Chen, Yan Duan, John Schulman, Filip De Turck, and Pieter Abbeel. Vime: Variational information maximizing exploration. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL `https://proceedings.neurips.cc/paper/2016/file/abd815286ba1007abfbb8415b83ae2cf-Paper.pdf`.

[34] Georg Martius, Ralf Der, and Nihat Ay. Information driven self-organization of complex robotic behaviors. *PLoS ONE*, 8(5):e63400, 2013. doi: 10.1371/journal.pone.0063400. URL `http://dx.doi.org/10.1371/journal.pone.0063400`.

[35] Keyan Zahedi, Georg Martius, and Nihat Ay. Linear combination of one-step predictive information with an external reward in an episodic policy gradient setting: a critical analysis. *Frontiers in Psychology*, 4(801), 2013. ISSN 1664-1078. doi: 10.3389/fpsyg.2013.00801. URL `http://www.frontiersin.org/cognitive_science/10.3389/fpsyg.2013.00801/abstract`.

[36] S. Sontakke, A. Mehrjou, L. Itti, and B. Schölkopf. Causal curiosity: Rl agents discovering self-supervised experiments for causal representation learning. In *Proceedings of 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 9848–9858. PMLR, July 2021. URL `https://proceedings.mlr.press/v139/sontakke21a.html`.

[37] Rui Zhao, Yang Gao, Pieter Abbeel, Volker Tresp, and Wei Xu. Mutual information state intrinsic control. In *International Conference on Learning Representations*, 2021. URL `https://openreview.net/forum?id=OthEq8I5v1`.

[38] John S. Watson. The development and generalization of contingency awareness in early infancy: Some hypotheses. *Merrill-Palmer Quarterly of Behavior and Development*, 12(2):123–135, 1966. ISSN 00260150. URL `http://www.jstor.org/stable/23082793`.

[39] Jongwook Choi, Yijie Guo, Marcin Moczulski, Junhyuk Oh, Neal Wu, Mohammad Norouzi, and Honglak Lee. Contingency-aware exploration in reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2019. URL `https://openreview.net/forum?id=HyxGB2AcY7`.

[40] Yuhang Song, Jianyi Wang, Thomas Lukasiewicz, Zhenghua Xu, Shangtong Zhang, Andrzej Wojcicki, and Mai Xu. Mega-Reward: Achieving human-level play without extrinsic rewards. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04):5826–5833, Apr. 2020. doi: 10.1609/aaai.v34i04.6040. URL `https://ojs.aaai.org/index.php/AAAI/article/view/6040`.

[41] Joseph Y. Halpern. *Actual Causality*. The MIT Press, 2016. ISBN 9780262035026.

[42] J. Pearl, M. Glymour, and N. Jewell. *Causal Inference in Statistics: A Primer*. Wiley, 2016. ISBN 978-1-119-18684-7.

[43] Hugo Touchette and Seth Lloyd. Information-theoretic approach to the study of control systems. *Physica A: Statistical Mechanics and its Applications*, 331(1):140–172, 2004. ISSN 0378-4371. doi: https://doi.org/10.1016/j.physa.2003.09.007. URL `https://www.sciencedirect.com/science/article/pii/S0378437103008100`.

[44] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley Series in Telecommunications and Signal Processing. Wiley-Interscience, 2nd ed edition, 2006.

[45] C. W. J. Granger. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica*, 37(3):424–438, 1969. URL `http://www.jstor.org/stable/1912791`.

[46] Christoph Salge, Cornelius Glackin, and Daniel Polani. *Empowerment–An Introduction*, pages 67–114. Springer Berlin Heidelberg, Berlin, Heidelberg, 2014. ISBN 978-3-642-53734-9. doi: 10.1007/978-3-642-53734-9_4. URL `https://doi.org/10.1007/978-3-642-53734-9_4`.

[47] Alexander Kraskov, Harald Stögbauer, and Peter Grassberger. Estimating mutual information. *Physical review. E, Statistical, nonlinear, and soft matter physics*, 69:066138, 07 2004. doi: 10.1103/PhysRevE.69.066138.

[48] Jakob Runge. Conditional independence testing based on a nearest-neighbor estimator of conditional mutual information. In Amos Storkey and Fernando Perez-Cruz, editors, *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84 of *Proceedings of Machine Learning Research*, pages 938–947. PMLR, 09–11 Apr 2018. URL `http://proceedings.mlr.press/v84/runge18a.html`.

[49] Kirthevasan Kandasamy, Akshay Krishnamurthy, Barnabas Poczos, Larry Wasserman, and james m robins. Nonparametric von Mises estimators for entropies, divergences and mutual informations. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015. URL `https://proceedings.neurips.cc/paper/2015/file/06138bc5af6023646ede0e1f7c1eac75-Paper.pdf`.

[50] Shuyang Gao, Greg Ver Steeg, and Aram Galstyan. Efficient Estimation of Mutual Information for Strongly Dependent Variables. In Guy Lebanon and S. V. N. Vishwanathan, editors, *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, volume 38 of *Proceedings of Machine Learning Research*, pages 277–286, San Diego, California, USA, 09–12 May 2015. PMLR. URL `http://proceedings.mlr.press/v38/gao15.html`.

[51] Ben Poole, Sherjil Ozair, Aaron Van Den Oord, Alex Alemi, and George Tucker. On variational bounds of mutual information. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 5171–5180. PMLR, 09–15 Jun 2019. URL `http://proceedings.mlr.press/v97/poole19a.html`.

[52] Jean-Louis Durrieu, J. Thiran, and Finnian Kelly. Lower and upper bounds for approximation of the kullback-leibler divergence between gaussian mixture models. *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4833–4836, 2012.

[53] Greg Brockman, Vicki Cheung, Ludwig Pettersson, J. Schneider, John Schulman, Jie Tang, and W. Zaremba. OpenAI Gym. *ArXiv*, abs/1606.01540, 2016.

[54] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL `https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf`.

[55] Matthias Plappert, Marcin Andrychowicz, Alex Ray, Bob McGrew, Bowen Baker, Glenn Powell, J. Schneider, Joshua Tobin, Maciek Chociej, P. Welinder, V. Kumar, and W. Zaremba. Multi-goal reinforcement learning: Challenging robotics environments and request for research. *ArXiv*, abs/1802.09464, 2018.

[56] T. Lillicrap, Jonathan J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2016.

[57] Marcin Andrychowicz, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, OpenAI Pieter Abbeel, and Wojciech Zaremba. Hindsight experience replay. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[58] Deepak Pathak, Dhiraj Gandhi, and Abhinav Gupta. Self-supervised exploration via disagreement. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 5062–5071. PMLR, 09–15 Jun 2019. URL `https://proceedings.mlr.press/v97/pathak19a.html`.

[59] Tom Schaul, John Quan, Ioannis Antonoglou, and David Silver. Prioritized experience replay. In *International Conference on Learning Representations (ICLR)*, 2016.

[60] Rui Zhao and Volker Tresp. Energy-based hindsight experience prioritization. In Aude Billard, Anca Dragan, Jan Peters, and Jun Morimoto, editors, *Proceedings of The 2nd Conference on Robot Learning*, volume 87 of *Proceedings of Machine Learning Research*, pages 113–122. PMLR, 29–31 Oct 2018.

[61] Christos Louizos, Uri Shalit, Joris M Mooij, David Sontag, Richard Zemel, and Max Welling. Causal effect inference with deep latent-variable models. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL `https://proceedings.neurips.cc/paper/2017/file/94b5bde6de888ddf9cde6748ad2523d1-Paper.pdf`.

[62] Anirudh Goyal and Yoshua Bengio. Inductive biases for deep learning of higher-level cognition. *ArXiv*, abs/2011.15091, 2020.

[63] John R. Hershey and Peder A. Olsen. Approximating the Kullback Leibler divergence between gaussian mixture models. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07*, volume 4, pages IV–317–IV–320, 2007. doi: 10.1109/ICASSP.2007.366913.

[64] E. Todorov, T. Erez, and Y. Tassa. MuJoCo: A physics engine for model-based control. *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033, 2012.

[65] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015.

[66] Andrew M. Saxe, James L. McClelland, and S. Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. In *International Conference on Learning Representations (ICLR)*, 2014.

[67] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Y. Yoshida. Spectral normalization for generative adversarial networks. In *International Conference on Learning Representations (ICLR)*, 2018.

[68] Zhizhou Ren, Kefan Dong, Yuan Zhou, Qiang Liu, and Jian Peng. Exploration via hindsight goal generation. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d. Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL `https://proceedings.neurips.cc/paper/2019/file/57db7d68d5335b52d5153a4e01adaa6b-Paper.pdf`.

[69] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *AISTATS*, 2010.

Appendix C

Bridging the Gap to Real-World Object-centric Learning

The following 15 pages have been published as:

Maximilian Seitzer, Max Horn, Andrii Zadaianchuk, Dominik Zietlow, Tianjun Xiao, Carl-Johann Simon-Gabriel, Tong He, Zheng Zhang, Bernhard Schölkopf, Thomas Brox, and Francesco Locatello (2023). "Bridging the Gap to Real-World Object-Centric Learning". In: *International Conference on Learning Representations (ICLR)*

A summary of this publication can be found in Chap. 6. The supplementary material to this publication is available in the online version.

BRIDGING THE GAP TO REAL-WORLD OBJECT-CENTRIC LEARNING

Maximilian Seitzer[1,†] **Max Horn**[2] **Andrii Zadaianchuk**[1,3,†] **Dominik Zietlow**[2]
Tianjun Xiao[2] **Carl-Johann Simon-Gabriel**[2] **Tong He**[2] **Zheng Zhang**[2]
Bernhard Schölkopf[2] **Thomas Brox**[2] **Francesco Locatello**[2]
[1]Max-Planck Institute for Intelligent Systems, Tübingen, Germany
[2]Amazon Web Services
[3]Department of Computer Science, ETH Zürich

ABSTRACT

Humans naturally decompose their environment into entities at the appropriate level of abstraction to act in the world. Allowing machine learning algorithms to derive this decomposition in an unsupervised way has become an important line of research. However, current methods are restricted to simulated data or require additional information in the form of motion or depth in order to successfully discover objects. In this work, we overcome this limitation by showing that reconstructing features from models trained in a self-supervised manner is a sufficient training signal for object-centric representations to arise in a fully unsupervised way. Our approach, DINOSAUR, significantly out-performs existing image-based object-centric learning models on simulated data and is the first unsupervised object-centric model that scales to real-world datasets such as COCO and PASCAL VOC. DINOSAUR is conceptually simple and shows competitive performance compared to more involved pipelines from the computer vision literature.

1 INTRODUCTION

Object-centric representation learning has the potential to greatly improve generalization of computer vision models, as it aligns with causal mechanisms that govern our physical world (Schölkopf et al., 2021; Dittadi et al., 2022). Due to the compositional nature of scenes (Greff et al., 2020), object-centric representations can be more robust towards out-of-distribution data (Dittadi et al., 2022) and support more complex tasks like reasoning (Assouel et al., 2022; Yang et al., 2020) and control (Zadaianchuk et al., 2020; Mambelli et al., 2022; Biza et al., 2022). They are in line with studies on the characterization of human perception and reasoning (Kahneman et al., 1992; Spelke & Kinzler, 2007). Inspired by the seemingly unlimited availability of unlabeled image data, this work focuses on *unsupervised* object-centric representation learning.

Most unsupervised object-centric learning approaches rely on a reconstruction objective, which struggles with the variation in real-world data. Existing approaches typically implement "slot"-structured bottlenecks which transform the input into a set of object representations and a corresponding decoding scheme which reconstructs the input data. The emergence of object representations is primed by the set bottleneck of models like Slot Attention (Locatello et al., 2020) that groups together independently repeating visual patterns across a fixed data set. While this approach was successful on simple synthetic datasets, where low-level features like color help to indicate the assignment of pixels to objects, those methods have failed to scale to complex synthetic or real-world data (Eslami et al., 2016; Greff et al., 2019; Burgess et al., 2019; Locatello et al., 2020; Engelcke et al., 2021).

To overcome these limitations, previous work has used additional information sources, e.g. motion or depth (Kipf et al., 2022; Elsayed et al., 2022). Like color, motion and depth act as grouping signals when objects move or stand-out in 3D-space. Unfortunately, this precludes training on most real-world

†: Work done during an internship at Amazon Web Services.
Correspondence to: hornmax@amazon.de, maximilian.seitzer@tuebingen.mpg.de

image datasets, which do not include depth annotations or motion cues. Following deep learning's mantra of scale, another appealing approach could be to increase the capacity of the Slot Attention architecture. However, our experiments (Sec. 4.3) suggest that scale alone is *not* sufficient to close the gap between synthetic and real-world datasets. We thus conjecture that the image reconstruction objective on its own does not provide sufficient inductive bias to give rise to object groupings when objects have complex appearance. But instead of relying on auxiliary external signals, we introduce an additional inductive bias by reconstructing features that have a high level of homogeneity within objects. Such features can easily be obtained via recent self-supervised learning techniques like DINO (Caron et al., 2021). We show that combining such a feature reconstruction loss with existing grouping modules such as Slot Attention leads to models that significantly out-perform other image-based object-centric methods and *bridge the gap to real-world object-centric representation learning*. The proposed architecture `DINOSAUR` (**DINO** and **S**lot **A**ttention **U**sing **R**eal-world data) is conceptually simple and highly competitive with existing unsupervised segmentation and object discovery methods in computer vision.

2 Related Work

Our research follows a body of work studying the emergence of *object-centric representations* in neural networks trained end-to-end with certain architectural biases (Eslami et al., 2016; Burgess et al., 2019; Greff et al., 2019; Lin et al., 2020; Engelcke et al., 2020; Locatello et al., 2020; Singh et al., 2022a). These approaches implicitly define objects as repeating patterns across a closed-world dataset that can be discovered e.g. via semantic discrete- or set-valued bottlenecks. As the grouping of low-level features into object entities is often somewhat arbitrary (it depends for example on the scale and level of detail considered), recent work has explored additional information sources such as video (Kosiorek et al., 2018; Jiang et al., 2020; Weis et al., 2021; Singh et al., 2022b; Traub et al., 2023), optical flow (Kipf et al., 2022; Elsayed et al., 2022; Bao et al., 2022), text descriptions of the scene (Xu et al., 2022) or some form of object-location information (e.g. with bounding boxes) (Kipf et al., 2022). In contrast, we completely avoid additional supervision by leveraging the implicit inductive bias contained in the self-supervised features we reconstruct, which present a high level of homogeneity within objects (Caron et al., 2021). This circumvents the scalability challenges of previous works that rely on pixel similarity as opposed to perceptual similarity (Dosovitskiy & Brox, 2016) and enables object discovery on real-world data without changing the existing grouping modules. Our approach can be considered similar to SLATE (Singh et al., 2022a), but with the crucial difference of reconstructing *global* features from a Vision Transformer (Dosovitskiy et al., 2021) instead of *local* features from a VQ-VAE (van den Oord et al., 2017).

Challenging object-centric methods by *scaling dataset complexity* has been of recent interest: Karazija et al. (2021) propose ClevrTex, a textured variant of the popular CLEVR dataset, and show that previous object-centric models perform mostly poorly on it. Greff et al. (2022) introduce the MOVi datasets with rendered videos of highly realistic objects with complex shape and appearance. Arguably the most advanced synthetic datasets to date, we find that current state-of-the-art models struggle with them in the unsupervised setting. Finally, Yang & Yang (2022) show that existing image-based object-centric methods catastrophically fail on real-world datasets such as COCO, likely because they can not cope with the diversity of shapes and appearances presented by natural data. In contrast, we demonstrate that our approach works well on both complex synthetic and real-world datasets.

