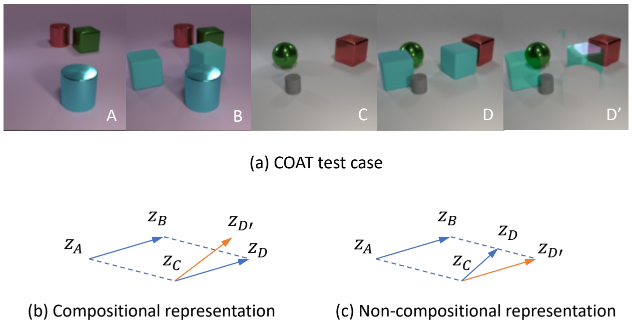
COAT: Measuring Object Compositionality in Emergent Representations

Sirui Xie [*1] **Ari Morcos** [2] **Song-Chun Zhu** [1 3] **Ramakrishna Vedantam** [2]

Abstract

Learning representations that can decompose a multi-object scene into its constituent objects and recompose them flexibly is desirable for object-oriented reasoning and planning. Built upon object masks in the pixel space, existing metrics for objectness can only evaluate generative models with an object-specific "slot" structure. We propose to directly measure compositionality in the representation space as a form of objectness, making such evaluations tractable for a wider class of models. Our metric, COAT (Compositional Object Algebra Test), evaluates if a generic representation exhibits certain geometric properties that underpin object compositionality beyond what is already captured by the raw pixel space. Our experiments on the popular CLEVR (Johnson et.al., 2018) domain reveal that existing disentanglement-based generative models are not as compositional as one might expect, suggesting room for further modeling improvements. We hope our work allows for a unified evaluation of object-centric representations, spanning generative as well as discriminative, self-supervised models.



(a) COAT test case

(b) Compositional representation

(c) Non-compositional representation

Figure 1. An illustration of the *parallelogram* in COAT. The transformation from scene A to scene B is to add two blue rubber cubes, so is the one from C to D. Hence in the representation space, we would expect the translation vector $\mathbf{z}_B - \mathbf{z}_A$ to be identical to $\mathbf{z}_D - \mathbf{z}_C$. Further, we would expect the parallelogram not to hold for A, B, C and hard negative D'. Here D' is the pixel-level hard negative, resulted from $B - A + C$ in the pixel space.

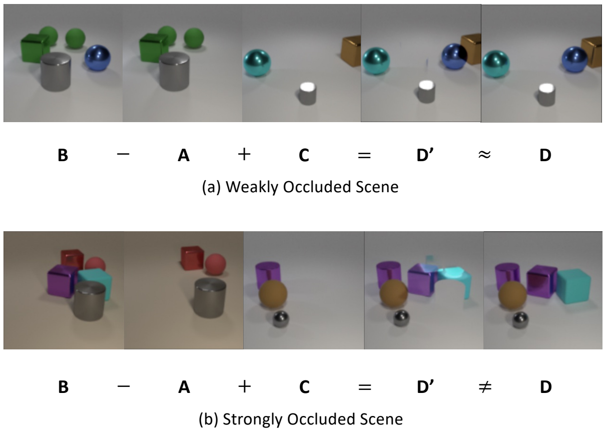
1. Introduction

Understanding a complex, visual scene in terms of the objects in it and their properties is an important scene understanding problem. A lot of work in representation learning recently has focused on the property of "objectness", where the goal is to form abstract, low dimensional representations $\mathbf{z} \in \mathbb{R}^D$ of input scenes $\mathbf{x} \in \mathbb{R}^N$ $(D << N)$ that parse the scene into its constituent objects such that these object primitives can be recomposed together to parse entirely novel scenes (Burgess et al., 2019; Bapst et al., 2019; Goyal et al., 2020; Greff et al., 2019; 2020; Kosiorek et al., 2018; Singh et al., 2021; van Steenkiste et al., 2018; Ali Eslami et al., 2016; Huang & Murphy, 2015). Such representations can support reasoning and higher level tasks such as counting (Chattopadhyay et al., 2017; Agrawal et al., 2015), abstraction (Higgins et al., 2016; Vedantam et al., 2020; 2018), puzzle solving (Barrett et al., 2018) and reinforcement learning (Bapst et al., 2019). More generally, objectness is a promising inductive bias for generalization to out-of-distribution scenes (Locatello et al., 2020; Chattopadhyay et al., 2017).

One can formalize the notion of object-centric representations or abstractions from the viewpoint of compositionality (Mikolov et al., 2013a) or disentanglement (Higgins et al., 2016). For compositionality, we aim to learn representations featuring low-level primitives that can be composed via simple vector operations to represent more complex inputs. Critically, such a representation should be learnable from a small set of primitive combinations and enable those vector operations to generalize to any combination of primitives (Lake & Baroni, 2018; Radford et al., 2015; Mitchell & Lapata, 2008). Disentanglement takes a stricter view in which we not only aim to infer and compose different factors of variation, but also require that these factors are partitioned into distinct dimensions z_i or K "slots", namely, $\mathbf{z}_k \in \mathbb{R}^{\frac{D}{K}}$ of the representation space $\mathcal{Z}$ to recover

Work done when SX was an intern at FAIR. [1]Department of Computer Science, UCLA [2]Fundamental AI Research (FAIR, Meta Inc.) [3]Department of Statistics, UCLA. Correspondence to: Sirui Xie <srxie@ucla.edu>.

Proceedings of the 39th International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022. Copyright 2022 by the author(s).



(a) Weakly Occluded Scene

(b) Strongly Occluded Scene

Figure 2. An illustration of *trivial compositionality* vs *object compositionality* and the importance of occlusion for measuring the latter. In (a), compositionality is trivial since an algebra $B-A+C$ in the pixel representation can obtain a D' that is almost the same as the D resulted from the transformation in the semantic space. In (b), compositionality is non-trivial and requires object-centric abstraction. An algebra $B-A+C$ in the pixel space results in a D' that is obviously different from the ground-truth D.

ground-truth, statistically independent factors of variation that generate the observations. However, representations may be compositional without being disentangled. For example, applying a change of basis through a rotation to a disentangled representation will yield a representation that is still compositional but no longer disentangled. Nevertheless, the goal of uncovering true factors of variation in a data driven manner remains conceptually appealing, and there has been a lot of interest in learning disentangled, object-centric representations (Locatello et al., 2020; Greff et al., 2019; Singh et al., 2021).

Slot-based Approaches. Slot-based approaches to objectness aim to disentangle K statistically independent subspaces or slots (Greff et al., 2019; Locatello et al., 2020) $\mathbf{z}_k \in \mathbb{R}^{\frac{D}{K}}$ such that each slot specializes to capture all relevant ground truth properties of one of the objects in the observed scene. Evaluation of such models is typically done using clustering-based metrics by decoding each of the slots back to the pixel space with a generator.

Slot-Free approaches. Although slot-based approaches to measuring objectness are relatively simple to define and evaluate, the strict requirement of slot-based delineation in the representations limits its applicability to architectures without slots. This downside is particularly relevant as most commonly utilized models for visual recognition such as residual networks (He et al., 2015) or newer models like vision transformers (Dosovitskiy et al., 2020) lack such slot-based structure. This challenge has led to a gap between the generic architectures used for visual recognition and slot-based approaches used for learning object-centric representations. In this paper, we bridge this gap by measuring compositionality in the emergent representations, providing a unified treatment of slot-based and slot-free approaches. We propose a new metric, COAT– Compositional Object Algebra Test – and associated testing corpus based on the CLEVR (Johnson et al., 2017) domain for measuring object-centric compositionality.

If a representation is compositional with respect to objects, we would expect that the change in a scene's representation when the same input space transformation is applied is consistent across scenes. For example, consider four images A, B, C, D where A:B::C:D (Figure 1). If two blue objects are added to A to yield B and the same blue objects are added to C (at the same locations) to yield D, then we would expect some analogical structure to be present in the corresponding representations as well. Taking a geometric perspective, if this is true, we would expect $\overrightarrow{AB}$ to be parallel to $\overrightarrow{CD}$. Equivalently, if one can compose D from A, B, C through translation operations in the latent space, to check their equivariance, it should form a parallelogram. Such emergent properties have previously been studied in the context of word representations (and have indeed emerged without any explicit slot-like structure or specific inductive bias in some cases (Mikolov et al., 2013b)). We demonstrate that one can evaluate object-centric representations against similar geometric structures, by showing how to translate the geometric desiderata into a concretely measurable device.

Interestingly, when measuring transformations across closely related scenes, even a representation such as $q(\mathbf{x}) = \mathbf{x}$ can be trivially compositional. For example, Figure 2a portrays a situation in which compositionality is trivially achieved in the pixel space representation. This occurs because there is very little occlusion in this scene and as such, the representation of objects in pixel-space is inherently independent since the pixels corresponding to each object are non-overlapping. Thus, when constructing a metric for compositionality one needs to carefully measure to what extent an approach captures more abstract, non-trivial compositionality. We do so by populating the testing corpus with images like Figure 2b with stronger occlusion such that the representations of objects is intertwined even in pixel space, and use $q(\mathbf{x}) = \mathbf{x}$ as a null, trivially compositional representation to contextualize our metric. Any learned representation, in order to exhibit non-trivial compositionality, has to be able to reject the null hypothesis that it is no better than the trivial model in order to receive a COAT score.

In terms of the metric itself, our key technical challenges are centered around how to best measure the extent to which a representation shows an (approximate) parallelogram structure. For instance, some representations might be very closely concentrated in some part of the output feature space, while others might be more spread out. Any comparisons of parallelogram structure have to take such global statistics into account in order to be relevant over the course of training a particular model, and across different models spanning different design choices (*e.g.,* generative *vs* contrastive, slot-based *vs* slot free, *etc.*). Moreover, a representation may short-circuit the parallelogram test by discarding information or not forming object abstraction at all. We address these desiderata by normalizing our metric, correcting for chance and a careful design of hard negative examples to capture compositionality in object-centric representations.

Overall, our contributions are as follows:

- We propose a novel measure, COAT for evaluating compositionality in emergent representations (see Section 3.3 for a discussion of the benefits of measuring compositionality over disentanglement)
- We demonstrate the importance of comparing to pixel-space as a null representation when evaluating compositionality (Section 4)
- We evaluate a suite of emergent representations, spanning models with various assumptions and inductive biases (Section 5)
- In an intriguing negative result, we demonstrate that representations learned by state-of-the-art models for disentangling objects are not as compositional as one might expect, especially with respect to pixel-space compositionality, hinting at the need for further modeling improvements (Table 2)

2. Related Work

Generative Models of Multi-Object Scenes. Generative modeling of multi-object scenes is a long-standing problem in computer vision (Zhu & Yuille, 1996; Tu & Zhu, 2002). While the previous work focuses on segmenting the pixel space to match the ground-truth, more recent interest has shifted to representation learning – hypothesizing that structural constraints in the representations will facilitate transfer to various visual reasoning (Barrett et al., 2018) or planning tasks (Schölkopf et al., 2021; Veerapaneni et al., 2019). MONet (Burgess et al., 2019), IODINE (Greff et al., 2019) and Slot Attention (Locatello et al., 2020), among other recent works (Greff et al., 2017; Kim & Mnih, 2018; Engelcke et al., 2019), demonstrate that one can learn disentangled, object-specific slots in latent representations on perceptually simple datasets like CLEVR (Johnson et al., 2017), although progress needs to be made to generalize to more perceptually challenging domains proposed by Karazija et al. (2021). In contrast, our focus is not on perceptual difficulty, but in simply measuring the extent to which compositionality is captured by existing CLEVR-domain models[1].

Disentanglement Metrics. Most of the slot-based object-centric models discussed above decode the slots back to the pixel space, and utilize clustering based approaches for comparing segmentations, such as the Adjusted Rand Index (ARI) (Hubert & Arabie, 1985) for evaluation. ARI aims to measure the similarity between the pixel mask for an object predicted by a generative model given a slot with the ground truth mask for that object in the original image, and has a number of desirable properties: it is bounded, normalized and corrected for chance, but it only indirectly measures the representations because a mapping back to the image space is always needed (Burgess et al., 2019; Greff et al., 2019; Locatello et al., 2020). In contrast, our metric is decoder-free, does not assume a slot-based structure and measures compositionality instead of disentanglement, broadening its applicability to a much wider range of models and settings. We anticipate this to facilitate more modeling avenues for research on object-centric representation learning.

As opposed to objects, there is another line of work which focuses on disentanglement at the attribute level, and which typically models scenes with single objects (though not always). To evaluate how well factors of variation such as pose, size, shape *etc.* are uncovered, these works adopt approaches like fitting a linear classifier or performing majority vote classifiers on each latent dimension (Higgins et al., 2016; Kim & Mnih, 2018; Chen et al., 2019), the mean distance between the classification errors of the two latent dimensions that are most predictable (Kumar et al., 2018), mutual information between the representation and the ground truth (Chen et al., 2016), and a dimension-wise entropy reflecting the usefulness of the dimension to predict a single factor to variation (Eastwood & Williams, 2018). In contrast to these methods, our metric does not need a generative model, nor a partitioning of the latent space in terms of individual factors of variation in the latent space[2], nor annotations of all the ground truth factors of variation. Moreover, previous work focuses on attribute-level disentanglement while we are concerned with object-centric, compositional representation learning.

Qualitative Analogical Structures. Our metric comes with an analogy test corpus which is inspired by pioneering work from Mikolov et al. (2013b); Radford et al. (2015), where they show the learned word embeddings enable simple linear algebra for analogical reasoning. Such a paradigm was later adopted by Eslami et al. (2018), Ha & Eck (2017) and Achlioptas et al. (2018) for static images, sketches, and 3D point clouds respectively. While these works largely provide qualitative analysis/ visualizations of representations, we aim to quantify the extent to which there exists such an analogical structure in the representations.

