Latent Noise Segmentation: How Neural Noise Leads to the Emergence of Segmentation and Grouping

Ben Lonnqvist [*1] **Zhengqing Wu** [*1] **Michael H. Herzog** [1]

Abstract

Humans are able to segment images effortlessly without supervision using perceptual grouping. Here, we propose a counter-intuitive computational approach to solving unsupervised perceptual grouping and segmentation: that they arise *because* of neural noise, rather than in spite of it. We (1) mathematically demonstrate that under realistic assumptions, neural noise can be used to separate objects from each other; (2) that adding noise in a DNN enables the network to segment images even though it was never trained on any segmentation labels; and (3) that segmenting objects using noise results in segmentation performance that aligns with the perceptual grouping phenomena observed in humans, and is sample-efficient. We introduce the Good Gestalt (GG) datasets — six datasets designed to specifically test perceptual grouping, and show that our DNN models reproduce many important phenomena in human perception, such as illusory contours, closure, continuity, proximity, and occlusion. Finally, we (4) show that our model improves performance on our GG datasets compared to other tested unsupervised models by 24.9%. Together, our results suggest a novel unsupervised segmentation method requiring few assumptions, a new explanation for the formation of perceptual grouping, and a novel potential benefit of neural noise.

1. Introduction and Related Work

Humans perceptually group object parts in scenes (Wagemans et al., 2012a;b; Jäkel et al., 2016; Roelfsema, 2006), which allows them to interpret and segment elements in context, even when they deviate from stored templates (Geirhos et al., 2019; Herzog et al., 2015; Holcombe & Cavanagh, 2001). For example, the black and white stripes of a zebra are perceptually grouped together despite their vastly different colors. Historically, perceptual grouping in humans has been studied by the Gestaltists, who view the Gestalt principles of grouping (such as grouping by proximity and continuity) as crucial computational principles. This human capability to perform perceptual grouping is in stark contrast to object recognition in Deep Neural Networks (DNNs), which have been shown to struggle with both robustness (Geirhos et al., 2018; Madry et al., 2018; Moosavi-Dezfooli et al., 2016) and grouping (Bowers et al., 2022; Linsley et al., 2018; Doerig et al., 2020; Biscione & Bowers, 2023). Understanding the underlying mechanisms of perceptual grouping is critical for better modeling of human vision, and a more fundamental understanding of segmentation in general.

While incorporating explicit or implicit segmentation and grouping mechanisms in DNNs has seen substantial rise in interest in recent years (Greff et al., 2020; Hinton, 2023; Linsley et al., 2018; Sabour et al., 2017), mainstream instance segmentation approaches often either train models directly to segment using labeled samples (Ronneberger et al., 2015; Long et al., 2015; Kirillov et al., 2023), or inject a substantial number of biases into the model architecture (Hamilton et al., 2022; Engelcke et al., 2022; 2020a; Wang et al., 2022). The current prevailing approach is the implementation of object slots in the model architecture (Locatello et al., 2020), which maintains information about object instances in a separately maintained vector. Slot-based approaches are conceptually limited by the fact that the number of slots is fixed in the model architecture and cannot be adapted by learning, nor on an image-by-image basis. An approach that attempts to remedy the problems above was proposed by Löwe et al. (2022), who suggest a neuroscience-inspired solution to object learning: the object identities are stored in phase values of complex variables, which mimics the temporal synchrony hypothesis of brain function (Milner, 1974). While the approach can handle a variable number of objects, it is limited in the maximum number of objects it can represent (but see Stanić et al. (2023)). To date, there is no unified and simple princi-

[*]Equal contribution [1]École Polytechnique Fédérale de Lausanne (EPFL). Correspondence to: Ben Lonnqvist <ben.lonnqvist@epfl.ch>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