In the computer vision literature, structuring natural scenes without any human annotations has also enjoyed popularity, with tasks such as *unsupervised semantic segmentation* and *object localization*. Those tasks are interesting for us because they constitute established real-world benchmarks related to unsupervised object discovery, and we show that our method is also competitive on them. We refer to App. A for a detailed discussion of prior research in these areas.

3 Method

Our approach essentially follows the usual autoencoder-like design of object-centric models and is summarized in Figure 1: a first module extracts features from the input data (the encoder), a second module groups them into a set of latent vectors called *slots*, and a final one (the decoder) tries to reconstruct some target signal from the latents. However, our method crucially differs from other

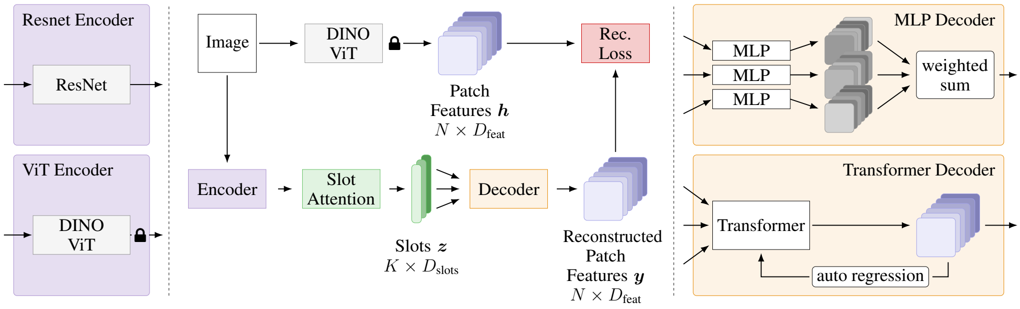


Figure 1: Overview of the proposed architecture DINOSAUR. The image is processed into a set of patch features $\boldsymbol{h}$ by a frozen DINO ViT model (pre-trained using the self-supervised DINO method) and encoded via either a ResNet or the DINO ViT. Slot attention groups the encoded features into a set of slots. The model is trained by reconstructing the DINO features from the slots, either independently per-slot (MLP decoder) or jointly via auto regression (Transformer decoder).

approaches in that instead of reconstructing the original inputs, the decoder is tasked to reconstruct *features from self-supervised pre-training*. We start with the discussion of this training signal in Sec. 3.1 and describe further architectural choices in Sec. 3.2.

3.1 Feature Reconstruction as a Training Signal

Why are models based on image reconstruction like Slot Attention not successful beyond simpler synthetic datasets? We hypothesize that reconstruction on the pixel level produces too weak of a signal for object-centricness to emerge; the task focuses (at least initially) strongly on low-level image features such as color statistics. This quickly decreases the reconstruction error, but the resulting model does not discover objects beyond datasets where objects are mostly determined by distinct object colors. Instead, if we had an (unsupervised) signal that required higher-level semantic information to reconstruct, there would be pressure on the slots to efficiently encode this information as well. Luckily, such signals can nowadays be easily obtained with self-supervised learning algorithms, which have been successful in learning powerful representations for vision tasks such as classification and object detection purely from images (Chen et al., 2020b; Grill et al., 2020; He et al., 2022). Thus, given K slots $\boldsymbol{z} \in \mathbb{R}^{K \times D_{\text{slots}}}$, the model is trained to reconstruct self-supervised features $\boldsymbol{h} \in \mathbb{R}^{N \times D_{\text{feat}}}$, by minimizing the following loss:

$$\mathcal{L}_{\text{rec}} = \|\boldsymbol{y} - \boldsymbol{h}\|^2, \qquad \boldsymbol{y} = \text{Decoder}(\boldsymbol{z}). \tag{1}$$

This loss can be viewed as a form of student-teacher knowledge distillation (Hinton et al., 2015), where the student has a particular form of bottleneck that condenses the high-dimensional, unstructured information contained in the teacher features into a lower-dimensional, structured form. We can also draw parallels between this loss and perceptual similarity losses for image generation (Dosovitskiy & Brox, 2016), that is, the optimization takes place in a space more semantic than pixel space.

For pre-training, we utilize the ImageNet dataset (Deng et al., 2009). From the student-teacher perspective, this means that the teacher additionally transports knowledge gained from a larger image collection to the (smaller) datasets at hand. It is well-known that using large datasets for pre-training can significantly improve performance, but to our knowledge, we are the first to exploit such transfer learning for object-centric learning. In general, studying the role additional *data* can play for object-centric learning is an interesting topic, but we leave that for future investigations.

Which self-supervised algorithm should we use? In our analysis (Sec. 4.3), we investigate several recent ones (DINO (Caron et al., 2021), MoCo-v3 (Chen et al., 2021), MSN (Assran et al., 2022), MAE (He et al., 2022)). Interestingly, we find that they all work reasonably well for the emergence of real-world object grouping. In the following, we mainly apply the DINO method (Caron et al., 2021), because of its good performance and accessibility in open source libraries (Wightman, 2019). We experiment with features from ResNets (He et al., 2015) and Vision Transformers (ViTs) (Dosovitskiy et al., 2021), and find that the latter yield significantly better results.

3.2 An Architecture for Real-World Object-Centric Learning

Encoder Previous work has shown that powerful feature extractors help in scaling object-centric methods to more complex data (Kipf et al., 2022). To this end, we experiment with two choices: a ResNet-34 encoder with increased spatial resolution used by Kipf et al. (2022), and Vision Transformers. Unfortunately, we were not able to optimize randomly initialized ViTs with our model, as training collapsed. Instead, we found it sufficient to initialize the ViT using weights from self-supervised pre-training, and keeping them fixed throughout training[1]. In terms of results, we find that the ResNet and the pre-trained ViT encoder perform similarly. However, the model converges faster with the pre-trained ViT, and it is also computationally more efficient: we can directly use the ViT outputs as the target features $\boldsymbol{h}$. Consequently, we mainly use the ViT encoder in the following.

Slot Attention Grouping The grouping stage of our model uses Slot Attention (Locatello et al., 2020) to turn the set of encoder features into a set of K slot vectors $\boldsymbol{z} \in \mathbb{R}^{K \times D_{\text{slots}}}$. This follows an iterative process where slots compete for input features using an attention mechanism, starting from randomly sampled initial slots. We largely use the original Slot Attention formulation (including GRU (Cho et al., 2014) and residual MLP modules), with one difference when using ViT features: we do not add positional encodings on the ViT features before Slot Attention, as we found the ViT's initial position encodings to be sufficient to support spatial grouping of the features. Additionally, we add a small one-hidden-layer MLP that transforms each encoder feature before Slot Attention.

Feature Decoding As we apply feature instead of image reconstruction as the training objective, we need a decoder architecture suitable for this purpose. To this end, we consider two different designs: a MLP decoder that is applied independently to each slot, and a Transformer decoder (Vaswani et al., 2017) that autoregressively reconstructs the set of features. We describe both options in turn.

The *MLP decoder* follows a similar design as the commonly used spatial broadcast decoder (Watters et al., 2019). Each slot is first broadcasted to the number of patches, resulting in a set of N tokens for each slot. To make the spatial positions of the tokens identifiable, a learned positional encoding is added to each token. The tokens for each slot are then processed token-wise by the same MLP, producing the reconstruction $\hat{\boldsymbol{y}}_k$ for slot k, plus an alpha map $\boldsymbol{\alpha}_k$ that signifies where the slot is active. The final reconstruction $\boldsymbol{y} \in \mathbb{R}^{N \times D_{\text{feat}}}$ is formed by taking a weighted sum across the slots:

$$\boldsymbol{y} = \sum_{k=1}^{K} \hat{\boldsymbol{y}}_k \odot \boldsymbol{m}_k, \qquad \boldsymbol{m}_k = \operatorname*{softmax}_k \boldsymbol{\alpha}_k \tag{2}$$

The advantage of this simple design is its computational efficiency: as the MLP is shared across slots and positions, decoding is heavily parallelizable.

The *Transformer decoder* (Vaswani et al., 2017) reconstructs features $\boldsymbol{y}$ jointly for all slots in an autoregressive manner. In particular, the feature at position n is generated while conditioning on the set of previously generated features $\boldsymbol{y}_{<n}$ *and* the set of slots $\boldsymbol{z}$: $\boldsymbol{y}_n = \text{Decoder}(\boldsymbol{y}_{<n}; \boldsymbol{z})$. This decoder design is more powerful than the MLP decoder as it can maintain global consistency across the reconstruction, which might be needed on more complex data. However, we found several drawbacks of the Transformer decoder: it does not work with training ResNet encoders from scratch, higher resolution target features (see App. D.5), and requires more effort to tune (see App. D.4). Thus, we recommend using the MLP decoder as the first choice when applying `DINOSAUR` to a new dataset. We note that Transformer decoders have also been previously explored by SLATE (Singh et al., 2022a) and STEVE (Singh et al., 2022b), but to reconstruct the discrete token map of a VQ-VAE (van den Oord et al., 2017).

Evaluation Object-centric methods are commonly evaluated by inspecting masks associated with each slots. Previous approaches reconstructing to image-level typically use the decoder's alpha mask for this purpose; for the MLP decoder, we also make use of this option. The Transformer decoder does not produce an alpha mask. Instead, we have two options: the attention masks of Slot Attention (used by SLATE), or the decoder's attention mask over the slots. We found that the latter performed better (see Sec. D.6), and we use it throughout. As the masks from feature reconstruction are of low resolution, we bilinearly resize them to image resolution before comparing them to ground truth masks.

[1] Another option would be further finetuning the pre-trained ViT, but we found that this leads to slots that do not focus on objects. Combining ViT training with Slot Attention might require very careful training recipes.

4 EXPERIMENTS

Broadly, we pursue two goals with our experiments: 1) demonstrating that our approach significantly extends the capabilities of object-centric models towards real-world applicability (Sec. 4.1), and 2) showing that our approach is competitive with more complex methods from the computer vision literature (Sec. 4.2). Additionally, we ablate key model components to find what is driving the success of our method (Sec. 4.3). The main task we consider in this work is *object discovery*, that is, finding pixel masks for all object instances in an image.

Datasets We consider two synthetic and two real-world image datasets. As synthetic datasets, we use the MOVi datasets (Greff et al., 2022), recently introduced as challenging testbeds for object-centric methods. In particular, we use the variants MOVi-C and MOVi-E, which contain around 1 000 realistic 3D-scanned objects on HD backgrounds. For our purposes, the main difference is that MOVi-C contains 3–10, and MOVi-E 11–23 objects per scene. Note that we treat the video-based MOVi datasets as image datasets by randomly sampling frames. As real-world datasets, we use PASCAL VOC 2012 (Everingham et al., 2012) and MS COCO 2017 (Lin et al., 2014), commonly used for object detection and segmentation. Whereas PASCAL VOC contains many images with only a single large object, COCO consists of images with at least two and often dozens of objects. Both datasets represent a significant step-up in complexity to what object-centric models have been tested on so far. In App. B.2, we also report preliminary results on the KITTI driving dataset.

Training Details We train `DINOSAUR` using the Adam optimizer (Kingma & Ba, 2015) with a learning rate of $4{\cdot}10^{-4}$, linear learning rate warm-up of 10 000 optimization steps and an exponentially decaying learning rate schedule. Further, we clip the gradient norm at 1 in order to stabilize training and train for 500k steps for the MOVI and COCO datasets and 250k steps for PASCAL VOC. The models were trained on 8 NVIDIA V100 GPUs with a local batch size of 8, with 16-bit mixed precision. For the experiments on synthetic data, we use a ViT with patch size 8 and the MLP decoder. For the experiments on real-world data, we use a ViT with patch size 16 and the Transformer decoder. We analyze the impact of different decoders in Sec. 4.3. The main results are averaged over 5 random seeds; other experiments use 3 seeds. Further implementation details can be found in App. E.1.

4.1 COMPARISON TO OBJECT-CENTRIC LEARNING METHODS

Our goal in this section is two-fold: 1) demonstrating that previous object-centric methods fail to produce meaningful results on real-world datasets and struggle even on synthetic datasets, and 2) showcase how our approach of incorporating strong pre-trained models results in a large step forward for object-centric models on both kinds of datasets.

Tasks We evaluate on the task object-centric models are most frequently tested on: object discovery (Burgess et al., 2019), that is, producing a set of masks that cover the independent objects appearing on an image. We also present preliminary results testing the quality of the learned representations on the COCO dataset in App. B.3, though this is not the main focus of our work.

Metrics As common in the object-centric literature, we evaluate this task using foreground adjusted rand index (FG-ARI), a metric measuring cluster similarity. Additionally, we compute a metric based on intersection-over-union (IoU), the mean best overlap (mBO) (Pont-Tuset et al., 2017). mBO is computed by assigning each ground truth mask the predicted mask with the largest overlap, and then averaging the IoUs of the assigned mask pairs. In contrast to ARI, mBO takes background pixels into account, thus also measuring how close masks fit to objects. On datasets where objects have a semantic label attached (e.g. on COCO), we can evaluate this metric with instance-level (i.e. object) masks, and semantic-level (i.e. class) masks. This allows us to find model preferences towards instance- or semantic-level groupings.

Baselines We compare our approach to a more powerful version of Slot Attention (Locatello et al., 2020) based on a ResNet encoder that has been shown to scale to more complex data (Elsayed et al., 2022). Further, we compare with SLATE (Singh et al., 2022a), a recent object-centric model that trains a discrete VQ-VAE (van den Oord et al., 2017) as the feature extractor and a Transformer as the decoder. We refer to App. E.2 for details about baseline configurations.

As it can be hard to gauge how well object-centric methods perform on new datasets solely from metrics, we add one trivial baseline: dividing the image into a set of regular blocks. These *block masks*

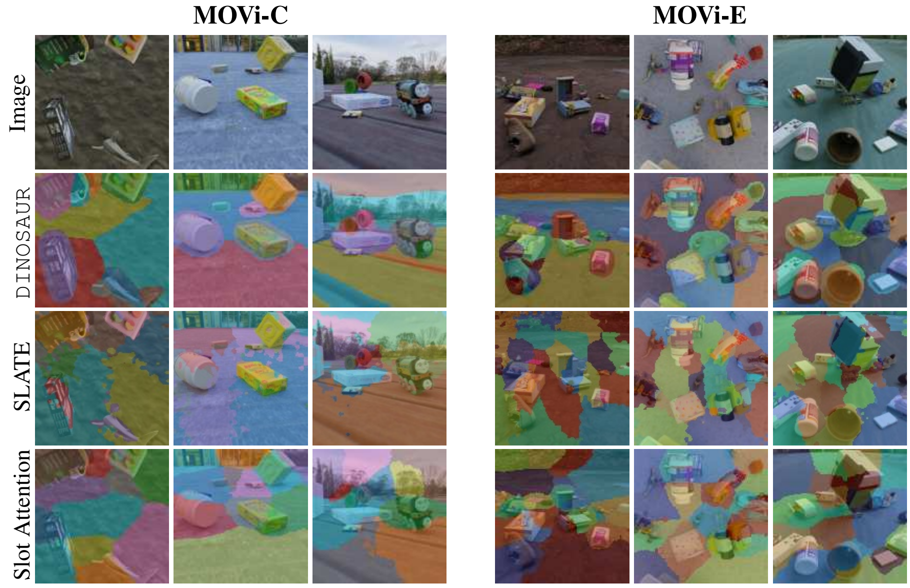


Figure 2: Example results on the synthetic MOVi-C and MOVi-E datasets (Greff et al., 2022).