[1] With a minor modification, namely that we allow different colors for the background

[2] Note this is more challenging for slots in object-centric learning, as opposed to attribute disentanglement where the size of a slot is usually one.

Quantifying Compositionality. Andreas (2019) propose a learning based method to identify the extent of compositional structure in any generic, emergent representation, by learning an approximated composition function. While their approach is very generic and broadly applicable compared to ours, our work makes a number of more specific innovations to appropriately measure compositional structure for object-centric representations. Firstly, we note that the metric proposed in Andreas (2019) does not account for pixel-level hard negatives, which is a potential shortcut test constructed by applying the compositional operator in the raw input space as opposed to the latent representation space (which was not a concern for their applications). Further, our metric corrects for representation collapse and by adjusting for chance, which the metric in Andreas (2019) does not account for. That metric may give a high score to a collapsed representation, say $q(\mathbf{x}) = \mathbf{0}$. Finally, Keysers et al. (2020) measure compositional generalization in sequence to sequence tasks by constructing various evaluation tasks with different difficulties for evaluation. In contrast, our evaluation is towards objectness in learned, continuous valued representations rather than the difficulty in compositional generalization entailed by a sequence to sequence task.

3. Background

In this section, we make connections between disentanglement representation learning (Locatello et al., 2020; Greff et al., 2019), causal representation learning (Schölkopf et al., 2021), compositionality (Lake & Baroni, 2018), and equivariance (Jayaraman & Grauman, 2015) – notions which underpin both the models we evaluate, as well as the properties we choose to measure in our COAT metric.

3.1. Disentanglement

Attribute-level disentanglement. For attribute-level disentanglement approaches (Higgins et al., 2016), the goal is to recover the ground truth factors of variation used to generate the dataset (*e.g.,* the size, color orientation and shape of 2-D shapes (Higgins et al., 2016)). This is achieved by learning representations using variational autoencoders with a factorized prior $p(\mathbf{z}) = \prod_i p(z_i)$ (Higgins et al., 2016) which indirectly encourages the aggregate posterior $\int_{\mathbf{x}} q(\mathbf{z}|\mathbf{x})p(\mathbf{x})dx$ that generates the representations to be factorized as well (Hoffman & Johnson, 2016) or by encouraging such behavior more explicitly (Chen et al., 2019).

Object-Centric disentanglement A parallel line of work focuses on object-centric disentanglement (Greff et al., 2019; Locatello et al., 2020), where instead of attempting to fit all the information corresponding to an entire object into a single, independent latent dimension, one attempts to learn K independent sub-spaces or 'slots', *i.e.,* $\mathbf{z} = [\mathbf{z}_0, \cdots, \mathbf{z}_K]$, with the hope of specializing each object to a slot in the input scene. In addition to this independence assumption used in the latent space, such work also often utilizes functional constraints, which are inspired by the independent causal mechanism (ICM) principle (Schölkopf et al., 2021). Essentially, they start with (1) a pixel-space representation $\mathbf{u}_i \in \mathbb{R}^N$ to disentangle each object separately in the pixel observations $\mathbf{x} \in \mathbb{R}^N$ and (2) a vector of latent abstract causal variables $\mathbf{z}_i \in \mathbb{R}^{\frac{D}{K}}$, with a functional constraint on the decoder, namely that $\mathbf{u}_i = f_i(\mathbf{z}_i, \epsilon)$, where ϵ is a source of noise. The $\mathbf{u}_i$ are then combined with a function g which is often hand-coded to be a mixture of gaussians (Burgess et al., 2019; Greff et al., 2019; Locatello et al., 2020) (where the weight of each gaussian is a pixel-level mask, and the predicted means are the pixel values) or a more complex generative process (Ali Eslami et al., 2016; Huang & Murphy, 2015). Learning often proceeds by fitting g, f and an image encoder q jointly, where q maps input images $\mathbf{x}$ to latent representations $\mathbf{z}$. Intuitively, the functional constraint means that the mechanics of image formation are the same, regardless of the object in question. In addition, the popular Slot Attention (Locatello et al., 2020) also has a strong inductive bias based on self-attention on the encoder which aids emergence of effective slot representations.

Object- *vs* Attribute- disentanglement Conceptually, popular attribute-level disentanglement models such as β-VAE (Higgins et al., 2016) or β-TC-VAE (Kim & Mnih, 2018) can be thought of as learning statistically independent slots of size 1, with a relaxation of the functional constraint that "object-level" disentanglement models have. As such, it is not *a priori* obvious that so called "attribute-level" disentanglement models would fail when trained on multi-object scenes to uncover object-centric representations. To our knowledge, our work provides the first evaluation to quantitatively measure objectness in such "attribute-level" models, along with an extensive study of the extent to which "object-level" disentanglement models are compositional.

3.2. Compositionality

Compositional representations of scenes that enable the flexible addition or removal of objects to/from a scene and the ability to reason with these objects are potentially useful for a large number of applications including counting (Chattopadhyay et al., 2017), reasoning (Barrett et al., 2018), and out of distribution generalization (Lake & Baroni, 2018; Gordon et al., 2019), and potentially useful inductive biases for effective regularization with techniques like techniques like mixup (Zhang et al., 2017). Beyond these benefits, a compositional representation is conceptually easier to measure, compared to a disentangled representation which often needs access to the ground truth factors of variation, which are not always available in a lot of real world applications. Finally, compositional representations can also aid interpretability (Andreas, 2019; Kottur et al., 2017).

Disentanglement implies (approximate) compositionality. Moreover, disentanglement is closely related to compositionality. Consider a continuous valued disentangled representation $\mathbf{z}$ where each scalar z_i represents a factor of variation. Let $\mathbf{z}_A$ be the representation of image A and $\mathbf{z}_C$ be the representation of image C. Let z_0 be the dimension in the disentangled feature that corresponds to the size of an object, for instance. Increasing z_0 by a value ϵ should increase the size of the object for both image A as well as C, yielding images B and D respectively. Thus, a translation by ϵ in this case gives us compositionality in terms of size in the input domain. Strictly speaking, one might have to translate by a different amount ϵ and say, δ to obtain the same increase in shape for A and C, in general but the resultant difference vectors are still parallel to each other (Figure 1). In our work, we measure both notions of compositionality, by checking for an exact parallelogram structure as well as simply checking if the vectors are parallel (Section 4). While we explain attribute level disentanglement for ease of explanation, a similar argument can be made for "slot" based disentanglement with functional constraints (see Appendix A.1 for an informal proof). Moreover, any full rank linear transformation of a disentangled representation (such as a rotation) will still be compositional (while retaining all the information in the original representation) but not disentangled (see Appendix A.2). Thus, exact disentanglement implies perfect compositionality with respect to the ground-truth factors of variation.

Connection to Equivariance. The notion of compositionality we discuss here is an instantiation of group equivariance which has been well studied in the representation learning literature (Jayaraman & Grauman, 2015; Gordon et al., 2019). Specifically, our measure of compositionality is more formally expressed as to translation equivariance in the representation space. Since a slot-based, disentangled representation is compositional with repsect to translations (as discussed above), we measure compositionality with respect to the same translation operation for slot-free models, essentially testing to what extent a slot-free model behaves compositionally as a slot-based model ideally would. However, in principle, one could evaluate with other operations in the latent space or even learn them (Andreas, 2019). Nevertheless, we believe our contributions such as choice of hard negatives, proper normalization *etc.* will prove to be useful regardless of these orthogonal design choices.

3.3. Benefits of measuring compositionality over disentanglement

Overall, measuring compositionality instead of disentanglement for object-centric representation learning has the following benefits:

- Directly evaluates representations (which is what will be used for downstream tasks) without requiring a decoder that maps the representations back to pixel space.
- Allows one to evaluate objectness in representations when slot-based structure is not present.
- Related to the above, one can potentially handle scenes with highly variable numbers of objects in a distributed representation since one does not need to pre-specify the number of object-specific slots[3].
- While evaluation of slot-based disentanglement requires annotation of all the objects in the scene and their properties, measuring compositionality only requires us to know the relationship between two scenes which is much easier to annotate or obtain from say, videos. More specifically, annotating changes in scenes (in videos) is easier than densely annotating all objects (in images), and temporally close frames could be useful hard negatives.

4. Methods

Desiderata. While measuring equivariance is necessary for a useful compositional representation, it is not sufficient in practice. We impose the following additional desiderata to identify useful, compositional representations:

1. **Avoiding Shortcuts:** It is possible to "shortcut", for example the analogy test in Figure 1 by ignoring the color of the blue objects being added and simply learning a translation operator for "adding two objects" instead of "adding two blue objects". Such a representation would still be compositional, but potentially not useful for downstream tasks. As discussed in Section 1, another shortcut is when one performs no better than say a null model $q(\mathbf{x}) = \mathbf{x}$ in terms of capturing abstract compositional structure. If the model exhibits parallelogram structure, but doesn't capture the relevant information regarding the scene, our metric should penalize the model.
2. **Consistent Blank Slots:** When evaluating a model with say K slots, it is important to have a consistent representation of scenes with any number $k < K$ objects. Specifically, for a scene with k slots, there are $K - k$ "blank slots". One would desire a consistent representation of such slots across different input scenes, otherwise it would be hard to preserve the consistency in the consequence of applying the same vector operation.
3. **Calibration:** Finally, it is a non-trivial question in cases where the parallelogram structure is not exactly followed (which will almost always be the case in practice) how to quantify the *degree* to which A, B, C, D (Figure 1) follow the parallelogram structure in order to obtain a measure that is calibrated to give sensible measurements both across training checkpoints and across models.

[3] Although some works (Ali Eslami et al., 2016) do not have such conceptual limitations, they do not work as well (so far) as approaches based on a fixed number of slots.

4.1. COAT measure

The COAT measure comprises three parts: 1) A corpus with carefully designed analogy tests, 2) A binary pass/fail hypothesis test to detect shortcuts to compositionality, 3) a normalized measure reported for the cases where a model passes step 2. We discuss each of these elements below.

Analogy Test. We create a corpus with a set of 1600 analogy / compositionality tests[4], each containing images A, B, C, D (Figure 1). Following the observations in Section 1 we utilize images with sufficient occlusion (Figure 2) where the null representation $q(\mathbf{x}) = \mathbf{x}$ is not sufficient to capture compositionality. Finally, in our analogy test we assume that B is related to A, and D is related to C via a given transformation $\texttt{add}(obj_0, obj_1, obj_2, ...)$ (without loss of generality) which adds, for example, two blue objects to a base scene A and C respectively (Figure 1). Denoting $\mathbf{z}_A$ as the representation for scene A, we would like the corresponding representations to satisfy $\mathbf{z}_B - \mathbf{z}_A + \mathbf{z}_C = \mathbf{z}_D$. The degree to which this is satisfied is measured through a loss function $\mathcal{L}(\mathbf{z}_A, \mathbf{z}_B, \mathbf{z}_C, \mathbf{z}_D)$. We either measure approximate parallelogram structure or the degree to which $\overrightarrow{BA}$ and $\overrightarrow{DC}$ are parallel. For the former, we use $\mathcal{L}(\mathbf{z}_A, \mathbf{z}_B, \mathbf{z}_C, \mathbf{z}_D) = ||\mathbf{z}_B - \mathbf{z}_A + \mathbf{z}_C - \mathbf{z}_D||_2$; for the latter we use $\mathcal{L}(\mathbf{z}_A, \mathbf{z}_B, \mathbf{z}_C, \mathbf{z}_D) = acos(\mathbf{z}_B - \mathbf{z}_A, \mathbf{z}_D - \mathbf{z}_C)$. Check Appendix B for example test cases.

Shortcut Detection. As discussed above, a model might appear equivariant but not be compositional in a manner useful for downstream tasks. To test for this, we employ hard negatives (denoted $D' \neq D$), and compute the losses $\mathcal{L}(\mathbf{z}_A, \mathbf{z}_B, \mathbf{z}_C, \mathbf{z}_D)$ and $\mathcal{L}(\mathbf{z}_A, \mathbf{z}_B, \mathbf{z}_C, \mathbf{z}_{D'})$ for a one-sample proportion hypothesis test. Our null hypothesis H_0 is that there is no difference between the two values, *i.e.*,, $\texttt{Pr}[\mathcal{L}(\mathbf{z}_A, \mathbf{z}_B, \mathbf{z}_C, \mathbf{z}_D) < \mathcal{L}(\mathbf{z}_A, \mathbf{z}_B, \mathbf{z}_C, \mathbf{z}_{D'})] = 0.5$, with the alternative hypothesis being that the hard negatives incur a higher loss, namely, $\texttt{Pr}[\mathcal{L}(\mathbf{z}_A, \mathbf{z}_B, \mathbf{z}_C, \mathbf{z}_D) < \mathcal{L}(\mathbf{z}_A, \mathbf{z}_B, \mathbf{z}_C, \mathbf{z}_{D'})] > 0.5$. We use the standard test statistic for the one-sample proportion test with a significance level=0.005 to reject or fail to reject the null hypothesis.

Concretely, we utilize the following hard negatives (D'):

- Object-level: in D' there is one object different from or dropped from D. This tests that the representation captures properties of all objects in the scene, and not just say a subset of objects that are used for the transofrmation Figure 1.
- Attribute-level: D' has one object with one attribute (color, material, shape or size) different from D. This tests that the representation is sensitive to the properties of the object, and not just the location (for example).
- Pixel-level: $D' = B - A + C$ in the pixel space. This validates that the evaluated representation is better than a trivial representation $q(\mathbf{x}) = \mathbf{x}$.