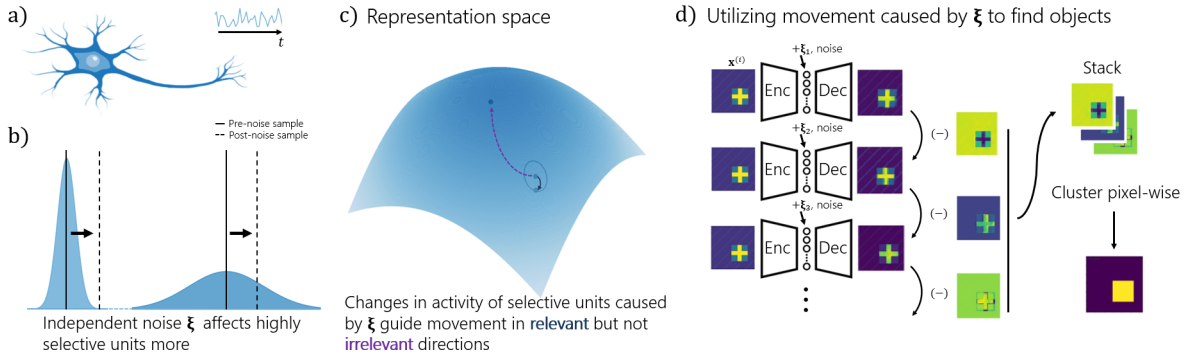


Figure 1: **Latent Noise Segmentation Schematic Illustration.** **(a)** Biological neurons are highly noisy. For example, thermal noise and ion channel shot noise (Manwani & Koch, 1998) contribute to independent noise in neurons. **(b)** Independent noise affects the output of neurons that are highly selective to a stimulus feature (*left*) than neurons that are less selective (*right*). Solid lines indicate the mean of the noise-free activity distribution, and dashed lines indicate the actual sample after independent noise is added. *The x-axis in the illustration is not meaningful.* **(c)** In the system's representational space (as indicated by the surface where input images are mapped to points on that surface), the changes caused by independent noise cause meaningful changes to the model's representation in relevant directions, e.g., local Principal Component (PC) directions, but not irrelevant directions (that would substantially change the model's representation of the input). **(d)** This yields information about the objects in the input image, and can be used to segment the input images. An input image $\mathbf{x}^{(i)}$ is fed to an autoencoder network, and noisy samples are drawn and consecutively subtracted from each other. These outputs contain information about the changes induced by noise in latent space, cast into image space. The outputs are stacked and clustered pixel-wise to generate a segmentation mask.

ple that allows DNNs to perform segmentation and grouping in a number of different, seemingly unrelated contexts with few *in-principle* limitations. In this work, we seek to remedy this gap.

Specifically, we develop a direct test benchmark of a network's segmentation and grouping capability by requiring it to segment images in a way that necessitates perceptual grouping —- furthermore, the network must do this without ever being trained on the stimuli in question, and without ever being trained on the task of segmentation, similarly to how the human visual system is not. To simultaneously accomplish all these goals, we posit a seemingly counterintuitive idea that we call Latent Noise Segmentation (LNS):

Neural noise enables deep neural networks to perceptually group and segment.

See Figure 1 for a schematic overview of the approach. In brief, we show that by injecting independent noise to a hidden layer of a DNN, we can turn DNNs trained on generic image reconstruction tasks to systems capable of perceptual grouping and segmentation.

In the following sections, we describe the general setting of LNS and its specific implementation in an autoencoder DNN (Section 2). We mathematically demonstrate that under certain (relatively loose) assumptions from basic optics, the presence of latent noise allows visual entities, which do not perfectly co-vary with each other, to be separated into individual objects (Section 2.1 and Appendix A.1). We develop a comprehensive test benchmark of segmentation tasks inspired by a century of Gestalt psychology and the study of grouping in humans (Wertheimer, 1923; Koffka, 1922) that we call the Good Gestalt (GG) datasets. We empirically show that LNS directly reproduces the result of the elusive Gestalt principles of grouping in segmentation masks (Section 2.3). Finally, we study how practically feasible LNS is (that is, how in-principle usable the method is by any system constrained by limitations like compute time or noise magnitude, such as the primate visual system) by evaluating how segmentation performance varies with different model learning rules, noise levels, and the number of time steps the model takes to segment (Section 3). Our results suggest that a practically feasible number of time steps (as few as a handful) are sufficient to reliably segment and that while encouraging a prior distribution in a model does not improve its segmentation performance, it stabilizes the optimal amount of noise needed for segmentation across all datasets and correctly identifies the appropriate number of objects.