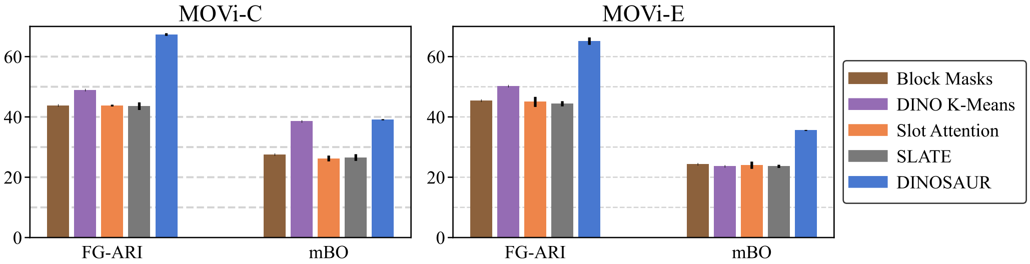


Figure 3: Object Discovery on synthetic datasets (mean ± standard dev., 5 seeds) with 11 (MOVi-C) and 24 slots (MOVi-E). We report foreground adjusted rand index (FG-ARI) and mean best overlap (mBO). DINOSAUR uses a ViT-B/8 encoder with the MLP decoder.

(see Fig. 18) thus show the performance of a method that only follows a geometric strategy to group the data, completely ignoring the semantic aspects of the image. Familiar to practitioners, this is a common failure mode of object-centric methods, particularly of Slot Attention. Last, we apply the *K-Means algorithm* on the DINO features and use the resulting clustering to generate spatial masks. This baseline shows to which extent objects are already trivially extractable from self-supervised features.

Results on Synthetic Datasets (Fig. 2 and Fig. 3) Both Slot Attention and SLATE struggle on the challenging MOVi datasets, performing similar to the naive block masks and worse than the K-Means baselines. Our model achieves good performance on both MOVI-C and MOVi-E. In App. B.1, we also find that our method compares favorably to video methods that can use temporal information and/or weak supervision (Elsayed et al., 2022; Singh et al., 2022b)

Results on Real-World Datasets (Fig. 4 and Fig. 5) As expected, Slot Attention can not handle the increased complexity of real-world data and degrades to non-semantic grouping patterns. For SLATE, semantic grouping begins to emerge (e.g. of backgrounds), but not consistently; it still performs worse than the K-Means baseline. Note that it was necessary to early-stop SLATE's training as performance would degrade to Slot Attention's level with more training. In contrast, DINOSAUR captures a variety of objects of different size, appearance and shape. To the best of our knowledge, we are the first to show a successful version of an object-centric model on unconstrained real-world images in the fully unsupervised setting. Our result represents a significant step-up in complexity of what object-centric

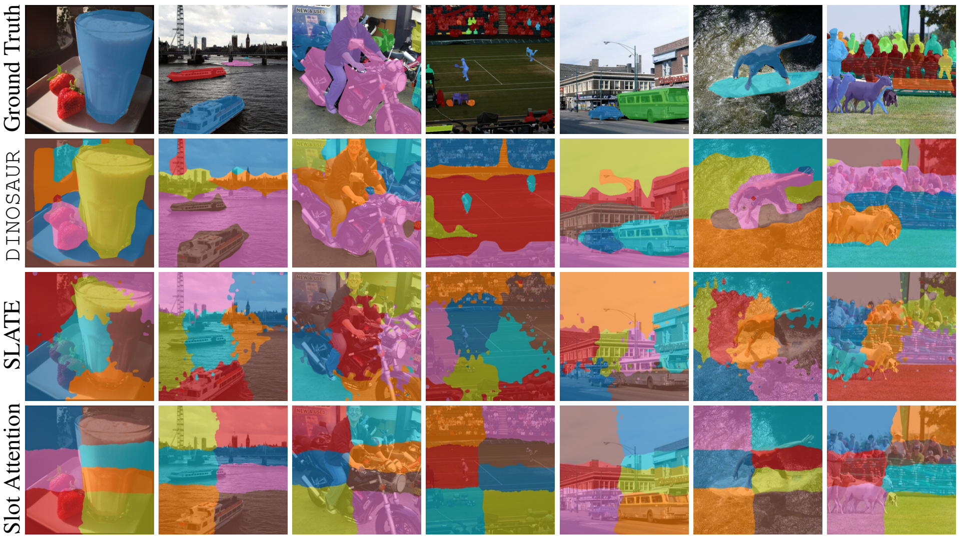


Figure 4: Example reults on COCO 2017, using 7 slots. Additional examples are provided in App. G.

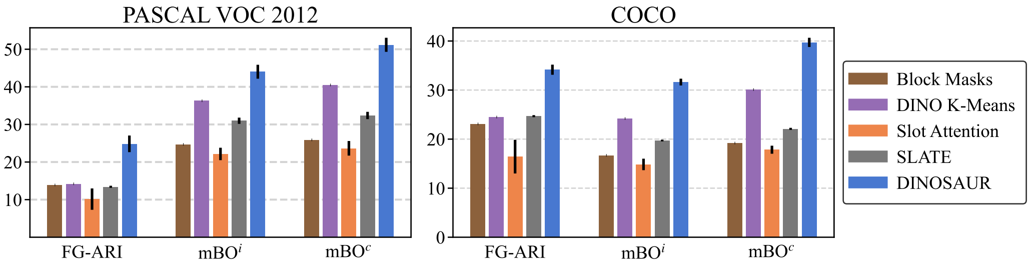


Figure 5: Object Discovery on real-world datasets (mean ± standard dev., 5 seeds) with 6 (PASCAL) and 7 slots (COCO). We report foreground adjusted rand index (FG-ARI) and instance/class mean best overlap (mBO^i/mBO^c). DINOSAUR uses a ViT-B/16 encoder with the Transformer decoder.

methods can handle. Note that the examples in Fig. 4 show mostly semantic rather than instance grouping emerging: this is a by-product of using the Transformer decoder. In contrast, the MLP decoder is biased towards instance grouping, an effect which we analyze in Sec. 4.3.

4.2 Comparison to Computer Vision Methods

In this section, our goal is to show that our method fares well on two tasks closely related to object discovery from the computer vision literature: unsupervised object localization and segmentation. Being competitive on these benchmarks is difficult, as there has been a stream of methods with quickly improving results recently (Wang et al., 2022; Hamilton et al., 2022; Zadaianchuk et al., 2023). Due to space issues, we defer most of the discussion to App. C.

Tasks, Metrics and Baselines We briefly introduce the two tasks: in *object localization*[2], the goal is to find object location and size by predicting bounding boxes, and in *unsupervised semantic segmentation* the goal is to separate the image into semantically consistent labeled regions. For the latter, we consider two variations: object segmentation, where only foreground objects should get segmented and labeled, and scene decomposition, where each pixel of the image has to be labeled with a semantic class. We evaluate object localization in terms the fraction of images on which at least one object was correctly localized (CorLoc) (Vo et al., 2020), and semantic segmentation in terms of mean intersection-over-union over classes (mIoU). For semantic segmentation, we obtain class

[2] This task is often called "object discovery" in the literature as well, but we term it "object localization" in this work in order to avoid confusion with the task evaluated in the object-centric literature.

Table 1: Representative comparisons on three tasks from the computer vision literature. We refer to App. C for a detailed discussion including more datasets, baselines, and metrics. Here, we compare with (a) DeepSpectral (Melas-Kyriazi et al., 2022) and TokenCut (Wang et al., 2022), (b) MaskContrast (Van Gansbeke et al., 2021) and COMUS (Zadaianchuk et al., 2023), and (c) SlotCon (Wen et al., 2022) and STEGO (Hamilton et al., 2022). DINOSAUR uses a ViT-B/16 encoder with the Transformer decoder (mean ± standard dev., 5 seeds).

(a) Unsup. Object Localization.

COCO-20k (CorLoc)	
DeepSpectral	52.2
TokenCut	58.8
DINOSAUR	67.2 ±1.5

(b) Unsup. Object Segmentation.

PASCAL VOC 2012 (mIoU)	
MaskContrast	35.0
COMUS	50.0
DINOSAUR	37.2 ±1.8

(c) Unsup. Scene Decomposition.

COCO-Stuff 27 (mIoU)	
SlotCon	18.3
STEGO	26.8
DINOSAUR	24.0 ±0.9

labels by running K-Means clustering on features associated with each slot after training the model, then assigning clusters to ground truth classes by maximizing IoU using Hungarian matching, similar to Van Gansbeke et al. (2021) (see App. C for details). On each task, we compare with the current state-of-the-art (Wang et al., 2022; Hamilton et al., 2022; Zadaianchuk et al., 2023), and a recent, but competitive method (Van Gansbeke et al., 2021; Melas-Kyriazi et al., 2022; Wen et al., 2022).

Results (Table 1) For object localization, our method reaches comparable results to what has been previously reported. For object segmentation, our method falls behind the state-of-the-art, though it is still competitive with other recent work. Note that the best methods on this task employ additional steps of training segmentation networks which improves results and allows them to run at the original image resolution. In contrast, the masks we evaluate are only of size 14×14; we leave it to future work to improve the resolution of the produced masks. For the task of scene decomposition, DINOSAUR comes close to the current state-of-the-art. All in all, our method is competitive with often more involved methods on these benchmarks, demonstrating a further step towards real-world usefulness of object-centric methods.

4.3 Analysis

In this section, we analyze different aspects of our approach: the importance of feature reconstruction, the impact of the method for self-supervised pre-training, and the role of the decoder. Additional experiments are included in App. D.

Insufficiency of Image Reconstruction We first test the hypothesis if a scaled-up Slot Attention model trained with image reconstruction could lead to real-world object grouping. Our experiments from Sec. 4.1 already show that a ResNet encoder is not sufficient. We additionally test a ViT-B/16 encoder under different training modes: training from scratch, frozen, or finetuning DINO pre-trained weights. We find that training from scratch results in divergence of the training process, and that both the frozen and finetuning setting fail to yield meaningful objects, resulting in striped mask patterns (see Fig. 12). Thus, even when starting from features that are highly semantic, image reconstruction does not give enough signal towards semantic grouping.

ResNet and Pre-Trained ViT Encoders Perform Similar Second, we analyze whether *pre-training the encoder* plays a crucial role for our method. To do so, we compare ResNet34 encoders trained from scratch with pre-trained ViT encoders, and find that the performance is overall similar (see Table 12). This also suggests that the feature reconstruction signal is the key component in our approach that allows object-centricness to emerge on real-world data. We expand in App. D.2.

Choice of Self-Supervised Targets (Table 2) We now analyze the role of the self-supervised pre-training algorithm. To this end, we train DINOSAUR with a ResNet34 encoder (from scratch) on COCO, but reconstruct targets obtained from ViTs pre-trained with different methods: DINO, MoCo-v3, MSN, and MAE. Remarkably, all self-supervised schemes perform well for the task of object discovery (examples in Fig. 24). This demonstrates that self-supervised pre-training on ImageNet translates into a useful, general bias for discovering objects.

Choice of Decoder (Table 3) We compare the choice of MLP vs. Transformer decoder for object discovery. Both options use a ViT-B/16 encoder. Generally, we find that the MLP decoder is better on ARI whereas the Transformer decoder is better on mBO. For MOVi-C, visual inspection (see Fig. 19)

Table 2: Comparing self-supervised reconstruction targets produced by a ViT-B/16 on COCO object discovery, with a ResNet34 encoder and the MLP decoder.

Algorithm	FG-ARI	mBO^i	mBO^c
DINO	40.9 ±0.2	27.9 ±0.0	31.1 ±0.1
MoCo-v3	40.4 ±0.6	28.1 ±0.2	31.1 ±0.1
MSN	40.7 ±0.3	27.6 ±0.1	30.7 ±0.1
MAE	37.7 ±0.1	28.1 ±0.1	31.7 ±0.0

Table 3: Comparing different decoders on object discovery, with a ViT-B/16 encoder. We also list mean squared reconstruction error (MSE).

Dataset	Decoder	ARI	$\text{mBO}^{(i,c)}$		MSE
MOVi-C	MLP	66.0	35.0		0.24
	Transformer	55.7	42.4		0.14
PASCAL	MLP	24.6	39.5	40.9	0.33
	Transformer	24.8	44.0	51.2	0.17
COCO	MLP	40.5	27.7	30.9	0.31
	Transformer	34.1	31.6	39.7	0.16

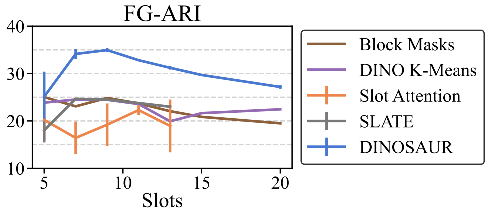


Figure 6: Sensitivity to number of slots on COCO object discovery (see also App. D.3).

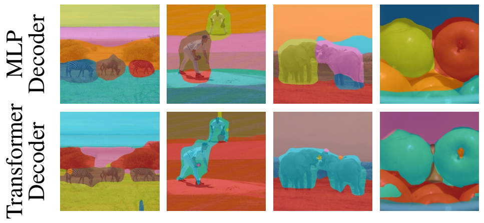


Figure 7: MLP and Transformer decoder have different biases in how they group objects.

shows that the Transformer tends to produce tighter masks and a cleaner background separation, but uses excess slots to split objects. For PASCAL and COCO, what's striking is the Transformer decoders' improvement of 9–10 class mBO. This reveals that the Transformer decoder is biased towards grouping semantically related instances into the same slot, which we suggest stems from its global view on the image, but also from the generally increased expressiveness of the architecture (cf. the lower reconstruction loss). In contrast, the MLP decoder is able to separate instances better (see Fig. 7 and also Fig. 23), which is reflected in the higher ARI scores. Researching how different decoder designs affect semantic vs. instance-level grouping is an interesting avenue for future work. In App. D.4 and App. D.5, we further study different decoder properties.

5 Conclusion

We presented the first image-based fully unsupervised approach for object-centric learning that scales to real-world data. Our experiments demonstrate significant improvements on both simulated and real-world data compared to previously suggested approaches and even achieve competitive performance with more involved pipeline methods from the computer vision literature.

This work only takes a first step towards the goal of representing the world in terms of objects. As such, some problems remain open. One issue concerns semantic vs. instance-level grouping. As evident from the presented examples, our approach covers a mix of both, with semantically related objects sometimes being grouped into a single slot. While we found the type of decoder to influence this behavior, more fine-grained control is needed. A related issue is the detail of the decomposition, e.g. whether objects are split into parts or stay whole. We found this to be dependent on the number of slots, with a fixed number often being inappropriate (see Fig. 6 and App. D.3). How models can dynamically choose a suitable level of detail while staying unsupervised but controllable will be an important challenge to fully master the ambiguities the real world inherently presents.

In this work, we mainly focused on object discovery. Future work could further examine the properties of the learned slot representations, for instance robustness to distribution shifts, generalization and usefulness for downstream tasks. Another interesting direction is how our approach can be combined with image generation to build flexible and compositional generative models of natural data.

REPRODUCIBILITY STATEMENT

Appendix E.1 contains detailed information about the DINOSAUR architecture and all hyperparamers used for all experiments. Appendix E.2 contains details about how baselines were trained. Appendix F contains information about task evaluation and datasets. All datasets used in this work (MOVi, PASCAL VOC 2012, COCO, KITTI) are public and can be obtained on their respective web pages. Source code will be made available under https://github.com/amazon-science/object-centric-learning-framework.