If a model fails these tests, COAT will not provide an accurate evaluation of the model's compositionality. Therefore, we only apply the COAT metric to models which can pass the above test. Empirically, we find the hard negative tests for attributes like shape, material and pixel-level representations are most difficult to pass.

Normalization and Correction for Chance.

When comparing across different models, we need to calibrate the metric to ensure that model-specific biases such as differences in the concentration of features do not confound our estimates of whether the structure we desire approximately exists. Denoting by $\hat{D}$ a random image from the minibatch of examples, we use the following normalization and chance correction:

$$\texttt{COAT} = 1 - \frac{\mathcal{L}(\mathbf{z}_A, \mathbf{z}_B, \mathbf{z}_C, \mathbf{z}_D)}{\mathbb{E}_{\hat{D}}\left[\mathcal{L}(\mathbf{z}_A, \mathbf{z}_B, \mathbf{z}_C, \mathbf{z}_{\hat{D}})\right]}, \qquad (1)$$

The key idea being that we compute the average loss incurred by a random datapoint to calibrate the extent to which one claims that the desired structure is present. In practice, we use the minibatch size of 64 for calculating $\mathbb{E}_{\hat{D}}[\cdot]$. In terms of the two concrete losses for "perfectness", $L2-$COAT and for "parallelness", $acos-$COAT, we hypothesize for downstream tasks such as counting "perfectness" might be more important (since magnitude of vectors is important) but in other reasoning tasks relaxing it to "parallelness" might be sufficient.

4.2. Sanity Checks and Baselines

As a sanity check, we apply the COAT metric to (1) a disentangled vector representation that concatenates the symbolic attributes of all objects and (2) the vector representation of a random full-rank linear projection from (1). Note that (1) is a disentangled representation while (2) is entangled, but they are both compositional by our definition (see Appendix A.2), so they should both obtain a perfect score of 1.0 in our metric. Indeed, this is validated by our experiments. In contrast, disentanglement metric such as training a linear regressor along with a Hungarian matching towards the set of object attribute vectors (Locatello et al., 2020) would rate (1) highly, but not (2), despite the presence of compositional structure in a different basis set.

We next provide a baseline in terms of the COAT score for the trivial representation $q(\mathbf{x}) = \mathbf{x}$ (*e.g.,* taking pixel-space as our representation). Overall the score for the baseline is 75.47% in terms of L_2 and 36.28% for $acos$, demonstrating

[4] These 1600 test cases are obtained by rejection sampling from 100,000 scenes under a strong occlusion criterion. Moreover, in progressively increasing the test volume n, we find that the COAT score as well as hard-negative tests appear to be stable with respect to the number of samples (n) used to compute them at $n = 1000$.

that pixel-space is already somewhat compositional. Note that here we do not use the pixel-baseline image as a hard negative for a hypothesis test, but as a trivial representation to compare COAT scores for models against.

5. Experiments

Our goal is to identify key modeling choices for object-centric compositionality (Section 3) and evaluate them in a unified manner (Section 4). From a conceptual standpoint, several factors of variation across models stand out:

1. Whether one has a "slot structure" in the representation in conjunction with functional independence in the decoder (`Slot Structure`)
2. Whether there is an independence prior on the latents or some other mechanism to enforce disentangling a factorized aggregate posterior distribution (*e.g.,* β-TC-VAE (Chen et al., 2019)) (`Factorized Prior`)
3. Whether the model is generative or discriminative to isolate if one requires the use of a generative model for learning object-centric representations (`Training Paradigm`)
4. Whether we train the models on an IID dataset or a highly correlated training dataset (`Train Set`) (see Appendix C for more details).

Table 2 (left side) breaks down several models of interest along these axes, *i.e.,* β-TC-VAE (Chen et al., 2019), vanilla Auto-Encoder (Kramer, 1991) which we implement as a β-TC-VAE with $\beta = 0$, Slot Attention (Locatello et al., 2020), IODINE (Greff et al., 2019), and MoCo v2 (He et al., 2020; Chen et al., 2020) in a matrix of these factors. All models are trained with the default architectures and hyperparameters except that in β-TC-VAE we use latent dimension 256, and use the same encoder for MoCo. We modified CLEVR to include multiple background colors, which forces models to represent the background explicitly for compositional evaluation (see Appendix C for some examples and an ablation study against the COAT measure). To further support the usefulness of COAT, we implemented a TRE measure (Andreas, 2019), learning a transformer to approximate the compositional function. See Appendix D.

5.1. Autoencoder and β-TC-VAE

We generally found that a β-TCVAE as well as a vanilla autoencoder ($\beta = 0$) fail to learn object-centric representations. We sweep over $\beta \in \{0, 1, 2, 3\}$ and generally found a trend that with $\beta = 1, 2$ the models performs the best on the most challenging pixel-level hard negative test, albeit not passing it (Figure A7b and Figure A6b). Thus, without the "slot" structure or functional independence assumptions it appears that vanilla autoencoders and β-TC-VAE style models do not achieve object-centric compositionality.

5.2. Slot Attention

Next, we study the slot attention model that has a slot structure in the latent space, functional independence in the decoder and a strong inductive bias on the encoder q which infers the slots. This model in previous evaluations using ARI has achieved really strong results and thus given that it is likely "disentangled" one would expect it to also be "compositional" (Section 3).

Is slot attention compositional? We align the slots for the same objects across A, B and C, D respectively to facilitate the application of the COAT metric, we perform greedy matching (see Appendix E for more details). Overall, while slot attention passes the hard-negative statistical test (Table 2), in terms of the COAT score, the vanilla representation from slot-attention does not outperform the trivial pixel-level representation, achieving an L_2-COAT score of 48.55 ± 14.11 as opposed to a pixel-level score of 75.47.

What causes the poor performance of slot-attention? One of the key bottlenecks is that slot attention often assigns two different slots to the same object (Figure A10). This redundancy in the latent representation causes difficulties with a proper compositional structure emerging in the latent space. We hypothesize this redundancy is caused by the lack of a sparsity prior on the latent space (Locatello et al., 2020) compared to other models such as Greff et al. (2019). To account for this when computing COAT, we detect duplicates by measuring the cosine similarity between different pairs of slots $\mathbf{z}_i \in \mathbb{R}^{\frac{D}{K}}$ and $\mathbf{z}_j \in \mathbb{R}^{\frac{D}{K}}$ and replacing all duplicates with the mean of all other slots. This improves the L_2-COAT score to 60.70 ± 15.15 (Table 2), which is still worse than the pixel baseline. This result also indicates that the ARI metric (Locatello et al., 2020) is not sensitive to redundancy in the representations. In essence, it measures recall but not precision of the latent factors – while COAT tests for precision as well as recall.

Another issue is that slot attention does not have a consistent representation of "blank slots" which contain no objects, but instead has a more general notion of "invisible slots" which do not contribute to the reconstruction / generation because they are masked out, but still have some content in them (Figure A11). This makes blank slots difficult to detect and standardize without access to masks from the generative decoder. However, utilizing the decoder in this manner does elicit performance on the L_2-COAT metric that surpasses the pixel level baseline (77.02 ± 0.72 *vs* 75.47). Check Figure A12 for an visualization of the matching, where we can see the imperfect score may be caused by missing objects. This indicates that on it's own, the slot attention model does not exhibit object-centric compositionality without access to the weights of the particular decoder that has been learned in a model run. While this is not an issue for generative modeling, these redundancies and external dependency on

Table 1. Models, their inductive biases, their training paradigms, their training sets, and their performance on ARI and COAT. "HN" is the Hard Negative Test; models need to pass all hard negative tests to obtain a COAT score, otherwise it is indicated with "-". Since representations directly obtained from slot attention do not perform well on the COAT metric, we also tried some post-processing: * indicates duplication removal, †indicates removing "invisible slots" with zero mask weights. (w/o occlusion) and (w/ occlusion) indicate non-occluded (Figure 2a) and strongly occluded (Figure 2b) test cases. Statistics are Mean and SEM summarized over 5 random seeds.

	Slot Structure $\mathbf{z} = [\mathbf{z}_0, \cdots, \mathbf{z}_K]$	Factorized Prior $p(\mathbf{z}) = \Pi_{k=1}^{K} p(\mathbf{z}_k)$	Training Paradigm	Train Set	ARI (%)	HN L_2	L_2-COAT (%)	HN acos	$acos$-COAT (%)
Pixel baseline (w/ occlusion)	N/A	N/A	N/A	N/A	N/A	N/A	75.47	N/A	36.28
Pixel baseline (w/o occlusion)	N/A	N/A	N/A	N/A	N/A	N/A	97.18	N/A	73.17
Auto-encoder(w/ occlusion)	No	No	Generative	IID	N/A	Fail	-	Fail	-
β-TC-VAE (w/ occlusion)	No	Yes	Generative	IID	N/A	Fail	-	Fail	-
Slot attention (w/ occlusion)	Yes	No	Generative	IID	95.53 ± 1.84	Pass	48.55 ± 14.11	Pass	21.53 ± 10.73
Slot attention* (w/ occlusion)	Yes	No	Generative	IID	95.53 ± 1.84	Pass	60.70 ± 15.55	Pass	31.18 ± 8.01
Slot attention*†(w/ occlusion)	Yes	No	Generative	IID	95.53 ± 1.84	Pass	77.07 ± 0.72	Pass	43.12 ± 0.78
Slot attention*†(w/o occlusion)	Yes	No	Generative	IID	95.53 ± 1.84	Pass	83.84 ± 6.23	Pass	47.45 ± 4.34
Slot attention*†(w/ occlusion)	Yes	No	Generative	CORR	69.12 ± 9.34	Pass	64.82 ± 9.20	Pass	31.95 ± 7.21
IODINE (w/ occlusion)	Yes	Yes	Generative	IID	92.21 ± 0.15	Pass	47.52 ± 0.29	Pass	16.33 ± 0.33
IODINE (w/ occlusion)	Yes	Yes	Generative	CORR	40.08 ± 8.90	Fail	-	Pass	9.16 ± 1.08
MoCo v2 ConvNet (w/ occlusion)	No	N/A	Discriminative	IID	N/A	Fail	-	Pass	14.05 ± 1.25

the generator might hurt performance of on downstream transfer learning or out-of-distribution generalization tasks where one usually does not have privileged access to a generator. Together, our results demonstrate that despite the strong inductive biases present in slot attention models, they do not exhibit improved object-centric compositionality relative to the raw pixel representation, demonstrating both the importance of comparison with the raw pixel baseline for contextualizing compositionality measures and highlighting potential for modeling improvements.

5.3. IODINE

In contrast to slot attention, IODINE (Greff et al., 2019) is a full Bayesian generative model with not only independence constraints in the variational encoder and decoder but also a factorized prior (key difference from slot attention being that the inductive bias in the encoder for computing the slots is more explicitly designed in IODINE instead of using a generic self-attention mechanism). Given the explicit prior, we found that IODINE more consistently represents "blank slots" (Figure A15) – however, while it passes all hard-negative tests (Table 2), it is not able to disentangle foreground objects from background properly (as also noted in the original paper (Greff et al., 2019) and a more recent work (Yu et al., 2021)), which sets it back in terms of the L_2-COAT score (47.52 ± 0.29). This is more or less equivalent to the vanilla slot attention model 48.55 ± 14.11 and again substantially lower than the pixel baseline of 75.47.

5.4. MoCo v2 with ConvNet

Next, we evaluate the MoCo v2 (He et al., 2019) self-supervised learning model trained on our IID set using the same architecture used for the β-TC-VAE. Over the course of training, the model appears to pass the hard-negative test in terms of the $acos$-COAT score (Table 2) (which the β-TC-VAE models failed to do) – indicating some initial promise. However, the model is still substantially worse than the corresponding raw pixel baseline (14.05 ± 1.25 *vs* 36.28) suggesting need for a more directed exploration of this direction, which is out of scope for our current paper.

5.5. Influence of correlated training set

Next, we induce correlations between objects in a given scene, to understand how patterns in the data impact different models' ability to learn object-centric compositional representations. Specifically, we generate a set of training images with extremely correlated and cluttered objects that have the same color and material (see Appendix C.2). For slot-attention, even when we utilize the generative model to detect "invisible slots", we still observe a drop in the performance (64.82 ± 9.20 *vs* 77.07 ± 0.72), indicating that the model is not as robust as one might have hoped. However, slot attention is still better than IODINE which fails the pixel-level hard negative test in this setting[5]. This again indicates that the inductive bias used in slot attention based on positional embeddings and self-attention is more robust than those in the encoder of IODINE.

6. Conclusion

In this work we presented a new metric, COAT, for measuring object-level compositionality in emergent representa-

[5] Locatello et al. (2020) show a similar comparison to IODINE using the ARI metric in grayscale images.

tions. Our metric comprises of two parts: 1) a hypothesis test, testing for a null hypothesis that the representations being evaluated do not capture any more compositionality than carefully crafted, trivial baselines such as pixel-space, and 2) a measure of the extent to which compositional or analogical structure is present in the representations with respect to translations in the feature space. We applied COAT to a number of object-centric representations – spanning a large set of modeling assumptions, finding that, somewhat surprisingly, state-of-the-art approaches for object-centric disentanglement are often not compositional beyond trivial pixel level baselines even in the presence of severe occlusion. We hope that our metric, COAT galvanizes future work on object-centric representation learning from a more unified, model-agnostic viewpoint.

Acknowledgements

Authors would like to thank Stephane Deny at Aalto University for his indispensable contributions in initiating and formulating this project, Justin Johnson, Kamalika Chaudhuri at FAIR and Marco Baroni at ICREA for useful discussion.

References

Achlioptas, P., Diamanti, O., Mitliagkas, I., and Guibas, L. Learning representations and generative models for 3d point clouds. In *International conference on machine learning*, pp. 40–49. PMLR, 2018.

Agrawal, A., Lu, J., Antol, S., Mitchell, M., Lawrence Zitnick, C., Batra, D., and Parikh, D. VQA: Visual question answering. May 2015.

Ali Eslami, S. M., Heess, N., Weber, T., Tassa, Y., Szepesvari, D., Kavukcuoglu, K., and Hinton, G. E. Attend, infer, repeat: Fast scene understanding with generative models. March 2016.

Andreas, J. Measuring compositionality in representation learning. February 2019.

Bapst, V., Sanchez-Gonzalez, A., Doersch, C., Stachenfeld, K. L., Kohli, P., Battaglia, P. W., and Hamrick, J. B. Structured agents for physical construction. April 2019.

Barrett, D. G. T., Hill, F., Santoro, A., Morcos, A. S., and Lillicrap, T. Measuring abstract reasoning in neural networks. In *International Conference on Machine Learning (ICML)*, 2018.

Burgess, C. P., Matthey, L., Watters, N., Kabra, R., Higgins, I., Botvinick, M., and Lerchner, A. Monet: Unsupervised scene decomposition and representation. *arXiv preprint arXiv:1901.11390*, 2019.

Chattopadhyay, P., Vedantam, R., Selvaraju, R. R., Batra, D., and Parikh, D. Counting everyday objects in everyday scenes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Chen, R. T., Li, X., Grosse, R., and Duvenaud, D. Isolating sources of disentanglement in vaes. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pp. 2615–2625, 2019.

Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., and Abbeel, P. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, pp. 2180–2188, 2016.

Chen, X., Fan, H., Girshick, R. B., and He, K. Improved baselines with momentum contrastive learning. *CoRR*, abs/2003.04297, 2020. URL https://arxiv.org/abs/2003.04297.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale. October 2020.

Eastwood, C. and Williams, C. K. A framework for the quantitative evaluation of disentangled representations. In *International Conference on Learning Representations*, 2018.

Engelcke, M., Kosiorek, A. R., Jones, O. P., and Posner, I. Genesis: Generative scene inference and sampling with object-centric latent representations. *arXiv preprint arXiv:1907.13052*, 2019.

Eslami, S. A., Rezende, D. J., Besse, F., Viola, F., Morcos, A. S., Garnelo, M., Ruderman, A., Rusu, A. A., Danihelka, I., Gregor, K., et al. Neural scene representation and rendering. *Science*, 360(6394):1204–1210, 2018.

Gordon, J., Lopez-Paz, D., Baroni, M., and Bouchacourt, D. Permutation equivariant models for compositional generalization in language. In *International Conference on Learning Representations*, September 2019.

Goyal, A., Lamb, A., Gampa, P., Beaudoin, P., Levine, S., Blundell, C., Bengio, Y., and Mozer, M. Object files and schemata: Factorizing declarative and procedural knowledge in dynamical systems. June 2020.

Greff, K., Van Steenkiste, S., and Schmidhuber, J. Neural expectation maximization. *arXiv preprint arXiv:1708.03498*, 2017.

Greff, K., Kaufman, R. L., Kabra, R., Watters, N., Burgess, C., Zoran, D., Matthey, L., Botvinick, M., and Lerchner, A. Multi-object representation learning with iterative variational inference. In *International Conference on Machine Learning*, pp. 2424–2433. PMLR, 2019.

Greff, K., van Steenkiste, S., and Schmidhuber, J. On the binding problem in artificial neural networks. December 2020.

Ha, D. and Eck, D. A neural representation of sketch drawings. *arXiv preprint arXiv:1704.03477*, 2017.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. December 2015.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. Momentum contrast for unsupervised visual representation learning. November 2019.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9729–9738, 2020.

Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., and Lerchner, A. beta-vae: Learning basic visual concepts with a constrained variational framework. 2016.

Hoffman, M. D. and Johnson, M. J. Elbo surgery: yet another way to carve up the variational evidence lower bound. In *Workshop in Advances in Approximate Bayesian Inference, NIPS*, 2016.

Huang, J. and Murphy, K. Efficient inference in occlusion-aware generative models of images. November 2015.

Hubert, L. and Arabie, P. Comparing partitions. *Journal of classification*, 2(1):193–218, 1985.

Jayaraman, D. and Grauman, K. Learning image representations tied to ego-motion. May 2015.

Johnson, J., Hariharan, B., Van Der Maaten, L., Fei-Fei, L., Lawrence Zitnick, C., and Girshick, R. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2901–2910, 2017.

Karazija, L., Laina, I., and Rupprecht, C. ClevrTex: A Texture-Rich benchmark for unsupervised Multi-Object segmentation. 2021.

Keysers, D., Schärli, N., Scales, N., Buisman, H., Furrer, D., Kashubin, S., Momchev, N., Sinopalnikov, D., Stafiniak, L., Tihon, T., Tsarkov, D., Wang, X., van Zee, M., and Bousquet, O. Measuring compositional generalization: A comprehensive method on realistic data. In *International Conference on Learning Representations (ICLR)*, 2020.

Kim, H. and Mnih, A. Disentangling by factorising. In *International Conference on Machine Learning*, pp. 2649–2658. PMLR, 2018.

Kosiorek, A. R., Kim, H., Posner, I., and Teh, Y. W. Sequential attend, infer, repeat: Generative modelling of moving objects. June 2018.

Kottur, S., Moura, J., Lee, S., and Batra, D. Natural language does not emerge 'naturally' in Multi-Agent dialog. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 2962–2967, Copenhagen, Denmark, September 2017. Association for Computational Linguistics.

Kramer, M. A. Nonlinear principal component analysis using autoassociative neural networks. *AIChE journal*, 37(2):233–243, 1991.

Kumar, A., Sattigeri, P., and Balakrishnan, A. Variational inference of disentangled latent concepts from unlabeled observations. In *International Conference on Learning Representations*, 2018.

Lake, B. M. and Baroni, M. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *International Conference on Machine Learning (ICML)*, 2018.

Locatello, F., Weissenborn, D., Unterthiner, T., Mahendran, A., Heigold, G., Uszkoreit, J., Dosovitskiy, A., and Kipf, T. Object-centric learning with slot attention. *arXiv preprint arXiv:2006.15055*, 2020.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., and Dean, J. Distributed representations of words and phrases and their compositionality. October 2013a.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pp. 3111–3119, 2013b.

Mitchell, J. and Lapata, M. Vector-based models of semantic composition. *ACL*, 2008.

Radford, A., Metz, L., and Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. November 2015.

Schölkopf, B., Locatello, F., Bauer, S., Ke, N. R., Kalchbrenner, N., Goyal, A., and Bengio, Y. Towards causal representation learning. February 2021.

Singh, G., Deng, F., and Ahn, S. Illiterate DALL-E learns to compose. October 2021.

Tu, Z. and Zhu, S.-C. Image segmentation by data-driven markov chain monte carlo. *IEEE Transactions on pattern analysis and machine intelligence*, 24(5):657–673, 2002.

van Steenkiste, S., Chang, M., Greff, K., and Schmidhuber, J. Relational neural expectation maximization: Unsupervised discovery of objects and their interactions. February 2018.

Vedantam, R., Fischer, I., Huang, J., and Murphy, K. Generative models of visually grounded imagination. In *International Conference on Learning Representations (ICLR)*, 2018.

Vedantam, R., Szlam, A., Nickel, M., Morcos, A., and Lake, B. CURI: A benchmark for productive concept learning under uncertainty. October 2020.

Veerapaneni, R., Co-Reyes, J. D., Chang, M., Janner, M., Finn, C., Wu, J., Tenenbaum, J. B., and Levine, S. Entity abstraction in visual Model-Based reinforcement learning. October 2019.

Yu, P., Xie, S., Ma, X., Zhu, Y., Wu, Y. N., and Zhu, S.-C. Unsupervised foreground extraction via deep region competition. *Advances in Neural Information Processing Systems*, 34, 2021.

Zhang, H., Cisse, M., Dauphin, Y. N., and Lopez-Paz, D. mixup: Beyond empirical risk minimization. October 2017.

Zhu, S. C. and Yuille, A. Region competition: Unifying snakes, region growing, and bayes/mdl for multiband image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 18(9):884–900, 1996.

A. Relation Between Compositionality and Disentangling

A.1. Slot-based Disentangling Models are Compositional with Respect to Object Additions

Consider a slot-based feature space $\mathbf{z} = [\mathbf{z}_0, \cdots, \mathbf{z}_K]$ where each $\mathbf{z}_i \in \mathbb{R}^{\frac{D}{K}}$ with a functional constraint that uses individual slots to decode them back into the pixel space, namely $f_i(\mathbf{z}_i) = \mathbf{u}_i$, where $\mathbf{u}_i \in \mathbb{R}^N$ as explained in Section 3 in the main paper. Further, assume that $f_i = f_j$ for all $i, j \in 1, \cdots, K$ (this constraint is typically followed by models as well). Also, assume that the model represents all blank slots consistently, specifically, assume that $\mathbf{z}_k = \mathbf{0}$ where $\mathbf{0}$ represents a vector in $\mathbf{R}^{\frac{D}{K}}$ where each entry is 0, without loss of generality.

Now, assume image A is the original image we add an object to, in order to obtain image B, for the analogy test in Figure 1. Further assume that A has k objects. Thus, by assumption that blank slots are consistently represented, $\forall k' > k$, $\mathbf{z}'_k = \mathbf{0}$. Let $\delta \in \mathbb{R}^{\frac{D}{K}}$ be the offset to be added to a blank slot $\mathbf{z}_{k+1}$ in A in order to obtain B, which essentially corresponds to the addition of a blue object. Then, in order to add the same object to image C with $\hat{k}$ objects, one would add the same offset vector δ to the blank slot $\mathbf{z}_{\hat{k}+1}$. This is true, since we assume that the function $f_i(\mathbf{z}_i)$ is independent for each slot, and also $f_i = f_j$ for all $i, j \in 1, \cdots, K$.

Thus, by construction one achieves a parallelogram in the full original vector space $\mathbf{z}$ in which the representation exists, meaning that an object-centric disentangled slot-based representation is also compositional with respect to addition and subtration of objects from the scene.

A.2. Full-rank Transformations of Disentangled Representations are Compositional

Given a disentangled representation $q(\mathbf{x})$, and let $\mathbf{z}_A = q(A)$, $\mathbf{z}_B = q(B)$, $\mathbf{z}_C = q(C)$ and $\mathbf{z}_D = q(D)$, where $\mathbf{z} \in \mathbb{R}^D$ as in the rest of the paper. be the representations for four scenes in the analogy test Figure 1. Now, since disentangling implies compositionality, we have:

$$\mathbf{z}_B - \mathbf{z}_A + \mathbf{z}_C = \mathbf{z}_D \tag{2}$$

Next, consider a full rank matrix $W \in \mathbb{R}^{D \times D}$. Then, multiplying by W on both sides above, we get,

$$W\mathbf{z}_B - W\mathbf{z}_A + W\mathbf{z}_C = W\mathbf{z}_D \tag{3}$$

Now, notice that in general, $W\mathbf{z}_B$ is no longer disentangled with respect to the original factors of variation (*e.g.,* W could be a rotation matrix or a permutation matrix), but the resultant representation still satisfies the parallelogram property. Moreover, given invertibility of W because of it being full rank, it has all the information present in the original disentangled space meaning that it avoids collapse of the representation which might enable it to lose information of attributes or find some other "shortcut" to pass the analogy test trivially.

B. Example Test Cases

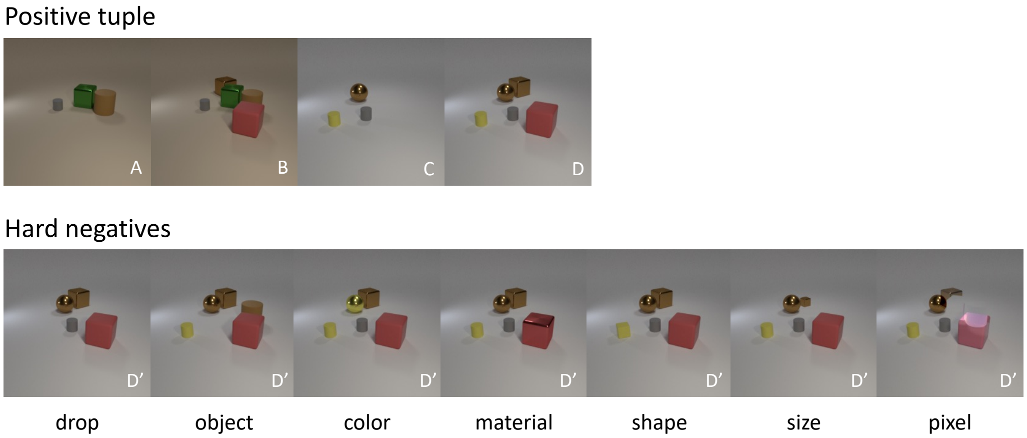


Figure A1. Example of the test corpus of COAT. A, B, C, D form the positive tuple, where the same transformation leads A to B and C to D. In the second row there are hard negatives D', where "drop" is dropping one object from D, "object" is changing one object from D, "color" is changing the color of one object from D, "material" is changing the material of one object from D, "shape" is changing the shape of one object from D, "size" is changing the size of one object from D, "pixel" is the result of $B - A + C$.