REFERENCES

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, Andrew Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karen Simonyan. Flamingo: a Visual Language Model for Few-Shot Learning. In *NeurIPS*, 2022. URL https://openreview.net/forum?id=EbMuimAbPbs.

Hassan Alhaija, Siva Mustikovela, Lars Mescheder, Andreas Geiger, and Carsten Rother. Augmented Reality Meets Computer Vision: Efficient Data Generation for Urban Driving Scenes. *International Journal of Computer Vision*, 2018. URL https://arxiv.org/abs/1708.01566.

Rim Assouel, Pau Rodriguez, Perouz Taslakian, David Vazquez, and Yoshua Bengio. Object-centric Compositional Imagination for Visual Abstract Reasoning. In *ICLR Workshop on the Elements of Reasoning: Objects, Structure and Causality*, 2022. URL https://openreview.net/forum?id=rCzfIruU5x5.

Mahmoud Assran, Mathilde Caron, Ishan Misra, Piotr Bojanowski, Florian Bordes, Pascal Vincent, Armand Joulin, Michael G. Rabbat, and Nicolas Ballas. Masked Siamese Networks for Label-Efficient Learning. In *ECCV*, 2022. URL https://arxiv.org/abs/2204.07141.

Zhipeng Bao, Pavel Tokmakov, Allan Jabri, Yu-Xiong Wang, Adrien Gaidon, and Martial Hebert. Discovering Objects that Can Move. *CVPR*, 2022. URL https://arxiv.org/abs/2203.10159.

Ondrej Biza, Robert Platt, Jan-Willem van de Meent, Lawson LS Wong, and Thomas Kipf. Binding Actions to Objects in World Models. In *ICLR Workshop on the Elements of Reasoning: Objects, Structure and Causality*, 2022. URL https://openreview.net/forum?id=HImz8BuUclc.

Christopher P. Burgess, Loic Matthey, Nicholas Watters, Rishabh Kabra, Irina Higgins, Matt Botvinick, and Alexander Lerchner. MONet: Unsupervised Scene Decomposition and Representation. *arXiv:1901.11390*, 2019. URL https://arxiv.org/abs/1901.11390.

Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. COCO-Stuff: Thing and Stuff Classes in Context. In *CVPR*, 2018. URL https://arxiv.org/abs/1612.03716.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging Properties in Self-Supervised Vision Transformers. *ICCV*, 2021. URL https://arxiv.org/abs/2104.14294.

Mark Chen, Alec Radford, Jeff Wu, Heewoo Jun, Prafulla Dhariwal, David Luan, and Ilya Sutskever. Generative Pretraining from Pixels. In *ICML*, 2020a. URL https://proceedings.mlr.press/v119/chen20s.html.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A Simple Framework for Contrastive Learning of Visual Representations. In *ICML*, 2020b. URL https://proceedings.mlr.press/v119/chen20j.html.

Xinlei Chen, Saining Xie, and Kaiming He. An Empirical Study of Training Self-Supervised Vision Transformers. *ICCV*, 2021. URL https://arxiv.org/abs/2104.02057.

Jang Hyun Cho, Utkarsh Mall, Kavita Bala, and Bharath Hariharan. PiCIE: Unsupervised Semantic Segmentation Using Invariance and Equivariance in Clustering. In *CVPR*, 2021. URL `https://arxiv.org/abs/2103.17070`.

Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. In *EMNLP*, 2014. URL `https://arxiv.org/abs/1406.1078`.

Minsu Cho, Suha Kwak, Cordelia Schmid, and Jean Ponce. Unsupervised Object Discovery and Localization in the Wild: Part-based Matching with Bottom-up Region Proposals. In *CVPR*, 2015. URL `https://arxiv.org/abs/1501.06170`.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A Large-scale Hierarchical Image Database. In *CVPR*, 2009. doi: 10.1109/CVPR.2009.5206848. URL `https://ieeexplore.ieee.org/document/5206848`.

Andrea Dittadi, Samuele Papa, Michele De Vita, Bernhard Schölkopf, Ole Winther, and Francesco Locatello. Generalization and Robustness Implications in Object-Centric Learning. In *ICML*, 2022. URL `https://proceedings.mlr.press/v162/dittadi22a.html`.

Alexey Dosovitskiy and Thomas Brox. Generating Images with Perceptual Similarity Metrics based on Deep Networks. In *NeurIPS*, 2016. URL `https://proceedings.neurips.cc/paper/2016/hash/371bce7dc83817b7893bcdeed13799b5-Abstract.html`.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *ICLR*, 2021. URL `https://openreview.net/forum?id=YicbFdNTTy`.

Gamaleldin Fathy Elsayed, Aravindh Mahendran, Sjoerd van Steenkiste, Klaus Greff, Michael Curtis Mozer, and Thomas Kipf. SAVi++: Towards End-to-End Object-Centric Learning from Real-World Videos. In *NeurIPS*, 2022. URL `https://openreview.net/forum?id=fT9W531LxNS`.

Martin Engelcke, Adam R. Kosiorek, Oiwi Parker Jones, and Ingmar Posner. GENESIS: Generative Scene Inference and Sampling with Object-Centric Latent Representations. In *ICLR*, 2020. URL `https://openreview.net/forum?id=BkxfaTVFwH`.

Martin Engelcke, Oiwi Parker Jones, and Ingmar Posner. GENESIS-V2: Inferring Unordered Object Representations without Iterative Refinement. In *NeurIPS*, 2021. URL `https://openreview.net/forum?id=nRBZWEUhIhW`.

S. M. Ali Eslami, Nicolas Heess, Theophane Weber, Yuval Tassa, David Szepesvari, Koray Kavukcuoglu, and Geoffrey E. Hinton. Attend, Infer, Repeat: Fast Scene Understanding with Generative Models. In *NeurIPS*, 2016. URL `https://proceedings.neurips.cc/paper/2016/hash/52947e0ade57a09e4a1386d08f17b656-Abstract.html`.

M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012), 2012. URL `http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html`.

Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets Robotics: The KITTI Dataset. *International Journal of Robotics Research*, 2013. URL `https://www.cvlibs.net/publications/Geiger2013IJRR.pdf`.

Anirudh Goyal, Alex Lamb, Jordan Hoffmann, Shagun Sodhani, Sergey Levine, Yoshua Bengio, and Bernhard Schölkopf. Recurrent Independent Mechanisms. In *ICLR*, 2021. URL `https://openreview.net/forum?id=mLcmdlEUxy-`.

Klaus Greff, Raphaël Lopez Kaufman, Rishabh Kabra, Nick Watters, Christopher Burgess, Daniel Zoran, Loic Matthey, Matthew Botvinick, and Alexander Lerchner. Multi-Object Representation Learning with Iterative Variational Inference. In *ICML*, 2019. URL `https://arxiv.org/abs/1903.00450`.

Klaus Greff, Sjoerd Van Steenkiste, and Jürgen Schmidhuber. On the Binding Problem in Artificial Neural Networks. *arXiv:2012.05208*, 2020. URL https://arxiv.org/abs/2012.05208.

Klaus Greff, Francois Belletti, Lucas Beyer, Carl Doersch, Yilun Du, Daniel Duckworth, David J. Fleet, Dan Gnanapragasam, Florian Golemo, Charles Herrmann, Thomas Kipf, Abhijit Kundu, Dmitry Lagun, Issam Laradji, Hsueh-Ti (Derek) Liu, Henning Meyer, Yishu Miao, Derek Nowrouzezahrai, Cengiz Oztireli, Etienne Pot, Noha Radwan, Daniel Rebain, Sara Sabour, Mehdi S. M. Sajjadi, Matan Sela, Vincent Sitzmann, Austin Stone, Deqing Sun, Suhani Vora, Ziyu Wang, Tianhao Wu, Kwang Moo Yi, Fangcheng Zhong, and Andrea Tagliasacchi. Kubric: A Scalable Dataset Generator. In *CVPR*, 2022. URL https://arxiv.org/abs/2203.03570.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Remi Munos, and Michal Valko. Bootstrap Your Own Latent - A New Approach to Self-Supervised Learning. In *NeurIPS*, 2020. URL https://arxiv.org/abs/2006.07733.

Mark Hamilton, Zhoutong Zhang, Bharath Hariharan, Noah Snavely, and William T. Freeman. Unsupervised Semantic Segmentation by Distilling Feature Correspondences. In *ICLR*, 2022. URL https://openreview.net/forum?id=SaKO6z6Hl0c.

Bharath Hariharan, Pablo Arbelaez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic Contours from Inverse Detectors. In *ICCV*, 2011. URL https://ieeexplore.ieee.org/document/6126343.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *CVPR*, 2015. URL https://arxiv.org/abs/1512.03385.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked Autoencoders are Scalable Vision Learners. In *CVPR*, 2022. URL https://arxiv.org/abs/2111.06377.

Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the Knowledge in a Neural Network. In *NeurIPS 2014 Deep Learning Workshop*, 2015. URL https://arxiv.org/abs/1503.02531.

Haiyang Huang, Zhi Chen, and Cynthia Rudin. SegDiscover: Visual Concept Discovery via Unsupervised Semantic Segmentation. *arXiv:2204.10926*, 2022. URL https://arxiv.org/abs/2204.10926.

Olivier J. Hénaff, Skanda Koppula, Evan Shelhamer, Daniel Zoran, Andrew Jaegle, Andrew Zisserman, João Carreira, and Relja Arandjelovi'c. Object Discovery and Representation Networks. In *ECCV*, 2022. URL https://arxiv.org/abs/2203.08777.

Andrew Jaegle, Sebastian Borgeaud, Jean-Baptiste Alayrac, Carl Doersch, Catalin Ionescu, David Ding, Skanda Koppula, Daniel Zoran, Andrew Brock, Evan Shelhamer, Olivier J. Henaff, Matthew Botvinick, Andrew Zisserman, Oriol Vinyals, and Joao Carreira. Perceiver IO: A General Architecture for Structured Inputs & Outputs. In *ICLR*, 2022. URL https://openreview.net/forum?id=fILj7WpI-g.

Xu Ji, João F. Henriques, and Andrea Vedaldi. Invariant Information Clustering for Unsupervised Image Classification and Segmentation. In *ICCV*, 2019. URL https://arxiv.org/abs/1807.06653.

Jindong Jiang, Sepehr Janghorbani, Gerard de Melo, and Sungjin Ahn. SCALOR: Generative World Models with Scalable Object Representations. In *ICLR*, 2020. URL https://openreview.net/pdf?id=SJxrKgStDH.

Daniel Kahneman, Anne Treisman, and Brian J. Gibbs. The Reviewing of Object Files: Object-specific Integration of Information. *Cognitive psychology*, 1992. URL https://www.sciencedirect.com/science/article/abs/pii/001002859290007O.

Laurynas Karazija, Iro Laina, and Christian Rupprecht. ClevrTex: A Texture-Rich Benchmark for Unsupervised Multi-Object Segmentation. In *NeurIPS Track on Datasets and Benchmarks*, 2021. URL https://arxiv.org/abs/2111.10265.

Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *ICLR*, 2015. URL https://arxiv.org/abs/1412.6980.

Thomas Kipf, Gamaleldin Fathy Elsayed, Aravindh Mahendran, Austin Stone, Sara Sabour, Georg Heigold, Rico Jonschkowski, Alexey Dosovitskiy, and Klaus Greff. Conditional Object-centric Learning from Video. In *ICLR*, 2022. URL https://openreview.net/forum?id=aD7uesX1GF_.

Adam Kosiorek, Hyunjik Kim, Yee Whye Teh, and Ingmar Posner. Sequential Attend, Infer, Repeat: Generative Modelling of Moving Objects. In *NeurIPS*, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/7417744a2bac776fabe5a09b21c707a2-Abstract.html.

Harold W. Kuhn. The Hungarian Method for the Assignment Problem. *Naval Research Logistics Quarterly*, 1955. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/nav.3800020109.

Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common Objects in Context. In *ECCV*, 2014. URL https://arxiv.org/abs/1405.0312.

Zhixuan Lin, Yi-Wu Fu, Skand Vishwanath Peri, Weihao Sun, Gautam Singh, Fei Deng, Jindong Jiang, and Sungjin Ahn. SPACE: Unsupervised Object-Oriented Scene Representation via Spatial Attention and Decomposition. In *ICLR*, 2020. URL https://openreview.net/forum?id=rkl03ySYDH.

Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-Centric Learning with Slot Attention. In *NeurIPS*, 2020. URL https://proceedings.neurips.cc/paper/2020/file/8511df98c02ab60aea1b2356c013bc0f-Paper.pdf.

Davide Mambelli, Frederik Träuble, Stefan Bauer, Bernhard Schölkopf, and Francesco Locatello. Compositional Multi-object Reinforcement Learning with Linear Relation Networks. In *ICLR Workshop on the Elements of Reasoning: Objects, Structure and Causality*, 2022. URL https://openreview.net/forum?id=HFUxPr_I5ec.

Luke Melas-Kyriazi, Christian Rupprecht, Iro Laina, and Andrea Vedaldi. Deep Spectral Methods: A Surprisingly Strong Baseline for Unsupervised Semantic Segmentation and Localization. In *CVPR*, 2022. URL https://arxiv.org/abs/2205.07839.

Roozbeh Mottaghi, Xianjie Chen, Xiaobai Liu, Nam-Gyu Cho, Seong-Whan Lee, Sanja Fidler, Raquel Urtasun, and Alan Yuille. The Role of Context for Object Detection and Semantic Segmentation in the Wild. In *CVPR*, 2014. URL https://ieeexplore.ieee.org/document/6909514.

Yassine Ouali, Céline Hudelot, and Myriam Tami. Autoregressive Unsupervised Image Segmentation. In *ECCV*, 2020. URL https://arxiv.org/abs/2007.08247.

Jordi Pont-Tuset, Pablo Arbeláez, Jonathan T. Barron, Ferran Marques, and Jitendra Malik. Multiscale Combinatorial Grouping for Image Segmentation and Object Proposal Generation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(1), 2017. doi: 10.1109/TPAMI.2016.2537320. URL https://ieeexplore.ieee.org/document/7423791.

Scott Reed, Konrad Zolna, Emilio Parisotto, Sergio Gómez Colmenarejo, Alexander Novikov, Barth maron Gabriel, Mai Giménez, Yury Sulsky, Jackie Kay, Jost Tobias Springenberg, Tom Eccles, Jake Bruce, Ali Razavi, Ashley Edwards, Nicolas Heess, Yutian Chen, Raia Hadsell, Oriol Vinyals, Mahyar Bordbar, and Nando de Freitas. A Generalist Agent. *TMLR*, 2022. URL https://openreview.net/forum?id=1ikK0kHjvj.

Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Towards Causal Representation Learning. *IEEE - Advances in Machine Learning and Deep Neural Networks*, 2021. URL https://arxiv.org/abs/2102.11107.

Oriane Simeoni, Gilles Puy, Huy V. Vo, Simon Roburin, Spyros Gidaris, Andrei Bursuc, Patrick Pérez, Renaud Marlet, and Jean Ponce. Localizing Objects with Self-Supervised Transformers and no Labels. In *BMVC*, 2021. URL https://arxiv.org/abs/2109.14279.

Gautam Singh, Fei Deng, and Sungjin Ahn. Illiterate DALL-E Learns to Compose. In *ICLR*, 2022a. URL https://openreview.net/forum?id=h0OYV0We3oh.

Gautam Singh, Yi-Fu Wu, and Sungjin Ahn. Simple Unsupervised Object-Centric Learning for Complex and Naturalistic Videos. In *NeurIPS*, 2022b. URL https://openreview.net/forum?id=eYfIM88MTUE.

Elizabeth S. Spelke and Katherine D. Kinzler. Core Knowledge. *Developmental Science*, 2007. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-7687.2007.00569.x.