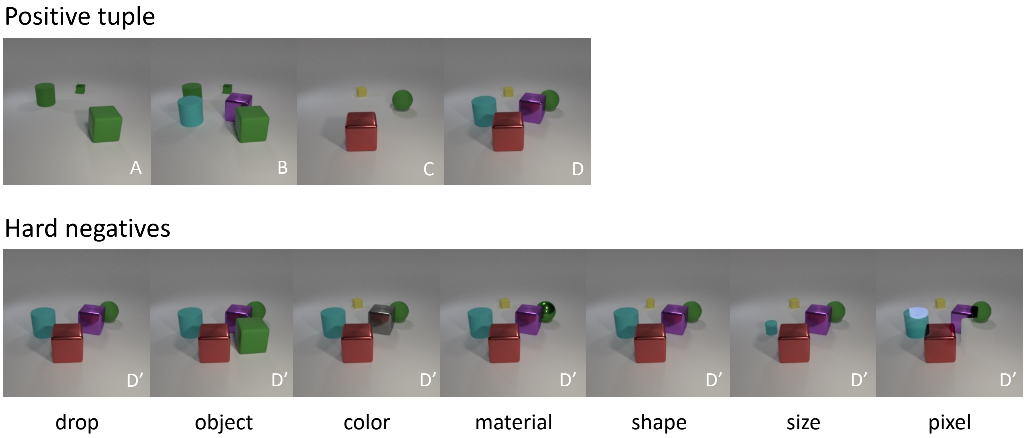


Figure A2. Example of the test corpus of COAT. A, B, C, D form the positive tuple, where the same transformation leads A to B and C to D. In the second row there are hard negatives D', where "drop" is dropping one object from D, "object" is changing one object from D, "color" is changing the color of one object from D, "material" is changing the material of one object from D, "shape" is changing the shape of one object from D, "size" is changing the size of one object from D, "pixel" is the result of $B - A + C$.

C. CLEVR with Colorful Background

C.1. Independently Identically Distributed Dataset

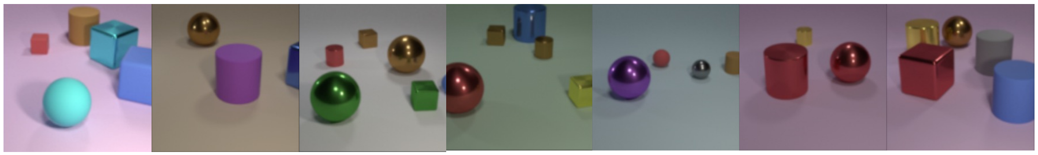


Figure A3. Example of the IID training data with colorful background

C.2. Correlated Dataset

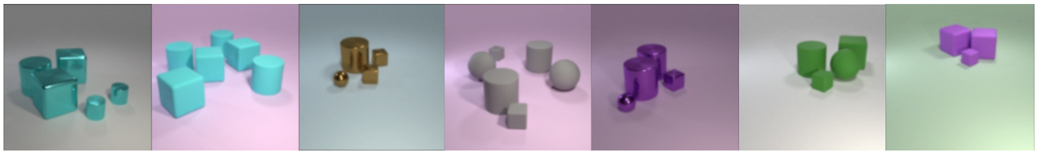


Figure A4. Example of the correlated training data with colorful background

C.3. Impact on COAT of using colorful background

To investigate the impact of colorful background on COAT, we apply the COAT measure to a test corpus with only white background. The COAT score of slot-based models (e.g. IODINE) are not affected much by this change, but slot-free models improve by $\approx 3\%$ (although they still fail the hard-negative tests). IODINE segments white background inconsistently in correlated images (see Fig. A15), which might explain this lack of improvement.

D. Results on Tree Reconstruction Error (TRE)

For completeness, we implemented a TRE measure, learning a transformer for the compositional function.

Table 2. Models, variants and Tree Reconstruction Error (TRE)

Slot attention	vanilla	5.79 ± 1.23	β-VAE	$\beta = 0$	10.32 ± 3.56
	no dup	4.68 ± 2.59		$\beta = 1$	2.25 ± 1.86
	no dup no inv	2.37 ± 1.76		$\beta = 2$	2.75 ± 0.47
				$\beta = 3$	2.15 ± 1.92

As discussed in the main paper, unlike TRE, COAT: 1) is more related to disentanglement since it focuses on translation equivariance instead of learning a composition function, 2) it utilizes hard negatives , and 3) it performs normalization.

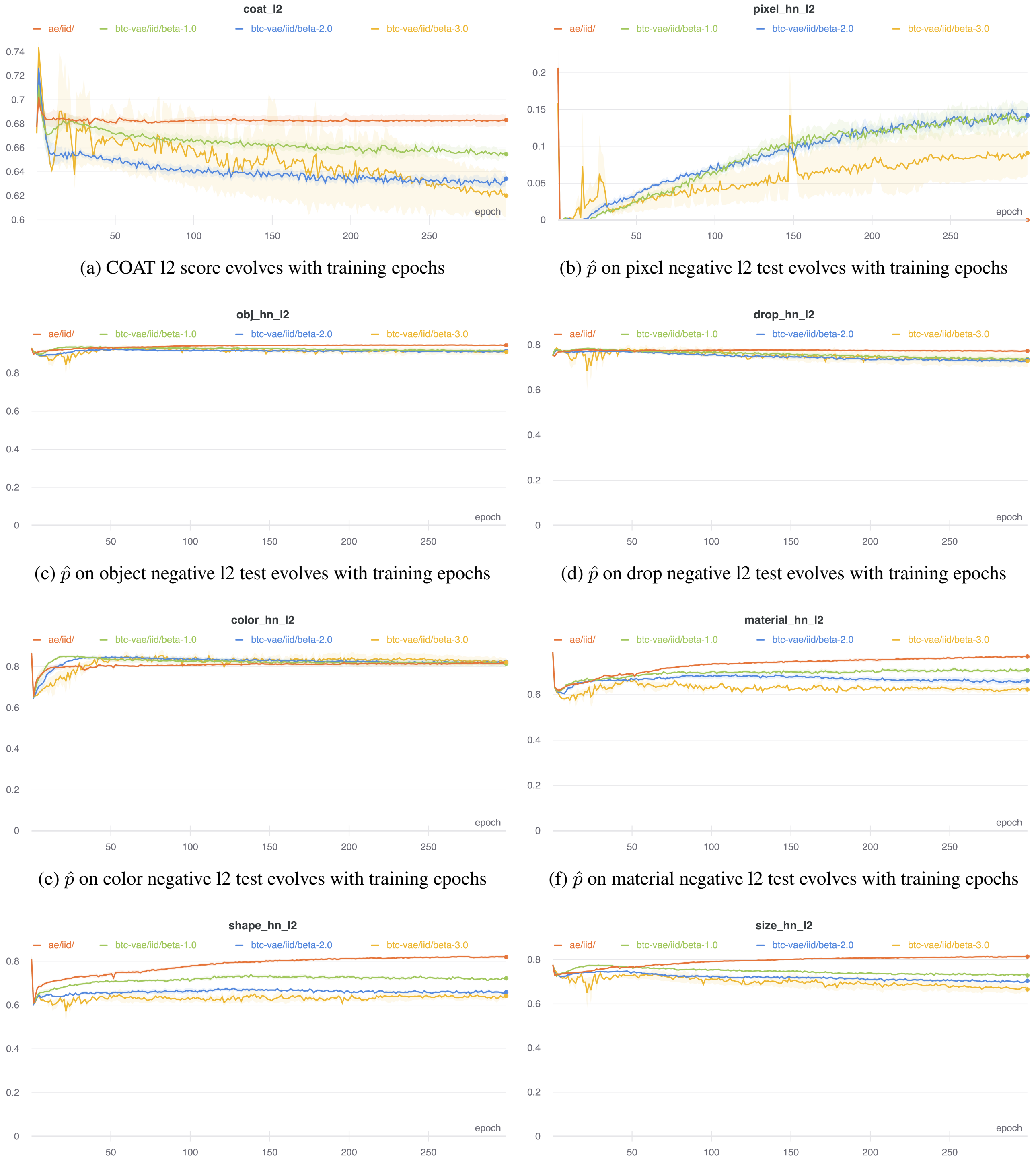
E. Greedy Matching Algorithm

When applying COAT metric to slot-based representations, a 4-way slot matching needs to be conducted to find the lowest matching cost. In this work, this is realized by iteratively picking one slot greedily from each representation z_A, z_B, z_C, z_D without replacement. And the criterion for this matching is the L2 residual $||z_B - z_A + z_C - z_D||_2$ from these slots. Empirically, we find this greedy matching algorithm perform very well even if it does not guarantee global optimum.

F. More Empirical Results

Here we provide the training curves of adjusted COAT metric and the sampled success rate $\hat{p}$ on hard negative tests.

F.1. Training Curves of Autoencoder and VAEs



(a) COAT l2 score evolves with training epochs

(b) $\hat{p}$ on pixel negative l2 test evolves with training epochs

(c) $\hat{p}$ on object negative l2 test evolves with training epochs

(d) $\hat{p}$ on drop negative l2 test evolves with training epochs

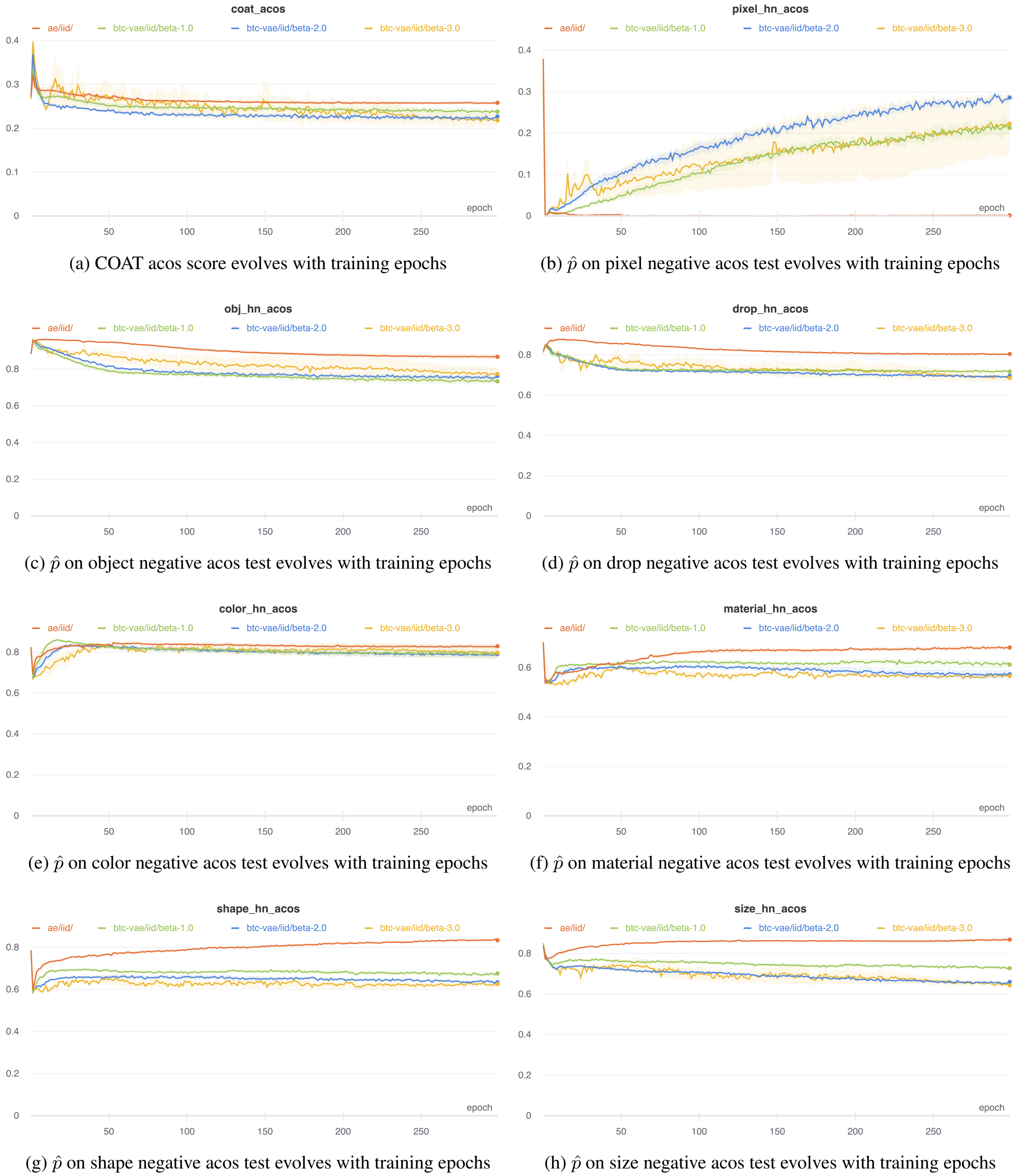
(e) $\hat{p}$ on color negative l2 test evolves with training epochs

(f) $\hat{p}$ on material negative l2 test evolves with training epochs

(g) $\hat{p}$ on shape negative l2 test evolves with training epochs

(h) $\hat{p}$ on size negative l2 test evolves with training epochs

Figure A5. COAT l2 for Autoencoder and VAEs



(a) COAT acos score evolves with training epochs

(b) $\hat{p}$ on pixel negative acos test evolves with training epochs

(c) $\hat{p}$ on object negative acos test evolves with training epochs

(d) $\hat{p}$ on drop negative acos test evolves with training epochs

(e) $\hat{p}$ on color negative acos test evolves with training epochs

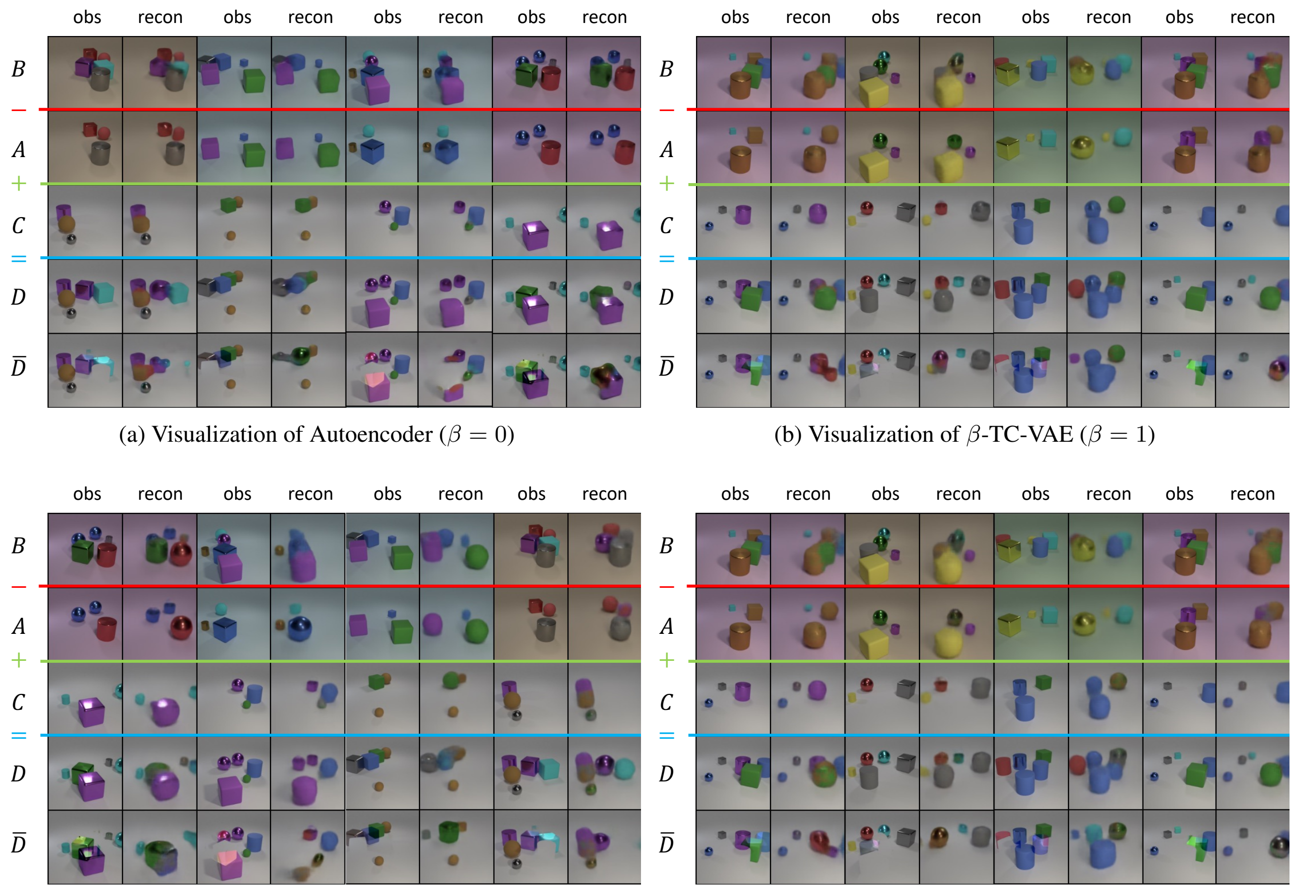
(f) $\hat{p}$ on material negative acos test evolves with training epochs

(g) $\hat{p}$ on shape negative acos test evolves with training epochs

(h) $\hat{p}$ on size negative acos test evolves with training epochs

Figure A6. COAT acos for Autoencoder and VAEs

F.2. Visualization of Autoencoder and VAEs



(a) Visualization of Autoencoder ($\beta = 0$)

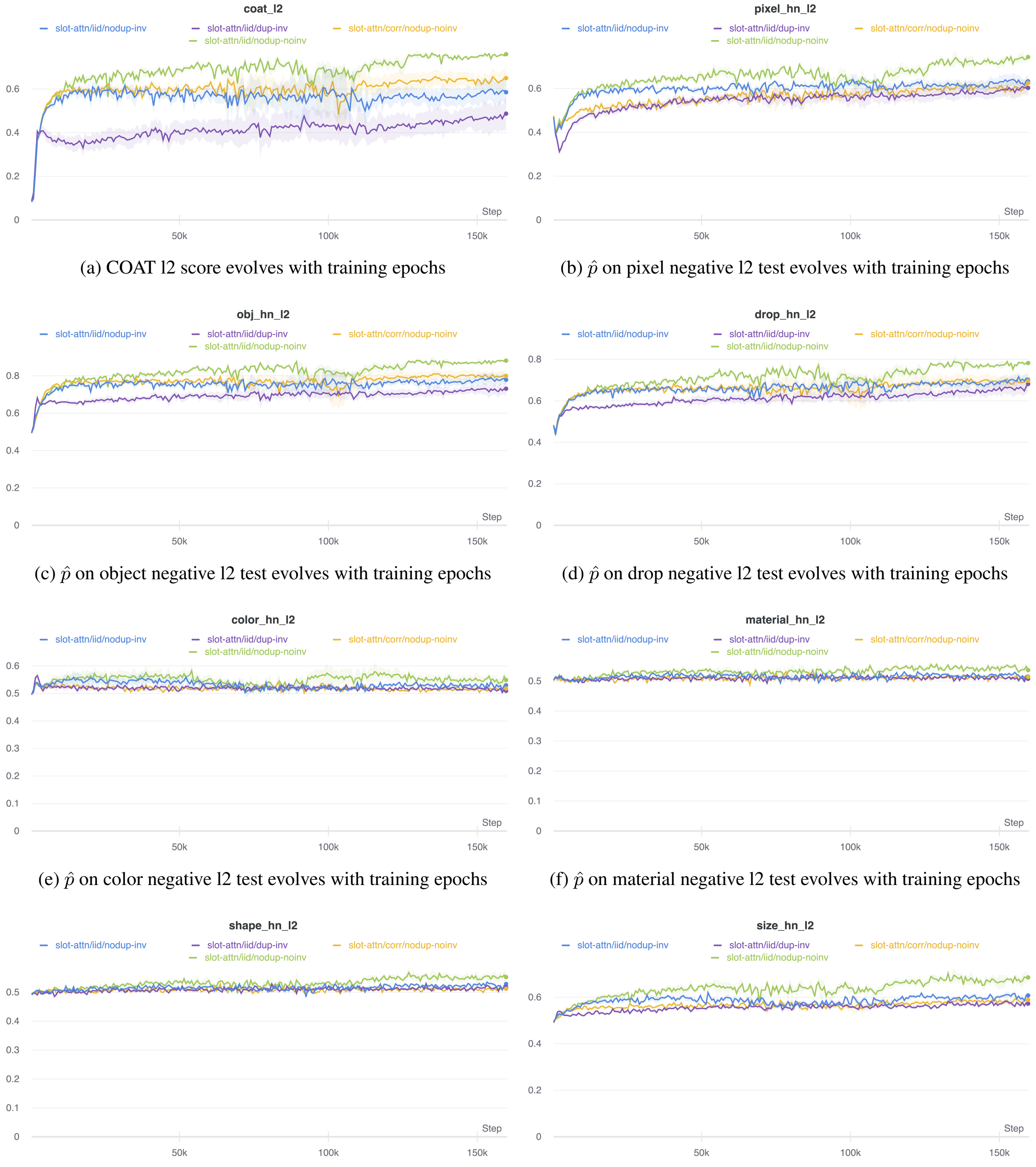
(b) Visualization of β-TC-VAE ($\beta = 1$)

(c) Visualization of β-TC-VAE ($\beta = 2$)

(d) Visualization of β-TC-VAE ($\beta = 3$)

Figure A7. Visulization of Autoencoder and VAEs. Odd columns are obserations A, B, C, D and pixel-level hard negative D'. Even columns are reconstructed A, B, C, D and decoded $\bar{D}$ from $z_B - z_A + z_C$. Interestingly, when $\beta = 0$, the decoded $z_B - z_A + z_C$ looks similar to the pixel-level hard negatives, while $\beta > 0$ gives more natural images in the decoded $z_B - z_A + z_C$.

F.3. Training Curves of Slot Attention



(a) COAT l2 score evolves with training epochs

(b) $\hat{p}$ on pixel negative l2 test evolves with training epochs

(c) $\hat{p}$ on object negative l2 test evolves with training epochs

(d) $\hat{p}$ on drop negative l2 test evolves with training epochs

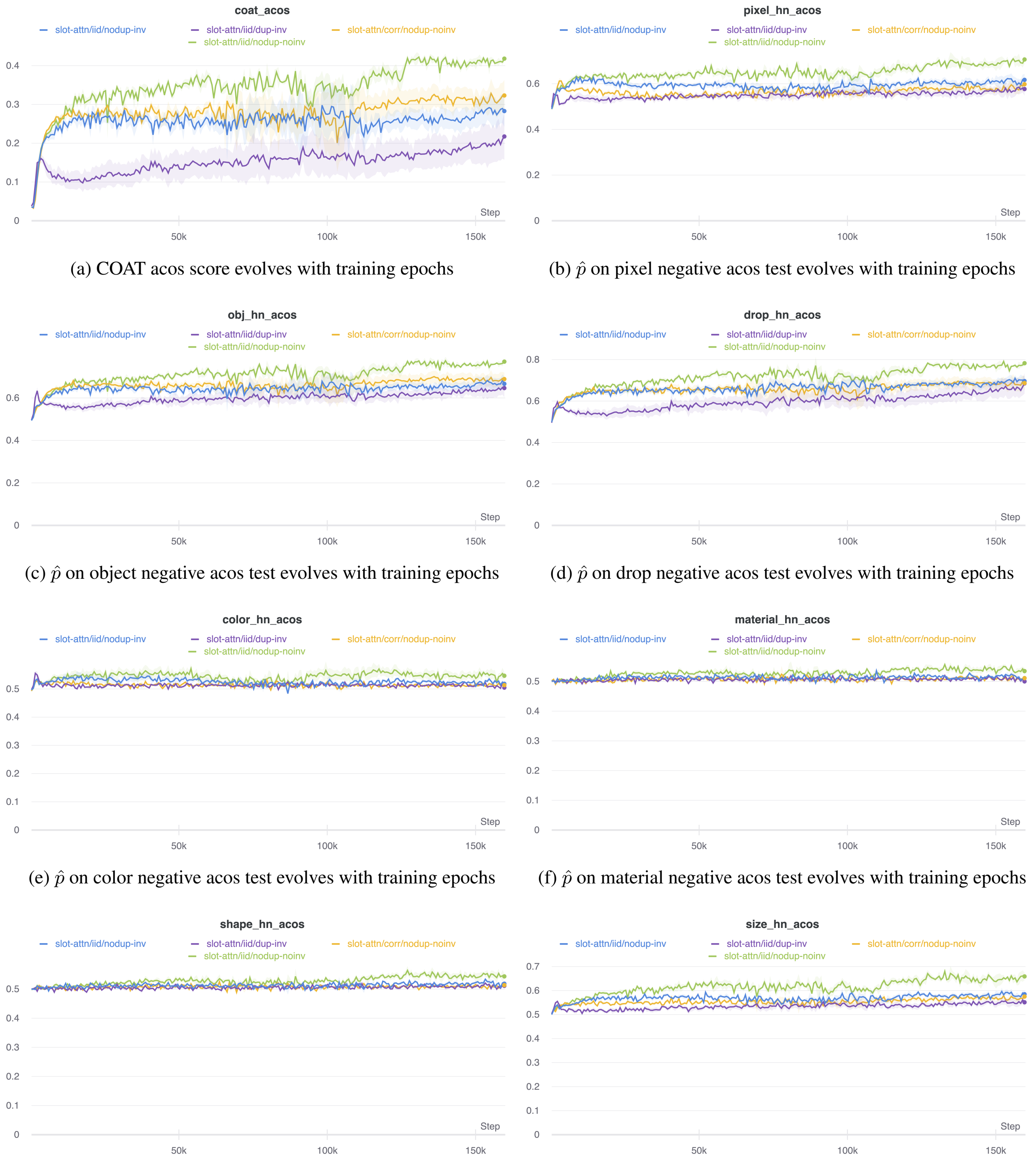
(e) $\hat{p}$ on color negative l2 test evolves with training epochs

(f) $\hat{p}$ on material negative l2 test evolves with training epochs

(g) $\hat{p}$ on shape negative l2 test evolves with training epochs

(h) $\hat{p}$ on size negative l2 test evolves with training epochs

Figure A8. COAT l2 for Slot Attentions



(a) COAT acos score evolves with training epochs

(b) $\hat{p}$ on pixel negative acos test evolves with training epochs

(c) $\hat{p}$ on object negative acos test evolves with training epochs

(d) $\hat{p}$ on drop negative acos test evolves with training epochs

(e) $\hat{p}$ on color negative acos test evolves with training epochs

(f) $\hat{p}$ on material negative acos test evolves with training epochs

(g) $\hat{p}$ on shape negative acos test evolves with training epochs

(h) $\hat{p}$ on size negative acos test evolves with training epochs

Figure A9. COAT acos for Slot Attentions

F.4. Imperfect Matching in Slot Attention without any Post-processing

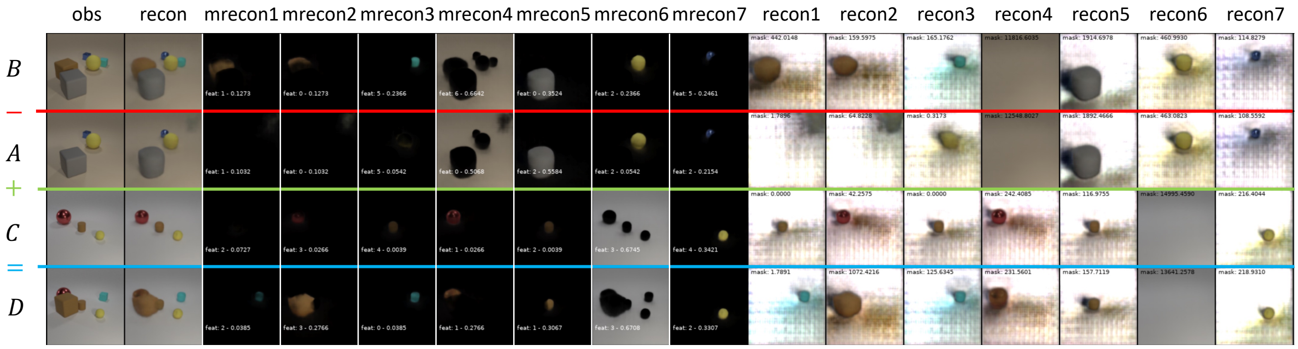


Figure A10. Visualization of Slot Attention's observation, combined reconstruction, masked reconstruction and reconstruction of each slot by columns and A, B, C, D by rows. These are after a greedy matching. White captions are index and the cosine similarity to the nearest slots. Black captions are the mask mass. The matching is generally good, and it can be observed that duplicated slots seem to be the bottleneck.