Manuel Traub, Sebastian Otte, Tobias Menge, Matthias Karlbauer, Jannik Thuemmel, and Martin V. Butz. Learning What and Where: Disentangling Location and Identity Tracking Without Supervision. In *ICLR*, 2023. URL https://openreview.net/forum?id=NeDc-Ak-H_.

Aäron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural Discrete Representation Learning. In *NeurIPS*, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/7a98af17e63a0ac09ce2e96d03992fbc-Abstract.html.

Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, and Luc Van Gool. Unsupervised Semantic Segmentation by Contrasting Object Mask Proposals. *ICCV*, 2021. URL https://arxiv.org/abs/2102.06191.

Wouter Van Gansbeke, Simon Vandenhende, and Luc Van Gool. Discovering Object Masks with Transformers for Unsupervised Semantic Segmentation. *arXiv:2206.06363*, 2022. URL https://arxiv.org/abs/2206.06363.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In *NeurIPS*, 2017. URL https://papers.nips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

Huy V. Vo, Patrick Pérez, and Jean Ponce. Toward Unsupervised, Multi-object Discovery in Large-scale Image Collections. In *ECCV*, 2020. URL https://arxiv.org/abs/2007.02662.

Huy V. Vo, Elena Sizikova, Cordelia Schmid, Patrick Pérez, and Jean Ponce. Large-scale Unsupervised Object Discovery. In *NeurIPS*, 2021. URL https://arxiv.org/abs/2106.06650.

Yangtao Wang, Xi Shen, Shell Xu Hu, Yuan Yuan, James L. Crowley, and Dominique Vaufreydaz. Self-Supervised Transformers for Unsupervised Object Discovery Using Normalized Cut. In *CVPR*, 2022. URL https://arxiv.org/abs/2202.11539.

Nick Watters, Loic Matthey, Chris P. Burgess, and Alexander Lerchner. Spatial Broadcast Decoder: A Simple Architecture for Disentangled Representations in VAEs. In *ICLR Learning from Limited Labeled Data Workshop*, 2019. URL https://openreview.net/forum?id=S1x7WjnzdV.

Marissa A Weis, Kashyap Chitta, Yash Sharma, Wieland Brendel, Matthias Bethge, Andreas Geiger, and Alexander S. Ecker. Benchmarking Unsupervised Object Representations for Video Sequences. *JMLR*, 2021. URL https://jmlr.org/papers/v22/21-0199.html.

Xin Wen, Bingchen Zhao, Anlin Zheng, Xiangyu Zhang, and Xiaojuan Qi. Self-Supervised Visual Representation Learning with Semantic Grouping. In *NeurIPS*, 2022. URL https://openreview.net/forum?id=H3JObxjd8S.

Ross Wightman. PyTorch Image Models, 2019. URL https://github.com/rwightman/pytorch-image-models.

Jiahao Xie, Xiaohang Zhan, Ziwei Liu, Yew-Soon Ong, and Chen Loy Change. Unsupervised Object-Level Representation Learning from Scene Images. In *NeurIPS*, 2021. URL https://openreview.net/forum?id=X2K8KVEaAXG.

Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tie-Yan Liu. On Layer Normalization in the Transformer Architecture. In *ICML*, 2020. URL https://proceedings.mlr.press/v119/xiong20b.html.

Jiarui Xu, Shalini De Mello, Sifei Liu, Wonmin Byeon, Thomas Breuel, Jan Kautz, and Xiaolong Wang. GroupViT: Semantic Segmentation Emerges from Text Supervision. In *CVPR*, 2022. URL https://arxiv.org/abs/2202.11094.

Jianwei Yang, Jiayuan Mao, Jiajun Wu, Devi Parikh, David D Cox, Joshua B. Tenenbaum, and Chuang Gan. Object-centric Diagnosis of Visual Reasoning. *arXiv:2012.11587*, 2020. URL https://arxiv.org/abs/2012.11587.

Yafei Yang and Bo Yang. Promising or Elusive? Unsupervised Object Segmentation from Real-world Single Images. In *NeurIPS*, 2022. URL https://openreview.net/forum?id=DzPWTwfby5d.

Andrii Zadaianchuk, Maximilian Seitzer, and Georg Martius. Self-supervised Visual Reinforcement Learning with Object-centric Representations. In *ICLR*, 2020. URL https://openreview.net/forum?id=xppLmXCbOw1.

Andrii Zadaianchuk, Matthaeus Kleindessner, Yi Zhu, Francesco Locatello, and Thomas Brox. Unsupervised Semantic Segmentation with Self-supervised Object-centric Representations. In *ICLR*, 2023. URL https://openreview.net/forum?id=1_jFneF07YC.

Appendix D

Object-Centric Learning for Real-World Videos by Predicting Temporal Feature Similarities

The following 14 pages have been published as:

Andrii Zadaianchuk*, **Maximilian Seitzer***, and Georg Martius (2023). "Object-Centric Learning for Real-World Videos by Predicting Temporal Feature Similarities". In: *Conference on Neural Information Processing Systems (NeurIPS)*. *equal contribution

A summary of this publication can be found in Chap. 7. The supplementary material to this publication is available in the online version.

Object-Centric Learning for Real-World Videos by Predicting Temporal Feature Similarities

Andrii Zadaianchuk^{1,2*} **Maximilian Seitzer**^{1*} **Georg Martius**^{1}
^{1} Max Planck Institute for Intelligent Systems, Tübingen, Germany
^{2} Department of Computer Science, ETH Zurich
`andrii.zadaianchuk@tuebingen.mpg.de`

Abstract

Unsupervised video-based object-centric learning is a promising avenue to learn structured representations from large, unlabeled video collections, but previous approaches have only managed to scale to real-world datasets in restricted domains. Recently, it was shown that the reconstruction of pre-trained self-supervised features leads to object-centric representations on unconstrained real-world image datasets. Building on this approach, we propose a novel way to use such pre-trained features in the form of a temporal feature similarity loss. This loss encodes semantic and temporal correlations between image patches and is a natural way to introduce a motion bias for object discovery. We demonstrate that this loss leads to state-of-the-art performance on the challenging synthetic MOVi datasets. When used in combination with the feature reconstruction loss, our model is the first object-centric video model that scales to unconstrained video datasets such as YouTube-VIS. `https://martius-lab.github.io/videosaur/`

1 Introduction

Autonomous systems should have the ability to understand the natural world in terms of independent entities. Towards this goal, unsupervised object-centric learning methods [1–3] learn to structure scenes into object representations solely from raw perceptual data. By leveraging large-scale datasets, these methods have the potential to obtain a robust object-based understanding of the natural world. Of particular interest in recent years have been video-based methods [4–7], not least because the temporal information in video presents a useful bias for object discovery [8]. However, these approaches are so far restricted to data of limited complexity, successfully discovering objects from natural videos only on closed-world datasets in restricted domains.

In this paper, we present the method ***Video** Slot **A**ttention **U**sing temporal feature simila**R**ity*, VideoSAUR, that scales video object-centric learning to unconstrained real-world datasets covering diverse domains. To achieve this, we build upon recent advances in image-based object-centric learning. In particular, Seitzer et al. [9] showed that reconstructing pre-trained features obtained from self-supervised methods like DINO [10] or MAE [11] leads to state-of-the-art object discovery on complex real-world images. We demonstrate that combining this feature reconstruction objective with a video object-centric model [5] also leads to promising results on real-world YouTube videos.

We then identify a weakness in the training objective of current unsupervised video object-centric architectures [4, 7]: the prevalent reconstruction loss does not exploit the temporal correlations existing in video data for object grouping. To address this issue, we propose a novel self-supervised loss based on *feature similarities* that explicitly incorporates temporal information (see Fig. 1). The loss works by predicting distributions over similarities between features of the current and future

[*] equal contribution

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

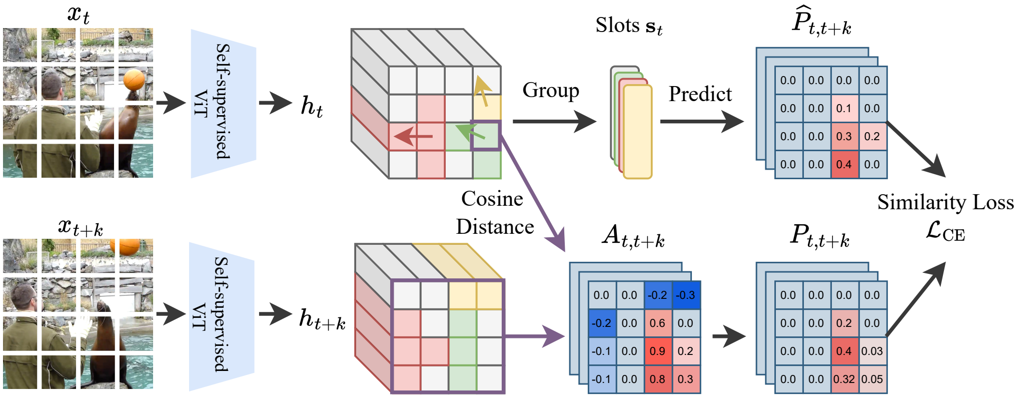


Figure 1: We propose a *self-supervised temporal similarity loss* for training object-centric video models. For each patch at time t, the model has to predict a distribution $\hat{\boldsymbol{P}}_{t,t+k}$ indicating where all semantically-similar patches have moved to k steps into the future. The target distribution $\boldsymbol{P}_{t,t+k}$ is computed with a softmax on the affinity matrix $\boldsymbol{A}_{t,t+k}$ containing the cosine distance between all patch features $\boldsymbol{h}_t$, $\boldsymbol{h}_{t+k}$. The loss incentivizes the model to group areas with consistent motion and semantics into slots.

frames. These distributions encode information about the motion of individual image patches. To efficiently predict those motions through the slot bottleneck, the model is incentivized to group patches with similar motion into the same slot, leading to better object groupings as patches belonging to an object tend to move consistently. In our experiments, we find that such a temporal similarity loss leads to state-of-the-art performance on challenging synthetic video datasets [12], and significantly boosts performance on real-world videos when used in conjunction with the feature reconstruction loss.

In video processing, model efficiency is of particular importance. Thus, we design an efficient object-centric video architecture by adapting the SlotMixer decoder [13] recently proposed for 3D object modeling for video decoding. Compared to previous decoder designs [3], the SlotMixer decoder scales gracefully with the number of slots, but has a weaker inductive bias for object grouping. We show that this weaker bias manifests in optimization difficulties in conjunction with conventional reconstruction losses, but trains robustly with our proposed temporal similarity loss. We attribute this to the *self-supervised nature* of the similarity loss: compared to reconstruction, it requires predicting information that is not directly contained in the input; the harder task seems to compensate for the weaker bias of the SlotMixer decoder.

To summarize, our contributions are as follows: (1) we propose a novel self-supervised loss for object-centric learning based on temporal feature similarities, (2) we combine this loss with an efficient video architecture based on the SlotMixer decoder where it synergistically reduces optimization difficulties, (3) we show that our model improves the state-of-the-art on the synthetic MOVi datasets by a large margin, and (4) we demonstrate that our model is able to learn video object-centric representations on the YouTube-VIS dataset [14], while staying fully unsupervised. This paper takes a large step towards unconstrained real-world object-centric learning on videos.

2 Related Work

Video Object-Centric Learning There is a rich body of work on discovering objects from video, with two broad categories of approaches: tracking bounding boxes [4, 15–17] or segmentation masks [2, 5–7, 18–25]. Architecturally, most recent image-based models for object-centric learning [3, 9, 26] are based on an auto-encoder framework with a latent slot attention grouping module [3] that extracts a set of slot representations. For processing video data, a common approach [5–7, 21, 24] is then to connect slots recurrently over input frames; the slots from the previous frame act as initialization for extracting the slots of the current frame. We also make use of this basic framework.

Scaling Object-Centric Learning Most recent work has attempted to increase the complexity of datasets where objects can successfully be discovered, such as the synthetic ClevrTex [27] and MOVi datasets [12]. On natural data, object discovery has so far been limited to restricted domains with a limited variety of objects, such as YouTube-Aquarium and -Cars [7], or autonomous driving datasets like WaymoOpen or KITTI [28]. On more open-ended datasets, previous approaches have struggled [29].

To achieve scaling, some works attempt to *improve the grouping module*, for example by introducing equivariances to slot pose transformations [30], smoothing attention maps [31], formulating grouping as graph cuts [32] or a stick-breaking process [33], or by overcoming optimization difficulties by introducing implicit differentiation [34, 35]. In contrast, we do not change the grouping module, but use the vanilla slot attention cell [3].

Another prominent approach is to introduce *better training signals* than the default choice of image reconstruction. For example, one line of work instead models the image as a distribution of discrete codes conditional on the slots, either autoregressively by a Transformer decoder [7, 26], or via diffusion [36, 37]. While this strategy shows promising results on synthetic data, it so far has failed to scale to unconstrained real-world data [9].

An alternative is to step away from fully-unsupervised representation learning by introducing *weak supervision*. For instance, SAVi [5] predicts optical flow, and SAVi++ [6] additionally predicts depth maps as a signal for object grouping. Other works add an auxiliary loss that regularizes slot attention's masks towards the masks of moving objects [8, 38]. Our model also has a loss that focuses on motion information, but uses an unsupervised formulation. OSRT [13] shows promising results on synthetic 3D datasets, but is restricted by the availability of posed multi-camera imagery. While all those approaches improve on the level of data complexity, it has not been demonstrated that they can scale to unconstrained real-world data.

The most promising avenue so far in terms of scaling to the real-world is to *reconstruct features from modern self-supervised pre-training methods* [10, 11, 39, 40]. Using this approach, DINOSAUR [9] showed that by optimizing in this highly semantic space, it is possible to discover objects on complex real-world image datasets like COCO or PASCAL VOC. In this work, we similarly use such self-supervised features, but for learning on video instead of images. Moreover, we improve upon reconstruction of features by introducing a novel loss based on similarities between features.

Concurrent Work Parallel to this work, two more slot attention-based methods were proposed that learn object-centric representations on real-world videos: SMTC [41] and SOLV [42]. SMTC learns to extracts objects from videos by enforcing semantic and instance consistency over time using a student-teacher approach. SOLV extracts per-frame slots using invariant slot attention [30], applies a temporal consistency module and merges slots using agglomerative clustering; the model is also trained using DINOSAUR-style feature reconstruction, but on masked out intermediate frames.

3 Method

In this section, we describe the main new components of VideoSAUR — our proposed object-centric video model — and its training: a pre-trained self-supervised ViT encoder extracting frame features (Sec. 3.1), a temporal similarity loss that adds a motion bias to object discovery (Sec. 3.2), and the SlotMixer decoder to achieve efficient video processing (Sec. 3.3). See Fig. 2 for an overview.

3.1 Slot Attention for Videos with Dense Self-Supervised Representations

VideoSAUR is based on the modular video object-centric architecture recently proposed by SAVi [5] and also used by STEVE [7]. Our model has three primary components: (1) a pre-trained self-supervised ViT feature encoder, (2) a recurrent grouping module for temporal slot updates, and (3) the *SlotMixer* decoder (detailed below in Sec. 3.3).