F.5. "Invisible Slots" in Slot Attention

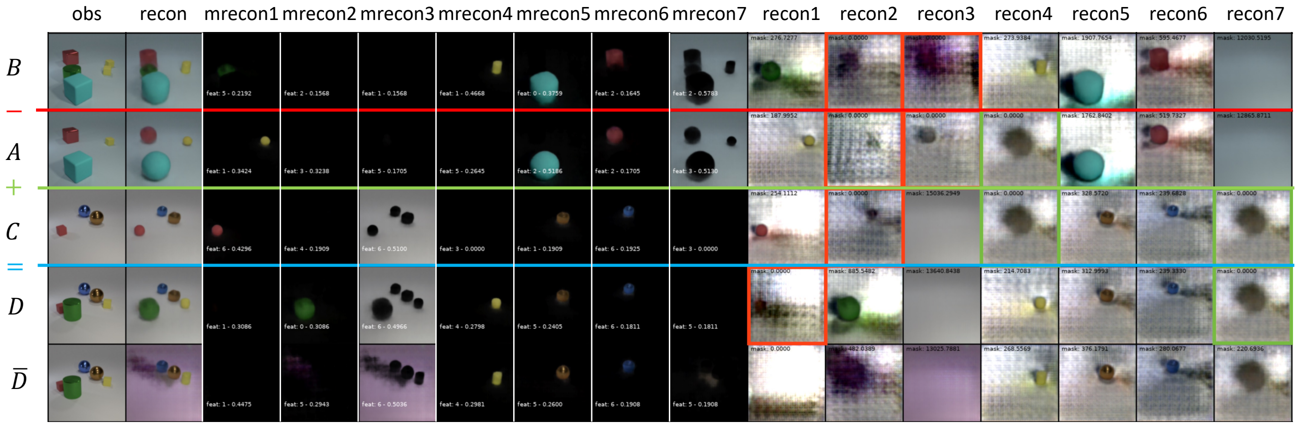


Figure A11. Visualization of Slot Attention's observation, combined reconstruction, masked reconstruction and reconstruction of each slot by columns and A, B, C, D, decoded $z_A - z_B + z_C$ by rows. These are greedy matching after removing duplicated. White captions are index and cosine similarity to the nearest slots. Black captions are the mask mass. Red boxes highlight "invisible slots", whose mask weights are zero, but apparently have different unmasked reconstructions. This inconsistency may cause the matching to fail. Green boxes highlight "pseudo blank slots", which are designed to be consistent.

F.6. Greedy Matching in Slot Attention after Removing Duplicate and Invisible Slots

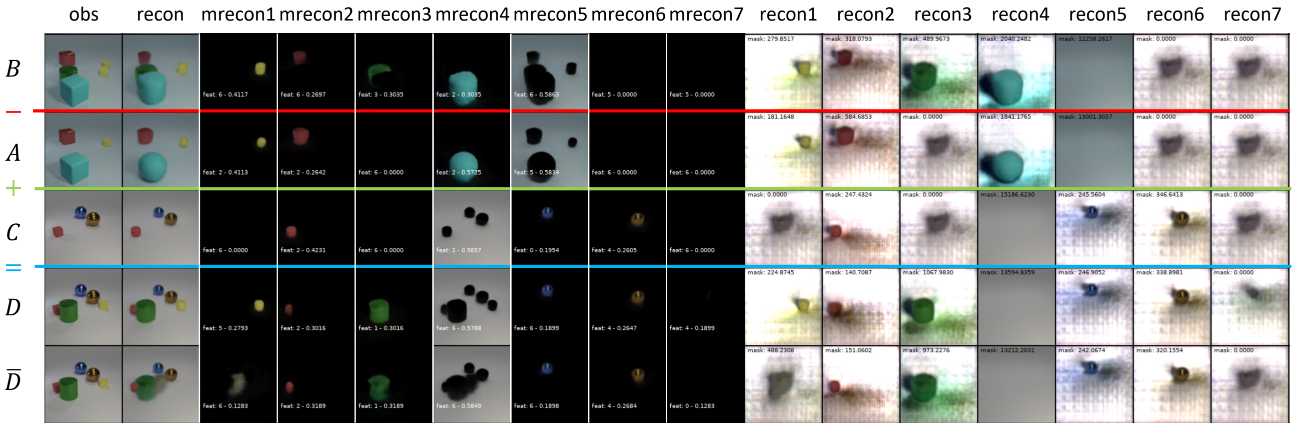
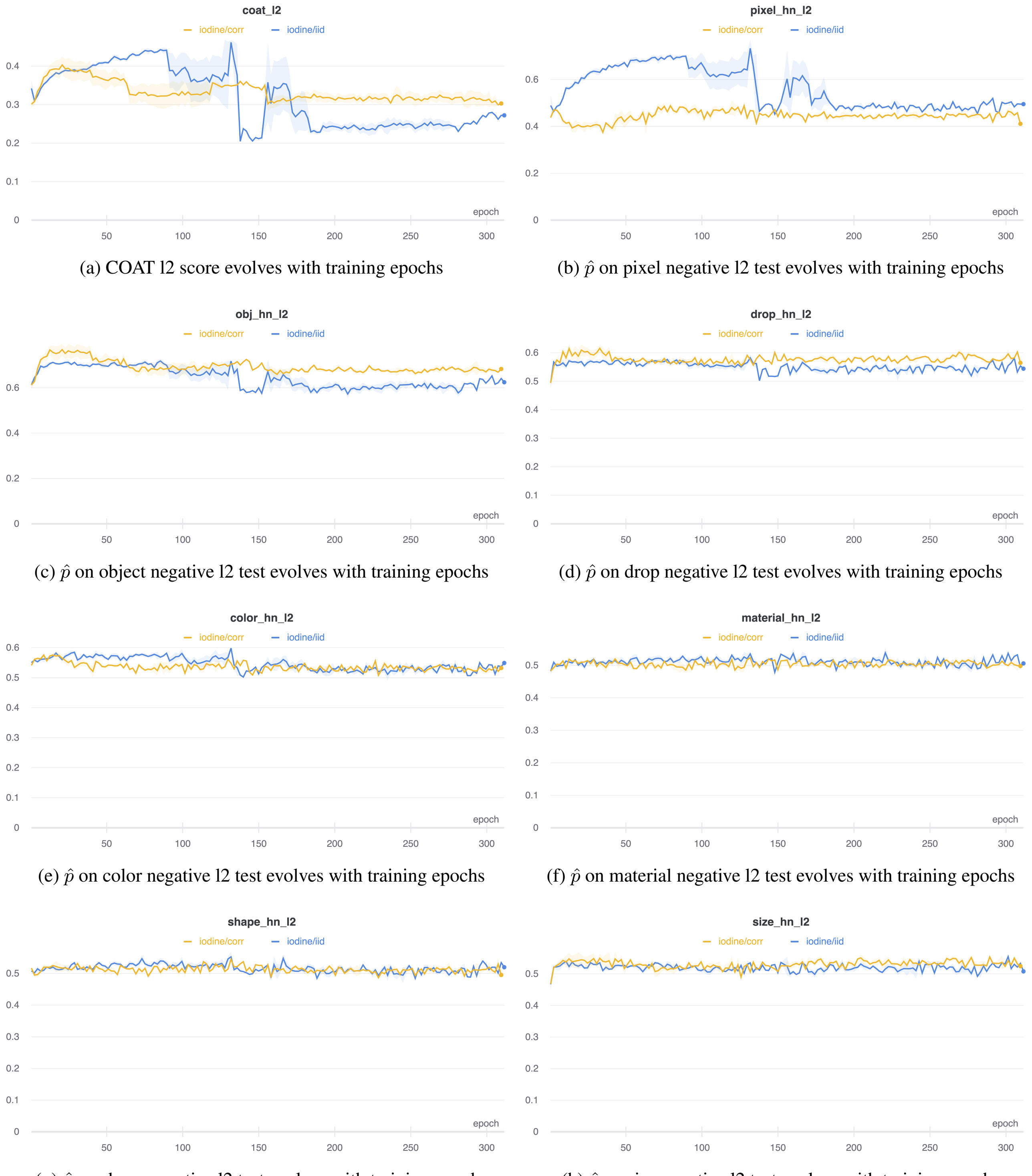


Figure A12. Visualization of Slot Attention's observation, combined reconstruction, masked reconstruction and reconstruction of each slot by columns and A, B, C, D, decoded $\mathbf{z}_A - \mathbf{z}_B + \mathbf{z}_C$ by rows. These are greedy matching after removing duplicated and invisible slots. White captions are index and cosine similarity to the nearest slots. Black captions are the mask mass. The matching is almost perfect, but we can still see the discrepency between 4th and 5th row due to (1) missing a yellow cube (2) uncertainty about the occluded green cylinder in A.

F.7. Training Curves of IODINE



(a) COAT l2 score evolves with training epochs

(b) $\hat{p}$ on pixel negative l2 test evolves with training epochs

(c) $\hat{p}$ on object negative l2 test evolves with training epochs

(d) $\hat{p}$ on drop negative l2 test evolves with training epochs

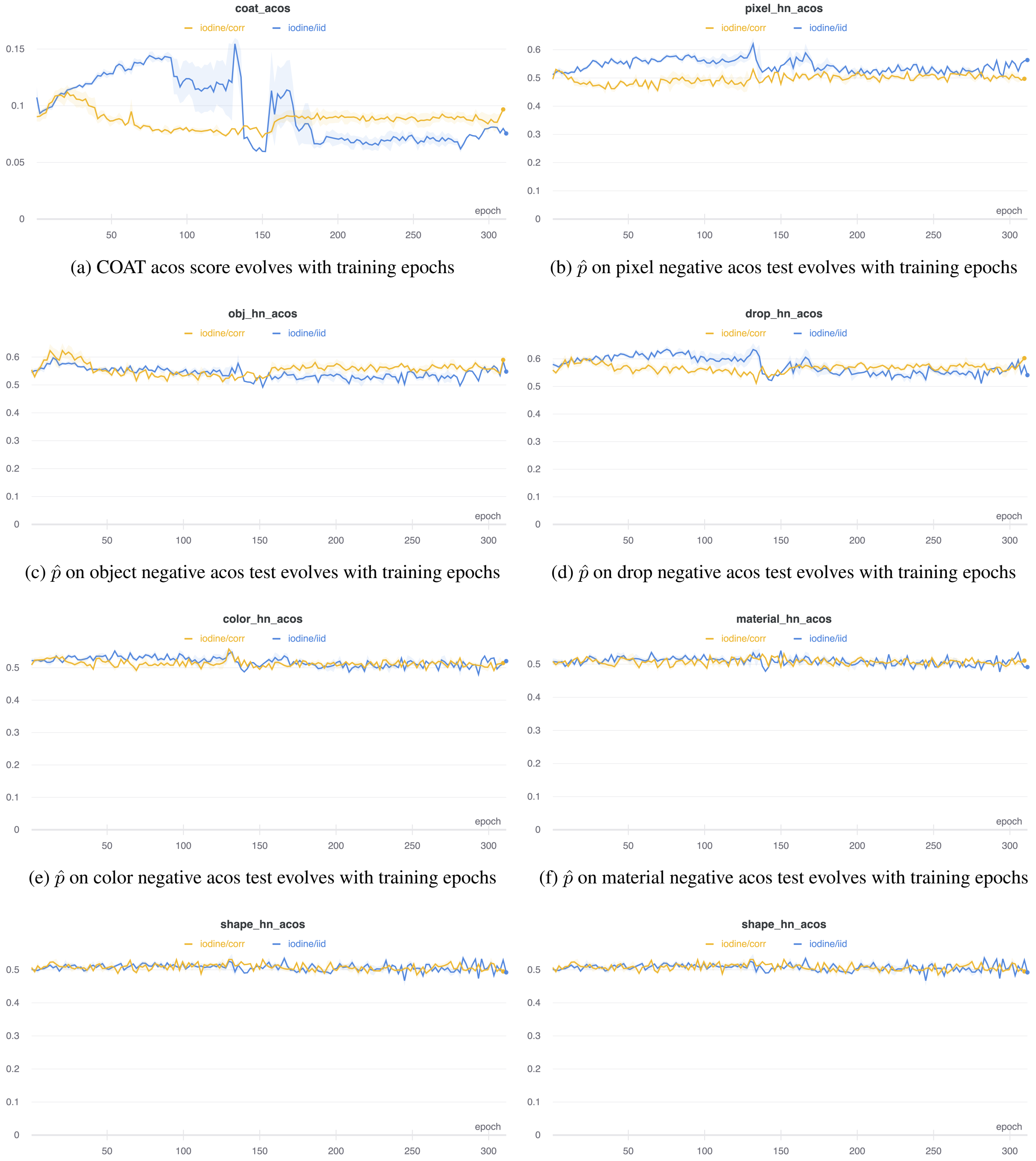
(e) $\hat{p}$ on color negative l2 test evolves with training epochs