We start by processing video frames $\boldsymbol{x}_t$, with time steps $t \in \{1, \dots T\}$, into patch features $\boldsymbol{h}_t$:

$$\boldsymbol{h}_t = f_\phi(\boldsymbol{x}_t), \quad \boldsymbol{h}_t \in \mathbb{R}^{L \times D} \tag{1}$$

where f_ϕ is a self-supervised Vision Transformer encoder (ViT) [43] with pre-trained parameters ϕ, and $\boldsymbol{x}_t$ is the input at time step t. The ViT encoder processes the image by splitting it to L non-overlapping patches of fixed size (e.g. 16×16 pixels), adding positional encoding, and transforming them into L feature vectors $\boldsymbol{h}_t$ (see App. C.2 for more details on ViTs). Note that the i'th feature retains an association to the i'th image patch; the features thus can be spatially arranged. Next,

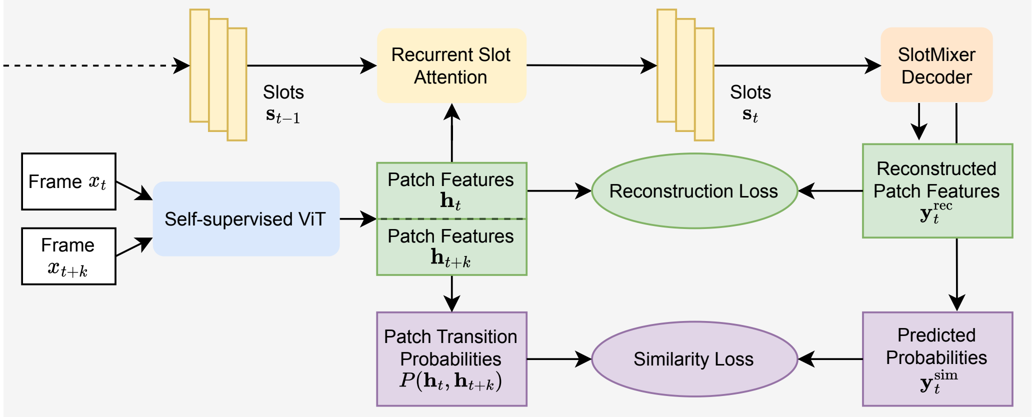

Recurrent Slot Attention
Slots $\mathbf{s}_{t-1}$
Slots $\mathbf{s}_t$
SlotMixer Decoder
Frame x_t
Frame x_{t+k}
Self-supervised ViT
Patch Features $\mathbf{h}_t$
Patch Features $\mathbf{h}_{t+k}$
Reconstruction Loss
Reconstructed Patch Features $\mathbf{y}_t^{\text{rec}}$
Patch Transition Probabilities $P(\mathbf{h}_t, \mathbf{h}_{t+k})$
Similarity Loss
Predicted Probabilities $\mathbf{y}_t^{\text{sim}}$

Figure 2: Overview of VideoSAUR. Object slots s_t are extracted from patch features $\boldsymbol{h}_t$ of a self-supervised ViT using time-recurrent slot attention, conditional on slots from the previous time step $t-1$. The model is trained by reconstructing the patch features $\boldsymbol{h}_t$ of the current frame x_t, and by predicting the similarity distribution over patches of a future frame x_{t+k} (see also Fig. 1). The predictions $\boldsymbol{y}_t^{\text{rec}}$ and $\boldsymbol{y}_t^{\text{sim}}$ are decoded efficiently using SlotMixer decoder.

we transform the features from the encoder with a slot attention module [3] to obtain a latent set $\boldsymbol{s}_t = \{\boldsymbol{s}_t^i\}_{i=1}^K$, $s_t^i \in \mathbb{R}^M$ with K slot representations:

$$\boldsymbol{s}_t = \mathrm{SA}_\theta(\boldsymbol{h}_t, \boldsymbol{s}_{t-1}). \tag{2}$$

Slot attention is recurrently initialized with the slots of the previous time step $t-1$, with initial slots s_0 sampled independently from a Gaussian distribution with learned location and scale. Slot attention works by grouping input features into slots by iterating competitive attention steps; we refer to Locatello et al. [3] for more details. To train the model, we use a SlotMixer decoder g_ψ (see Sec. 3.3) to transform the slots $\boldsymbol{s}_t$ to outputs $\boldsymbol{y}_t = g_\psi(\boldsymbol{s}_t)$. Those outputs are used as model predictions for the reconstruction and similarity losses introduced next.

3.2 Self-Supervised Object Discovery by Predicting Temporal Similarities

We now motivate our novel loss function based on predicting temporal feature similarities. Video affords the opportunity to discover objects from motion: pixels that consistently move together should be considered as one object, sometimes called the "common fate" principle [44]. However, the widely used reconstruction objective — whether of pixels [5], discrete codes [7] or features [9] — does not exploit this bias, as to reconstruct the input frame, the changes between frames do not have to be taken into account.

Taking inspiration from prior work using optical flow as a prediction target [5], we design a self-supervised objective that requires *predicting patch motion*: for each patch, the model needs to predict where all *semantically-similar* patches have moved to k steps into the future. By comparing self-supervised features describing the patches, we integrate both semantic and motion information; this is in contrast to optical flow prediction, which only relies on motion. Specifically, we construct an affinity matrix $\boldsymbol{A}_{t,t+k}$ with the cosine similarities between all patch features from the present frame $\boldsymbol{h}_t$ and all features from some future frame $\boldsymbol{h}_{t+k}$:

$$\boldsymbol{A}_{t,t+k} = \frac{\boldsymbol{h}_t}{\|\boldsymbol{h}_t\|} \cdot \left(\frac{\boldsymbol{h}_{t+k}}{\|\boldsymbol{h}_{t+k}\|}\right)^\top, \quad \boldsymbol{A}_{t,t+k} \in [-1,1]^{L\times L}. \tag{3}$$

As self-supervised features are highly semantic, the obtained feature similarities are high for patches that share the same semantic interpretation. Due to the ViT's positional encoding, the similarities also take spatial closeness of patches into account. Figure 3 shows several example affinity matrices.

Because there are ambiguities in our similarity-based derivation of feature movements, we frame the prediction task as *modeling a probability distribution* over target patches — instead of forcing the prediction of an exact target location, like with optical flow prediction. Thus, we define the

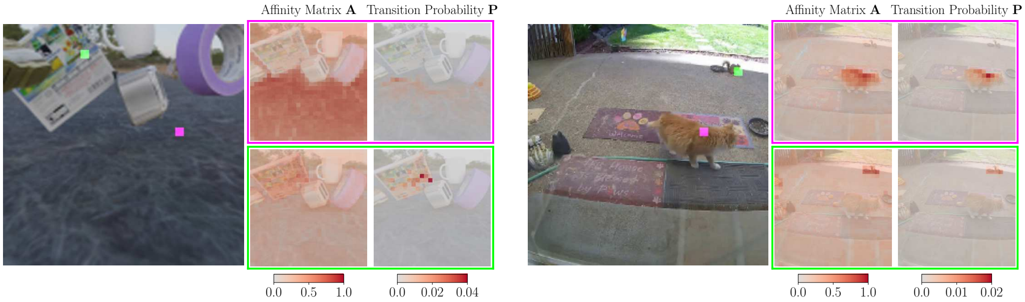


Figure 3: Affinity matrix $\boldsymbol{A}_{t,t+k}$ and transition probabilities $\boldsymbol{P}_{t,t+k}$ values between patches (marked by purple and green) of the frame $\boldsymbol{x}_t$ and patches of the future frame $\boldsymbol{x}_{t+k}$ in MOVi-C (left) and YT-VIS (right). Red indicates maximum affinity/probability. Also see Fig. B.4 for more examples, and our website for an interactive visualization of temporal feature similarities.

probability that patch i moves to patch j by normalizing the rows of the affinity matrix with the softmax, while masking negative similarity values (superscripts refer to the elements of the matrix):

$$\boldsymbol{P}^{ij} = \begin{cases} \dfrac{\exp(\boldsymbol{A}^{ij}/\tau)}{\sum\limits_{k \in \{j \mid \boldsymbol{A}^{ij} \geq 0\}} \exp(\boldsymbol{A}^{ik}/\tau)} & \text{if } \boldsymbol{A}^{ij} \geq 0, \\ 0 & \text{if } \boldsymbol{A}^{ij} < 0, \end{cases} \tag{4}$$

where τ is the softmax temperature. The resulting distribution can be interpreted as the *transition probabilities* of a random walk along a graph with image patches as nodes [45]. Then, we define the similarity loss as the cross entropy between decoder outputs and transition probabilities:

$$\mathcal{L}^{\text{sim}}_{\theta,\psi} = \sum_{l=1}^{L} \text{CE}(\boldsymbol{P}^{l}_{t,t+k}; \boldsymbol{y}^{l}_{t}). \tag{5}$$

Figure 1 illustrates the loss computation for an example pair of input frames.

Why is this Loss Useful for Object Discovery? Predicting which parts of the videos move consistently is most efficient with an object decomposition that captures moving objects. This is similar to previous losses predicting optical flow [5]. But in contrast, our loss (Eq. 5) also yields a useful signal for grouping when parts of the frame are *not* moving: as feature similarities capture semantic aspects, the task also requires predicting which patches are semantically similar, helping the grouping into objects e.g. by distinguishing fore- and background (see Fig. 3). Optical flow for grouping also has limits when camera motion is introduced; in our experiments, we find that our loss is more robust in such situations. Methods based on optical flow or motion masks can also struggle with inaccurate flow/motion mask labels — unlike our method, which does not require such labels. This is of particular importance for in-the-wild video, where motion estimation is challenging.

Role of Hyperparameters. The loss has two hyperparameters: the time shift into the future k and the softmax temperature τ. The optimal time shift depends on the expected time scales of movements in the modeled videos and should be chosen accordingly. The temperature τ controls the concentration of the distribution onto the maximum. Thus, it effectively modulates between two different tasks: accurately estimating the patch motion (low τ), and predicting the similarity of each patch to all other patches (high τ). In particular in scenes with little movement, the latter may be important to maintain a meaningful prediction task. In our experiments, we find that the best performance is obtained with a balance between the two, showing that both modes are important.

Final Loss. While the temporal similarity loss yields state-of-the-art performance on synthetic datasets, as shown below, we found that on real-world data, performance can be further improved by adding the feature reconstruction objective as introduced in Seitzer et al. [9]. We hypothesize this is because the semantic nature of feature reconstruction adds another useful bias for object discovery. Thus, the final loss is given by:

$$\mathcal{L}_{\theta,\psi} = \sum_{t=1}^{T-k} \mathcal{L}^{\text{sim}}_{\theta,\psi}(\boldsymbol{P}_{t,t+k}, \boldsymbol{y}^{\text{sim}}_{t}) + \alpha \mathcal{L}^{\text{rec}}_{\theta,\psi}(\boldsymbol{h}_t, \boldsymbol{y}^{\text{rec}}_{t}), \tag{6}$$

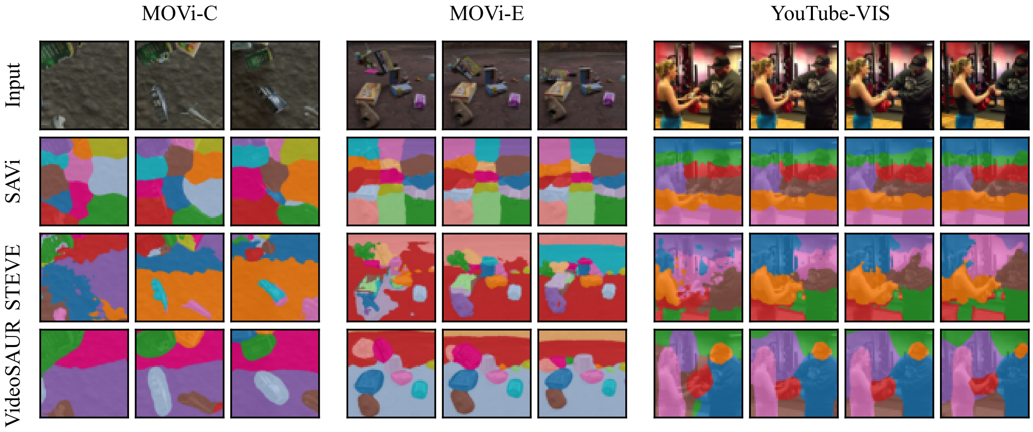


Figure 4: Example predictions of VideoSAUR compared to recent video object-centric methods.

where $\boldsymbol{y}_t = [\boldsymbol{y}_t^{\text{sim}} \in \mathbb{R}^{L\times L}, \boldsymbol{y}_t^{\text{rec}} \in \mathbb{R}^{L\times D}]$ is the output of the SlotMixer decoder g_ψ and α is a weighting factor used to make the scales of the two losses similar (we use a fixed value of $\alpha = 0.1$ for all experiments on real-world datasets). Like in Seitzer et al. [9], we do not train the ViT encoder f_ϕ.

3.3 Efficient Video Object-Centric Learning with the SlotMixer Decoder

In video models, resource efficiency is of particular concern: recurrent frame processing increases the load on compute and memory. The standard mixture-based decoder design [3] decodes each output K-times, where K is the number of slots, and thus scales linearly with K both in compute and memory. This can become prohibitive even for a moderate number of slots. The recently introduced SlotMixer decoder [13] for 3D object-centric learning instead has, for all practical purposes, constant overhead in the number of slots, by only decoding once per output. Thus, we propose to use a SlotMixer decoder g_ψ for predicting the probabilities $\boldsymbol{P}_{t,t+k}$ from the slots $\boldsymbol{s}_t$. To adapt the decoder from 3D to 2D outputs, we change the conditioning on 3D query rays to L learned positional embeddings, corresponding to L patch outputs $\boldsymbol{y}_t^l$. See App. C.1 for more details on the SlotMixer module.

As a consequence of the increased efficiency of SlotMixer, there also is increased flexibility of how slots can be combined to form the outputs. Because of this, this decoder has a weaker inductive bias towards object-based groupings compared to the standard mixture-based decoder. With the standard reconstruction loss, we observed that this manifests in training runs in which no object groupings are discovered. But in combination with our temporal similarity loss, these instabilities disappear (see App. B.4). We attribute this to the *self-supervised nature* of the similarity loss[2]; having to predict information that is not directly contained in the input increases the difficulty of the task, reducing the viability of non-object based groupings.

4 Experiments

We have conducted a number of experiments to answer the following questions: (1) Can object-centric representations be learned from a large number of diverse real-world videos? (2) How does VideoSAUR perform in comparison to other methods on well-established realistic synthetic datasets? (3) What are the effects of our proposed temporal feature similarity loss and its parameters? (4) Can we transfer the learned object-grouping to unseen datasets? (5) How efficient is the SlotMixer decoder in contrast to the mixture-based decoder?

4.1 Experimental Setup

Datasets To investigate the characteristics of our proposed method, we utilize three synthetic datasets and three real-world datasets. For synthetic datasets, we selected the MOVi-C, MOVi-D

[2] Novel-view synthesis, the original task for which SlotMixer was proposed, is similarly a self-supervised prediction task. This may have contributed to the success of SlotMixer in that setting.

Table 1: Comparison with state-of-the-art methods on the MOVi-C, MOVi-E, and YT-VIS datasets. We report foreground adjusted rand index (FG-ARI) and mean best overlap (mBO) over 5 random seeds. Both metrics are computed for the whole video (24 frames for MOVi, 6 frames for YT-VIS).