(f) $\hat{p}$ on material negative l2 test evolves with training epochs

(g) $\hat{p}$ on shape negative l2 test evolves with training epochs

(h) $\hat{p}$ on size negative l2 test evolves with training epochs

Figure A13. COAT l2 for IODINE



(a) COAT acos score evolves with training epochs

(b) $\hat{p}$ on pixel negative acos test evolves with training epochs

(c) $\hat{p}$ on object negative acos test evolves with training epochs

(d) $\hat{p}$ on drop negative acos test evolves with training epochs

(e) $\hat{p}$ on color negative acos test evolves with training epochs

(f) $\hat{p}$ on material negative acos test evolves with training epochs

(g) $\hat{p}$ on shape negative acos test evolves with training epochs

(h) $\hat{p}$ on size negative acos test evolves with training epochs

Figure A14. COAT acos for IODINE

F.8. Visualization of IODINE

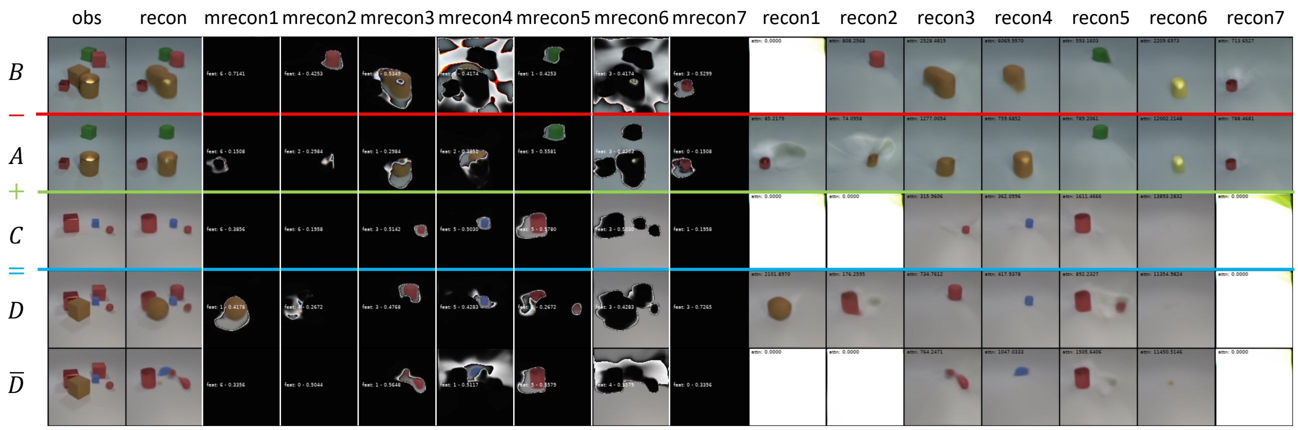
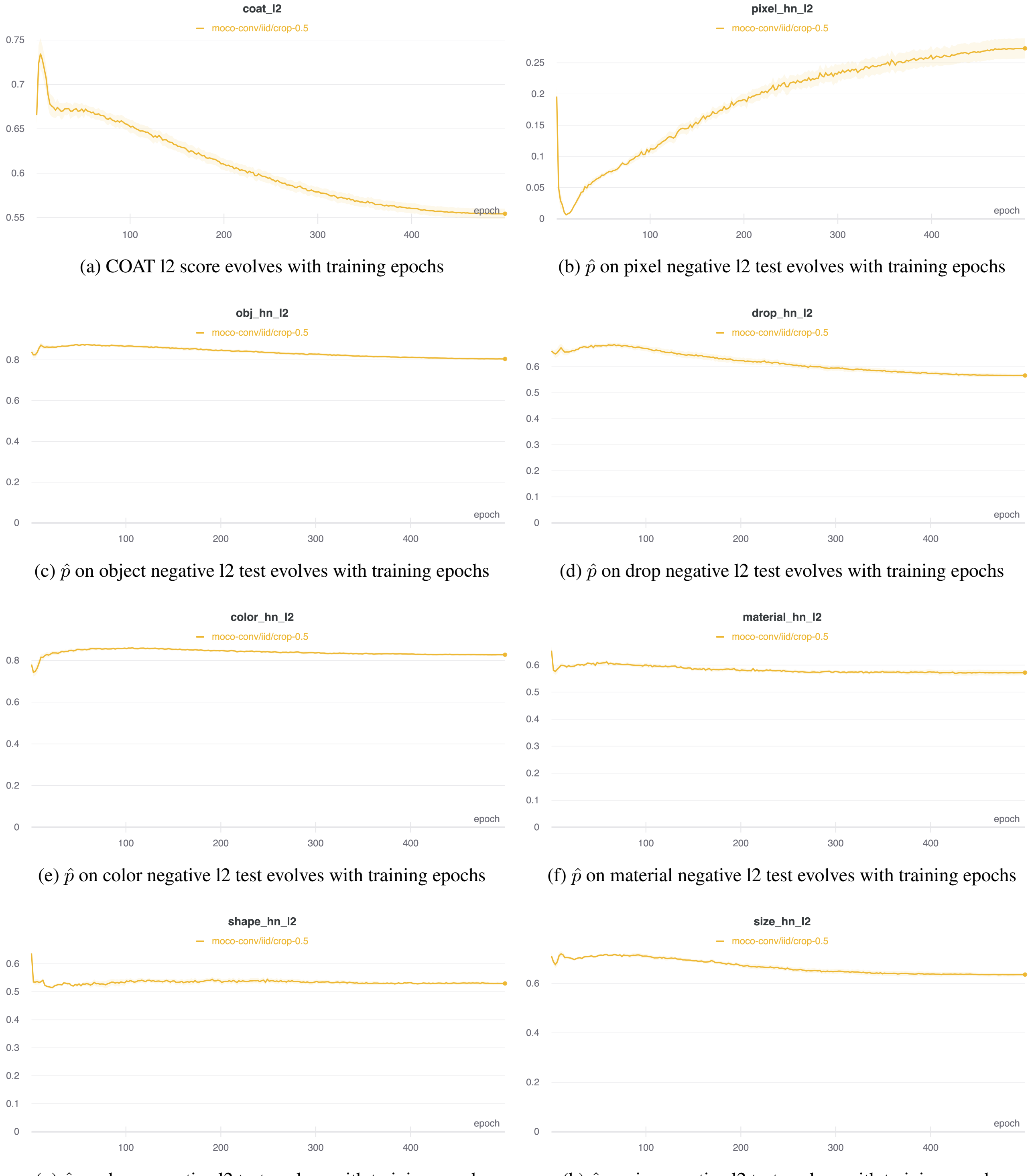


Figure A15. Visualization of IODINE's observation, combined reconstruction, masked reconstruction and reconstruction of each slot by columns and A, B, C, D, decoded $z_A - z_B + z_C$ by rows. The emergent "blank slots" are consistent – they are almost white in the unmasked reconstructions – so the matching is good in general. However, the objectness in slots is not consistent. It seems every slot has some background content in the masked reconstruction. Some occluded objects are not disentangled. Some objects are over-segmented into multiple slots and these slots cannot be detected with cosine similarity as duplicates. All these lead to the discrepency between D and decoded $z_A - z_B + z_C$, which may explain the unsatisfying performance of IODINE on our metric.

F.9. Training Curves of MoCo with ConvNet



(a) COAT l2 score evolves with training epochs

(b) $\hat{p}$ on pixel negative l2 test evolves with training epochs

(c) $\hat{p}$ on object negative l2 test evolves with training epochs

(d) $\hat{p}$ on drop negative l2 test evolves with training epochs

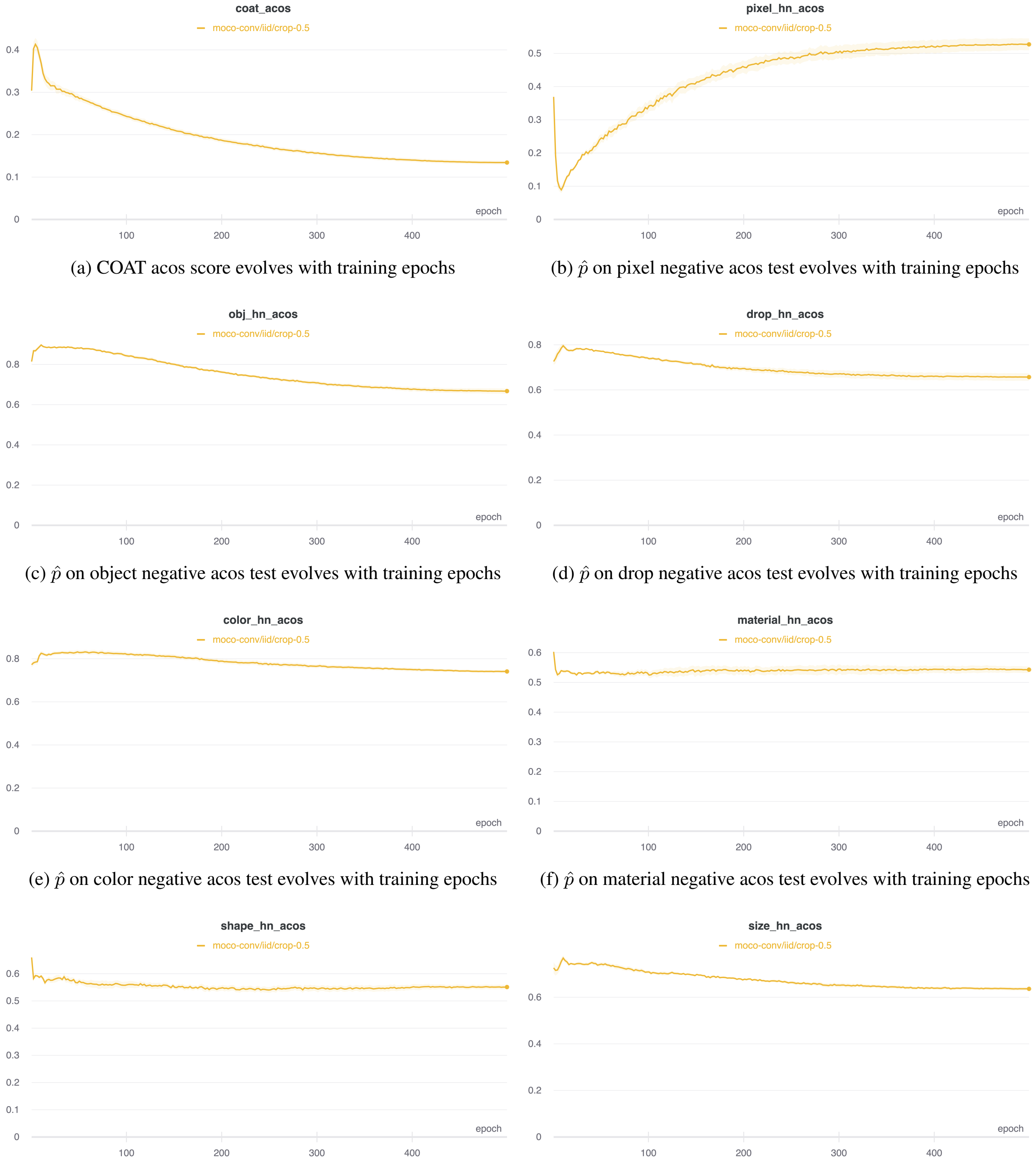
(e) $\hat{p}$ on color negative l2 test evolves with training epochs

(f) $\hat{p}$ on material negative l2 test evolves with training epochs

(g) $\hat{p}$ on shape negative l2 test evolves with training epochs

(h) $\hat{p}$ on size negative l2 test evolves with training epochs

Figure A16. COAT l2 for MoCo with ConvNet



(a) COAT acos score evolves with training epochs

(b) $\hat{p}$ on pixel negative acos test evolves with training epochs

(c) $\hat{p}$ on object negative acos test evolves with training epochs

(d) $\hat{p}$ on drop negative acos test evolves with training epochs

(e) $\hat{p}$ on color negative acos test evolves with training epochs

(f) $\hat{p}$ on material negative acos test evolves with training epochs

(g) $\hat{p}$ on shape negative acos test evolves with training epochs

(h) $\hat{p}$ on size negative acos test evolves with training epochs

Figure A17. COAT acos for MoCo with ConvNet