	MOVi-C		MOVi-E		YT-VIS	
	FG-ARI	mBO	FG-ARI	mBO	FG-ARI	mBO
Block Pattern	24.2	11.1	36.0	16.5	24	14.9
SAVi [5]	22.2 ± 2.1	13.6 ± 1.6	42.8 ± 0.9	16.0 ± 0.3	11.1 ± 5.6	12.7 ± 2.3
STEVE [7]	36.1 ± 2.3	26.5 ± 1.1	50.6 ± 1.7	26.6 ± 0.9	20.0 ± 1.5	20.9 ± 0.5
VideoSAUR	**64.8 ± 1.2**	**38.9 ± 0.6**	**73.9 ± 1.1**	**35.6 ± 0.5**	**39.5 ± 0.6**	**29.1 ± 0.4**

and MOVi-E datasets [12] that consist of numerous moving objects on complex backgrounds. Additionally, we evaluate the performance of our method on the challenging YouTube Video Instance Segmentation (YT-VIS) 2021 dataset [14] as an unconstrained real-world dataset. Furthermore, we examine how well our model performs when transferred from YT-VIS 2021 to YT-VIS 2019 [46] and DAVIS [47]. Finally, we use the COCO dataset [48] to study our proposed similarity loss function with image-based object-centric learning.

Metrics We evaluate our approach in terms of the quality of the discovered slot masks (output by the decoder), using two metrics: video foreground ARI (FG-ARI) [2] and video mean best overlap (mBO) [49]. FG-ARI is a video version of a widely used metric in the object-centric literature that measures the similarity of the discovered objects masks to ground truth masks. This metric mainly measures *how well objects are split*. mBO assesses the correspondence of the predicted and the ground truth masks using the intersection-over-union (IoU) measure. In particular, each ground truth mask is matched to the predicted mask with the highest IoU, and the average IoU is then computed across all assigned pairs. Unlike FG-ARI, mBO also considers background pixels, and provides a measure of *how accurately the masks fit the objects*. Both metrics also consider the consistency of the assigned object masks over the whole video.

In addition, we also use image-based versions of those metrics (*Image FG-ARI* and *Image mBO*, computed on individual frames) for comparing with image-based methods.

Baselines We compare our method with two recently proposed methods for unsupervised object-centric learning for videos: SAVi [5] and STEVE [7]. SAVi uses a mixture-based decoder and is trained with image reconstruction. We use the unconditional version of SAVi. STEVE uses a transformer decoder and is trained by reconstructing discrete codes of a dVAE [50]. Similar to Seitzer et al. [9], we also add a regular block pattern baseline, corresponding to splitting the video into regular block masks of similar size that do not change over time. By showing the metric values for a trivial decomposition of the video, this baseline is useful to contextualize the results of the other methods. In addition to video-based methods, we compare our model to image-based methods, including DINOSAUR [9], LSD [36] and Slot Diffusion [37], showing that our approach performs well in both object separation and mask sharpness. Last, we also compare our model to two concurrent works discovering objects from real-world video, SMTC [41] and SOLV [42].

4.2 Comparison with State-of-the-Art Object-Centric Learning Methods

When comparing VideoSAUR to STEVE and SAVi, it is evident that VideoSAUR outperforms the baselines by a significant margin, both in terms of FG-ARI and mBO (see Table 1 and Fig. 4). On the most challenging synthetic dataset (MOVi-E), VideoSAUR reaches 73.9 FG-ARI. Notably, for the challenging YT-VIS 2021 dataset, both baselines perform comparable or worse than the block pattern baseline in terms of FG-ARI, showing that previous methods struggle to decompose real-world videos into consistent objects. We additionally compare VideoSAUR to image-based methods in App. A.1, including strong recent methods (LSD, SlotDiffusion and DINOSAUR), and find that our approach also outperforms the prior image-based SoTA. Finally, in App. A.2, we find that our method performs competitively with concurrent work.

Next, we report how well our method performs in terms of zero-shot transfer to other datasets to show that the learned object discovery does generalize to unseen data. In particular, we train VideoSAUR

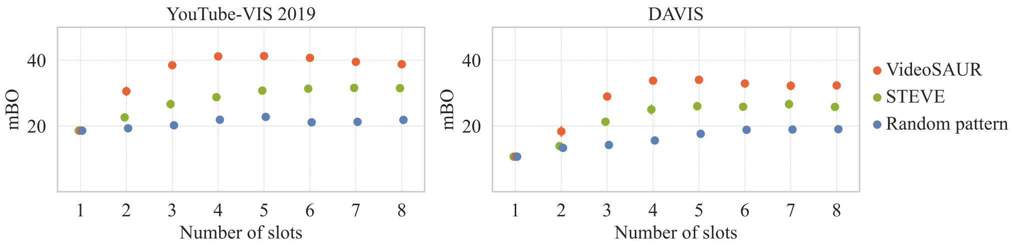


Figure 5: Zero-shot transfer of learned object-centric representations on YT-VIS 2021 to the YT-VIS 2019 and DAVIS datasets for different number of slots.

on the YT-VIS 2021 dataset and evaluate it on the YT-VIS 2019 and DAVIS datasets. YT-VIS 2019 has similar object categories, but a smaller number of objects per image. The DAVIS dataset consists of videos from a fully different distribution than YT-VIS 2021. As the number of slots can be changed during evaluation, we test VideoSAUR with different number of slots, revealing that the optimal number of slots is indeed smaller for these datasets. We find that our method achieves a performance of 41.3 ± 0.9 mBO on YT-VIS 2019 dataset and 34.0 ± 0.4 mBO on DAVIS dataset (see Fig. 5), illustrating its capability to effectively transfer the learned representations to previously unseen data with different object categories and numbers of objects.

Long-term Video Consistency In addition to studying how VideoSAUR performs on relatively short 6-frame video segments from YT-VIS, we also evaluate our method on longer videos. In App. B.1, we show the performance for 12-frame and full YT-VIS videos. While, as can be expected, performance on longer video segments is smaller in terms of FG-ARI, we show that the gap between VideoSAUR and the baselines is large, indicating that VideoSAUR can track the main objects in videos over longer time intervals. Closing the gap between short-term and long-term consistency using memory modules [24, 51] is an interesting future direction that could be useful for video prediction [52] as well as for object-centric goal-based [53, 54] and model-based [55] reinforcement learning.

4.3 Analysis

In this section, we analyze various aspects of our approach, including the importance of the similarity loss, the impact of hyperparameters (time-shift k and softmax temperature τ), and the effect of the choice of self-supervised features and decoder.

Choice of Loss Function (Table 2 and Table 3) We conduct an ablation study to demonstrate the importance of the proposed temporal similarity loss, comparing and combining it with the feature reconstruction loss [9]. We also consider predicting the features of the *next frame* (see App. C.4 for implementation details). For all datasets, feature reconstruction alone performs significantly worse than the combination of feature reconstruction and temporal similarity loss. Predicting the features of the next frame in addition to feature reconstruction also yields improved performance, but is worse than the temporal similarity, suggesting that the success of our loss can be partially explained by the integration of temporal information through future prediction. Interestingly, on MOVi-C, using the temporal similarity loss alone significantly improves the performance over feature reconstruction (+20 FG-ARI, +7 mBO). To provide insight into the qualitative differences between the losses, we analyze the videos with the most significant differences in FG-ARI (see Fig. E.4): unlike feature reconstruction, the temporal similarity loss does not fragment the background or large objects into numerous slots, and it exhibits improved object-tracking capabilities even when object size changes. To gain further insights, we also consider (ground truth) *optical flow* as a prediction target that only captures motion, but no semantic information (see App. B.2 for a detailed discussion). We find that only predicting optical flow is not enough for a successful scene decomposition, underscoring the importance of integrating both motion and semantic information for real-world object discovery.

Robustness to Camera Motion (Table 4) Next, we investigate if VideoSAUR training with the similarity loss is robust to camera motion, as such motion makes isolating the object motion more difficult. As a controlled experiment, we compare between MOVi-D (without camera motion) and

Table 2: Loss ablation on MOVi-C.

Loss Type			Metric	
Feat. Rec.	Next Frame Feat. Pred.	Temp. Sim.	FG-ARI	mBO
✓			40.2	23.5
✓	✓		47.2	24.7
		✓	**60.8**	**30.5**
✓		✓	60.7	30.3

Table 3: Loss ablation on YT-VIS.

Loss Type			Metric	
Feat. Rec.	Next Frame Feat. Pred.	Temp. Sim.	FG-ARI	mBO
✓			35.4	26.7
✓	✓		37.9	27.3
		✓	26.2	**29.1**
✓		✓	**39.5**	**29.1**

Table 4: Robustness to introducing camera motion (MOVi-D → MOVi-E).

	MOVi-D	MOVi-E
SAVi (optical flow) [12]	19.4	2.7
VideoSAUR (temporal sim.)	55.7	62.5

Table 5: Decoder comparison on MOVi-C and YT-VIS.

	MOVi-C		YT-VIS		Memory
	FG-ARI	mBO	FG-ARI	mBO	GB @24 slots
Mixer	60.8	30.5	39.5	29.1	24
MLP	64.2	27.2	39.0	29.1	70

MOVi-E (with camera motion), and train VideoSAUR using only the temporal similarity loss. We contrast with SAVi trained with optical flow prediction[3], and find that VideoSAUR is more robust to camera motion, performing better on the MOVi-E dataset than on the MOVi-D dataset ($+6.8$ vs -16.7 FG-ARI for SAVi).

Choice of Decoder (Table 5) We analyze how our method performs with different decoders and find that both the MLP broadcast decoder [9] and our proposed SlotMixer decoder can be used for optimizing the temporal similarity loss. VideoSAUR with the MLP broadcast decoder achieves similar performance on YT-VIS and MOVi datasets, but requires 2–3 times more GPU memory (see App. C.1 for more details and Table B.3 for the detailed comparison of decoders on MOVI-E dataset). Thus, we suggest to use the SlotMixer decoder for efficient video processing.

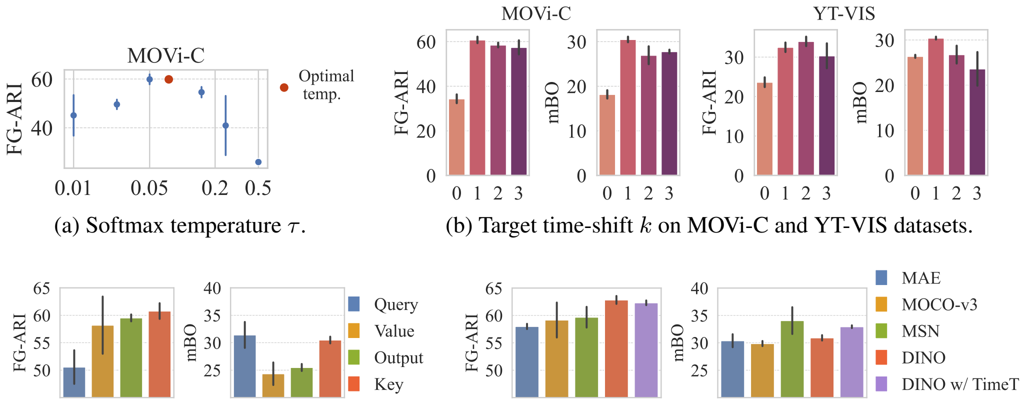
Softmax Temperature (Figure 6a) We train VideoSAUR with DINO S/16 features using different softmax temperatures τ. We find that there is a sweet spot in terms of grouping performance at $\tau = 0.075$. Lower and higher temperatures lead to high variance across seeds, potentially because there is not enough training signal with very peaked (low τ) and diffuse (high τ) target distributions.

Target Time-shift (Figure 6b) We train VideoSAUR with DINO S/16 features using different time-shifts k to construct the affinity matrix $\boldsymbol{A}_{t,t+k}$. On both synthetic and real-world datasets, $k = 1$ generally performs best. Interestingly, we find that for $k = 0$, performance drops, indicating that predicting pure self-similarities is not a sufficient task for discovering objects on its own.

Choices for Self-Supervised Features (Figures 6c and 6d) We study two questions about the usage of the ViT features: which ViT features (queries/keys/values/outputs) should be used for the temporal similarity loss? Do different self-supervised representations result in different performance? In Fig. 6c, we observe that using DINO "key" and "query" features leads to significantly larger mBO, while for FG-ARI "query" is worse and the other features are similar. Potentially, this is because keys are used in the ViT's self-attention and thus could be particularly good to compare with the scalar product similarity. Consequently, VideoSAUR uses "key" features in all other experiments. Moreover, we study if the temporal similarity loss is compatible with different self-supervised representations. In Fig. 6d, we show that VideoSAUR works well with 4 different types of representations, with MSN [39] and DINO [10] performing slightly better than MAE [11] and MOCO-v3 [40]. We also demonstrate that *further fine-tuning the DINO features* utilizing a self-supervised temporal-alignment clustering approach named TimeTuning [56] on unlabeled videos enhances the mask quality of VideoSAUR.

Pre-training Dataset (Table 6) All self-supervised methods we utilize are trained on the ImageNet dataset, which a) has a strong bias towards object-centricness as its images mostly contain single objects, and b) introduces a large number of additional images external to the dataset we are training

[3] SAVi results with optical flow are from Greff et al. [12].



(a) Softmax temperature τ.

(b) Target time-shift k on MOVi-C and YT-VIS datasets.

(c) DINO ViT features.

(d) Self-supervised representation method.

Figure 6: Studying the effect of different parameters of the temporal similarity loss.

Table 6: Comparing VideoSAUR with features trained on MOVi-E (*MAE+MOVi-E*) to features trained on ImageNet (*MAE+ImageNet*). For MAE+MOVi-E, we pre-train a ViT-B/16 using the self-supervised MAE method on MOVi-E for 200 epochs. VideoSAUR is able to perform high-quality object discovery even without access to any external data.

	MOVi-C		MOVi-E	
	FG-ARI	mBO	FG-ARI	mBO
VideoSAUR w/ *MAE+ImageNet* features	58.0	30.4	72.8	27.1
VideoSAUR w/ *MAE+MOVi-E* features	59.8	27.5	70.6	23.3

VideoSAUR on. An interesting question is whether a) and b) are actually required for the success of our method. To answer it, we train a ViT-B/16 encoder from scratch on the MOVi-E dataset using the MAE method, and then train VideoSAUR using the obtained features. Interestingly, we find that the features from MOVi-E yield similar results compared to ImageNet-trained features (although with slight drops in mask quality), demonstrating that VideoSAUR is able to perform high-quality object discovery even without access to external data. This result also has broader implications as it potentially increases the applicability of feature reconstruction-based object-centric methods to datasets fully out of the domain of ImageNet. It also raises a follow-up question: what properties of the pre-training dataset (and method) are important to obtain good target features for object discovery?

5 Conclusion

This paper presents the first method for unsupervised video-based object-centric learning that scales to diverse, unconstrained real-world datasets such as YouTube-VIS. By leveraging dense self-supervised features and extracting motion information with temporal similarity loss, we demonstrate superior performance on both synthetic and real-world video datasets. We hope our new loss function can inspire the design of further self-supervised losses for object-centric learning, especially in the video domain where natural self-supervision is available.

Still, our method does not come without limitations: in longer videos with occlusions, slots can get reassigned to different objects or the background (see Fig. B.5 for visualizations of failure cases). VideoSAUR also inherits a limitation of all slot attention-based method, namely that the the number of slots is static and needs to be chosen a priori. Similar to DINOSAUR [9], the quality of the object masks is restricted by the patch-based nature of the decoder. Finally, while the datasets we use in this work are significantly less constrained compared to datasets used by prior work, they still do not capture the full open-world setting that object-centric learning aspires to solve. Overcoming these limitations is a great direction for future work.

Acknowledgements

We thank Martin Butz, Cansu Sancaktar and Manuel Traub for useful suggestions and discussions. The authors thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting Maximilian Seitzer. Andrii Zadaianchuk is supported by the Max Planck ETH Center for Learning Systems. Georg Martius is a member of the Machine Learning Cluster of Excellence, funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy – EXC number 2064/1 – Project number 390727645. We acknowledge the support from the German Federal Ministry of Education and Research (BMBF) through the Tübingen AI Center (FKZ: 01IS18039B).

References

[1] Christopher P. Burgess, Loic Matthey, Nicholas Watters, Rishabh Kabra, Irina Higgins, Matt Botvinick, and Alexander Lerchner. MONet: Unsupervised Scene Decomposition and Representation. *arXiv:1901.11390*, 2019. URL `https://arxiv.org/abs/1901.11390`.

[2] Klaus Greff, Raphaël Lopez Kaufman, Rishabh Kabra, Nick Watters, Christopher Burgess, Daniel Zoran, Loic Matthey, Matthew Botvinick, and Alexander Lerchner. Multi-Object Representation Learning with Iterative Variational Inference. In *ICML*, 2019. URL `https://arxiv.org/abs/1903.00450`.

[3] Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-Centric Learning with Slot Attention. In *NeurIPS*, 2020. URL `https://proceedings.neurips.cc/paper/2020/file/8511df98c02ab60aea1b2356c013bc0f-Paper.pdf`.

[4] Jindong Jiang, Sepehr Janghorbani, Gerard de Melo, and Sungjin Ahn. SCALOR: Generative World Models with Scalable Object Representations. In *ICLR*, 2020. URL `https://openreview.net/pdf?id=SJxrKgStDH`.

[5] Thomas Kipf, Gamaleldin Fathy Elsayed, Aravindh Mahendran, Austin Stone, Sara Sabour, Georg Heigold, Rico Jonschkowski, Alexey Dosovitskiy, and Klaus Greff. Conditional Object-centric Learning from Video. In *ICLR*, 2022. URL `https://openreview.net/forum?id=aD7uesX1GF_`.

[6] Gamaleldin Fathy Elsayed, Aravindh Mahendran, Sjoerd van Steenkiste, Klaus Greff, Michael Curtis Mozer, and Thomas Kipf. SAVi++: Towards End-to-End Object-Centric Learning from Real-World Videos. In *NeurIPS*, 2022. URL `https://openreview.net/forum?id=fT9W531LxNS`.

[7] Gautam Singh, Yi-Fu Wu, and Sungjin Ahn. Simple Unsupervised Object-Centric Learning for Complex and Naturalistic Videos. In *NeurIPS*, 2022. URL `https://openreview.net/forum?id=eYfIM88MTUE`.

[8] Zhipeng Bao, Pavel Tokmakov, Allan Jabri, Yu-Xiong Wang, Adrien Gaidon, and Martial Hebert. Discovering Objects that Can Move. *CVPR*, 2022. URL `https://arxiv.org/abs/2203.10159`.

[9] Maximilian Seitzer, Max Horn, Andrii Zadaianchuk, Dominik Zietlow, Tianjun Xiao, Carl-Johann Simon-Gabriel, Tong He, Zheng Zhang, Bernhard Schölkopf, Thomas Brox, and Francesco Locatello. Bridging the gap to real-world object-centric learning. In *ICLR*, 2023. URL `https://openreview.net/forum?id=b9tUk-f_aG`.

[10] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging Properties in Self-Supervised Vision Transformers. *ICCV*, 2021. URL `https://arxiv.org/abs/2104.14294`.

[11] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked Autoencoders are Scalable Vision Learners. In *CVPR*, 2022. URL `https://arxiv.org/abs/2111.06377`.

[12] Klaus Greff, Francois Belletti, Lucas Beyer, Carl Doersch, Yilun Du, Daniel Duckworth, David J. Fleet, Dan Gnanapragasam, Florian Golemo, Charles Herrmann, Thomas Kipf, Abhijit Kundu, Dmitry Lagun, Issam Laradji, Hsueh-Ti (Derek) Liu, Henning Meyer, Yishu Miao, Derek Nowrouzezahrai, Cengiz Oztireli, Etienne Pot, Noha Radwan, Daniel Rebain, Sara Sabour, Mehdi S. M. Sajjadi, Matan Sela, Vincent Sitzmann, Austin Stone, Deqing Sun, Suhani Vora, Ziyu Wang, Tianhao Wu, Kwang Moo Yi, Fangcheng Zhong, and Andrea Tagliasacchi. Kubric: A Scalable Dataset Generator. In *CVPR*, 2022. URL `https://arxiv.org/abs/2203.03570`.

[13] Mehdi SM Sajjadi, Daniel Duckworth, Aravindh Mahendran, Sjoerd van Steenkiste, Filip Pavetic, Mario Lucic, Leonidas J Guibas, Klaus Greff, and Thomas Kipf. Object scene representation transformer. In *NeurIPS*, 2022. URL `https://arxiv.org/abs/2206.06922`.

[14] Linjie Yang, Yuchen Fan, Yang Fu, and Ning Xu. The 3rd large-scale video object segmentation challenge - video instance segmentation track, June 2021. URL `https://youtube-vos.org/dataset/vis`.

[15] Adam Kosiorek, Hyunjik Kim, Yee Whye Teh, and Ingmar Posner. Sequential Attend, Infer, Repeat: Generative Modelling of Moving Objects. In *NeurIPS*, 2018. URL `https://proceedings.neurips.cc/paper/2018/hash/7417744a2bac776fabe5a09b21c707a2-Abstract.html`.

[16] Eric Crawford and Joelle Pineau. Spatially invariant unsupervised object detection with convolutional neural networks. In *AAAI*, 2020. URL `https://arxiv.org/abs/1911.09033`.

[17] Zhixuan Lin, Yi-Wu Fu, Skand Vishwanath Peri, Weihao Sun, Gautam Singh, Fei Deng, Jindong Jiang, and Sungjin Ahn. SPACE: Unsupervised Object-Oriented Scene Representation via Spatial Attention and Decomposition. In *ICLR*, 2020. URL `https://openreview.net/forum?id=rkl03ySYDH`.

[18] Klaus Greff, Sjoerd van Steenkiste, and Jürgen Schmidhuber. Neural expectation maximization. In *NeurIPS*, 2017. URL `https://arxiv.org/abs/1708.03498`.

[19] Sjoerd van Steenkiste, Michael Chang, Klaus Greff, and Jürgen Schmidhuber. Relational neural expectation maximization: Unsupervised discovery of objects and their interactions. In *ICLR*, 2018. URL `https://openreview.net/forum?id=ryH20GbRW`.

[20] Rishi Veerapaneni, John D. Co-Reyes, Michael Chang, Michael Janner, Chelsea Finn, Jiajun Wu, Joshua Tenenbaum, and Sergey Levine. Entity Abstraction in Visual Model-based Reinforcement Learning. In *Conference on Robot Learning*, 2019. URL `https://arxiv.org/abs/1910.12827`.

[21] Daniel Zoran, Rishabh Kabra, Alexander Lerchner, and Danilo J. Rezende. PARTS: Unsupervised segmentation with slots, attention and independence maximization. In *ICCV*, 2021. URL `https://ieeexplore.ieee.org/document/9711314`.

[22] Marissa A Weis, Kashyap Chitta, Yash Sharma, Wieland Brendel, Matthias Bethge, Andreas Geiger, and Alexander S. Ecker. Benchmarking Unsupervised Object Representations for Video Sequences. *JMLR*, 2021. URL `https://jmlr.org/papers/v22/21-0199.html`.

[23] Rishabh Kabra, Daniel Zoran, Goker Erdogan, Loic Matthey, Antonia Creswell, Matthew Botvinick, Alexander Lerchner, and Christopher P. Burgess. SIMONe: View-invariant, temporally-abstracted object representations via unsupervised video decomposition. In *NeurIPS*, 2021. URL `https://openreview.net/forum?id=YSzTMntO1KY`.

[24] Manuel Traub, Sebastian Otte, Tobias Menge, Matthias Karlbauer, Jannik Thuemmel, and Martin V. Butz. Learning What and Where: Disentangling Location and Identity Tracking Without Supervision. In *ICLR*, 2023. URL `https://openreview.net/forum?id=NeDc-Ak-H_`.

[25] Sadra Safadoust and Fatma Güney. Multi-object discovery by low-dimensional object motion. In *ICCV*, 2023. URL `https://arxiv.org/abs/2307.08027`.

[26] Gautam Singh, Fei Deng, and Sungjin Ahn. Illiterate DALL-E Learns to Compose. In *ICLR*, 2022. URL `https://openreview.net/forum?id=h0OYV0We3oh`.

[27] Laurynas Karazija, Iro Laina, and Christian Rupprecht. ClevrTex: A Texture-Rich Benchmark for Unsupervised Multi-Object Segmentation. In *NeurIPS Track on Datasets and Benchmarks*, 2021. URL https://arxiv.org/abs/2111.10265.

[28] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets Robotics: The KITTI Dataset. *International Journal of Robotics Research*, 2013. URL https://www.cvlibs.net/publications/Geiger2013IJRR.pdf.

[29] Yafei Yang and Bo Yang. Promising or Elusive? Unsupervised Object Segmentation from Real-world Single Images. In *NeurIPS*, 2022. URL https://openreview.net/forum?id=DzPWTwfby5d.

[30] Ondrej Biza, Sjoerd van Steenkiste, Mehdi S. M. Sajjadi, Gamaleldin F. Elsayed, Aravindh Mahendran, and Thomas Kipf. Invariant slot attention: Object discovery with slot-centric reference frames. In *ICML*, 2023. URL https://arxiv.org/abs/2302.04973.

[31] Jinwoo Kim, Janghyuk Choi, Ho-Jin Choi, and Seon Joo Kim. Shepherding slots to objects: Towards stable and robust object-centric learning. In *CVPR*, 2023. URL https://arxiv.org/abs/2303.17842.

[32] Adeel Pervez, Phillip Lippe, and Efstratios Gavves. Differentiable mathematical programming for object-centric representation learning. In *ICLR*, 2023. URL https://openreview.net/forum?id=1J-ZTr7aypY.

[33] Martin Engelcke, Oiwi Parker Jones, and Ingmar Posner. GENESIS-V2: Inferring Unordered Object Representations without Iterative Refinement. In *NeurIPS*, 2021. URL https://openreview.net/forum?id=nRBZWEUhIhW.

[34] Michael Chang, Thomas L. Griffiths, and Sergey Levine. Object representations as fixed points: Training iterative refinement algorithms with implicit differentiation. In *NeurIPS*, 2022. URL https://arxiv.org/abs/2207.00787.

[35] Baoxiong Jia, Yu Liu, and Siyuan Huang. Improving object-centric learning with query optimization. In *ICLR*, 2023. URL https://arxiv.org/abs/2210.08990.

[36] Jindong Jiang, Fei Deng, Gautam Singh, and Sungjin Ahn. Object-centric slot diffusion. In *NeurIPS*, 2023. URL https://arxiv.org/abs/2303.10834.

[37] Ziyi Wu, Jingyu Hu, Wuyue Lu, Igor Gilitschenski, and Animesh Garg. Slotdiffusion: Object-centric generative modeling with diffusion models. In *NeurIPS*, 2023. URL https://arxiv.org/abs/2305.11281.

[38] Zhipeng Bao, Pavel Tokmakov, Yu-Xiong Wang, Adrien Gaidon, and Martial Hebert. Object discovery from motion-guided tokens. In *CVPR*, 2023. URL https://arxiv.org/abs/2303.15555.

[39] Mahmoud Assran, Mathilde Caron, Ishan Misra, Piotr Bojanowski, Florian Bordes, Pascal Vincent, Armand Joulin, Michael G. Rabbat, and Nicolas Ballas. Masked Siamese Networks for Label-Efficient Learning. In *ECCV*, 2022. URL https://arxiv.org/abs/2204.07141.

[40] Xinlei Chen, Saining Xie, and Kaiming He. An Empirical Study of Training Self-Supervised Vision Transformers. *ICCV*, 2021. URL https://arxiv.org/abs/2104.02057.

[41] Rui Qian, Shuangrui Ding, Xian Liu, and Dahua Lin. Semantics meets temporal correspondence: Self-supervised object-centric learning in videos. In *ICCV*, 2023. URL https://arxiv.org/abs/2308.09951.

[42] Görkay Aydemir, Weidi Xie, and Fatma Güney. Self-supervised object-centric learning for videos. In *NeurIPS*, 2023. URL https://arxiv.org/abs/2310.06907.

[43] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *ICLR*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

[44] Matthias Tangemann, Steffen Schneider, Julius von Kügelgen, Francesco Locatello, Peter Gehler, Thomas Brox, Matthias Kümmerer, Matthias Bethge, and Bernhard Schölkopf. Unsupervised object learning via common fate. In *CLeaR*, 2023. URL `https://arxiv.org/abs/2110.06562`.

[45] Allan Jabri, Andrew Owens, and Alexei Efros. Space-time correspondence as a contrastive random walk. In *NeurIPS*, 2020. URL `https://arxiv.org/abs/2006.14613`.

[46] Linjie Yang, Yuchen Fan, and Ning Xu. Video instance segmentation. In *ICCV*, 2019. URL `https://arxiv.org/abs/1905.04804`.

[47] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alexander Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv:1704.00675*, 2017. URL `https://arxiv.org/abs/1704.00675`.

[48] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common Objects in Context. In *ECCV*, 2014. URL `https://arxiv.org/abs/1405.0312`.

[49] Jordi Pont-Tuset, Pablo Arbeláez, Jonathan T. Barron, Ferran Marques, and Jitendra Malik. Multiscale Combinatorial Grouping for Image Segmentation and Object Proposal Generation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(1), 2017. doi: 10.1109/TPAMI.2016.2537320. URL `https://ieeexplore.ieee.org/document/7423791`.

[50] Jason Tyler Rolfe. Discrete variational autoencoders. In *ICLR*, 2017. URL `https://openreview.net/forum?id=ryMxXPFex`.

[51] Christian Gumbsch, Martin V Butz, and Georg Martius. Sparsely changing latent states for prediction and planning in partially observable domains. In *NeurIPS*, 2021. URL `https://arxiv.org/abs/2110.15949`.

[52] Ziyi Wu, Nikita Dvornik, Klaus Greff, Thomas Kipf, and Animesh Garg. Slotformer: Unsupervised visual dynamics simulation with object-centric models. In *ICLR*, 2023. URL `https://openreview.net/forum?id=TFbwV6I0VLg`.

[53] Andrii Zadaianchuk, Maximilian Seitzer, and Georg Martius. Self-supervised Visual Reinforcement Learning with Object-centric Representations. In *ICLR*, 2020. URL `https://openreview.net/forum?id=xppLmXCbOw1`.

[54] Davide Mambelli, Frederik Träuble, Stefan Bauer, Bernhard Schölkopf, and Francesco Locatello. Compositional Multi-object Reinforcement Learning with Linear Relation Networks. In *ICLR Workshop on the Elements of Reasoning: Objects, Structure and Causality*, 2022. URL `https://openreview.net/forum?id=HFUxPr_I5ec`.

[55] Fan Feng and Sara Magliacane. Learning dynamic attribute-factored world models for efficient multi-object reinforcement learning. In *NeurIPS*, 2023. URL `https://arxiv.org/abs/2307.09205`.

[56] Mohammadreza Salehi, Efstratios Gavves, Cees G.M. Snoek, and Yuki M. Asano. Time does tell: Self-supervised time-tuning of dense image representations. In *ICCV*, 2023. URL `https://arxiv.org/abs/2308.11796`.

[57] Junyu Xie, Weidi Xie, and Andrew Zisserman. Segmenting moving objects via an object-centric layered representation. In *NeurIPS*, 2022. URL `https://arxiv.org/abs/2207.02206`.

[58] Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision, 2023. URL `https://arxiv.org/abs/2304.07193`.