2. Model and Methods

Basic setup. Our model architecture consists of two primary components. The backbone of the model is a pretrained (Variational) Autoencoder (VAE; Kramer (1991); Kingma & Welling (2014); Higgins et al. (2017)), where

an encoder $\mathrm{Enc}(\mathbf{x}) = q_\phi(\mathbf{z}|\mathbf{x})$ learns a compressed latent distribution, and a decoder $\mathrm{Dec}(\mathbf{z}) = p_\theta(\mathbf{x}|\mathbf{z})$ models the data that generated the latent representation $\mathbf{z}$. In practice, the model is optimized using the Evidence Lower Bound (ELBO):

$$ELBO = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - \beta D_{KL}(q_\phi(\mathbf{z}|\mathbf{x})||p(\mathbf{z})), \tag{1}$$

where D_{KL} is the Kullback-Leibler divergence, and $p(\mathbf{z})$ is the prior set to $\mathcal{N}(0, \mathbf{I})$. β is a configurable coefficient facilitating reconstruction quality, which we set automatically through the GECO loss (Rezende & Viola, 2018). For a discussion about alternate architectures, see Appendix A.1.5.

Algorithm 1 Latent Noise Segmentation.

Require: image $\mathbf{x}^i$, time steps N, small noise variance σ^2_{small}
Require: trained (β)-VAE with $\mathrm{Enc}_\phi, \mathrm{Dec}_\theta$
Procedure SEGMENT($\mathbf{x}^i, N$)
$\boldsymbol{\mu}(\mathbf{x}^i) \leftarrow \mathrm{Enc}(\mathbf{x}^i)$
for $n = 1, \ldots, N$ **do**
  $\boldsymbol{\xi}_n \leftarrow \mathcal{N}(0, \sigma^2_{small})$
  $\tilde{\mathbf{x}}^i_n \leftarrow \mathrm{Dec}\left(\boldsymbol{\mu}\left(\mathbf{x}^i\right) + \boldsymbol{\xi}_n\right)$
  if $n \geq 2$ **then**
    $\Delta\tilde{\mathbf{x}}^i_{n-1} \leftarrow \tilde{\mathbf{x}}^i_n - \tilde{\mathbf{x}}^i_{n-1}$
  end if
end for
Let $\Delta\tilde{\mathbf{X}}^i \equiv [\Delta\tilde{\mathbf{x}}^i_1, \Delta\tilde{\mathbf{x}}^i_2, \ldots, \Delta\tilde{\mathbf{x}}^i_{N-1}]$ $\{\Delta\tilde{\mathbf{X}}^i$ has dimension $img_x \times img_y \times c \times (N-1)\}$
Cluster $\Delta\tilde{\mathbf{X}}^i$
Assign segmentation mask values to pixels according to clustering
End Procedure

2.1. Latent Noise Segmentation

After training the model backbone, we build a noisy segmentation process which enables the extraction and combination of information from the model's latent space to generate segmentation masks. To do this, we add i.i.d. noise $\boldsymbol{\xi}_n$ to the latent variables $\mathbf{z}$. The process is repeated N times, allowing us to obtain information about which objects exist in the model outputs. For every time step, two consecutive model outputs are subtracted from each other to find what in the output images $\tilde{\mathbf{x}}$ *changed with respect to changes in latent space* $\mathbf{z}$. At the end of the process, the stack of subtracted outputs is clustered using Agglomerative Clustering (Pedregosa et al., 2011), resulting in a segmentation mask. Full details of the algorithm are shown in Algorithm 1, a schematic is shown in Figure 1, and additional details can be found in Appendix E.2.

2.2. The Role of Independent Noise

Latent Noise Segmentation works due to several key reasons. Independent noise $\boldsymbol{\xi}$ induces the locally monotone mapping $f : \Delta\mathbf{z} \rightarrow \Delta\tilde{\mathbf{x}}$ from the latent variables to the output learned by the model. A potential problem is that this mapping happens for all units — even those that are not meaningful for the task of segmentation — and as such, they might adversely affect segmentation performance. To solve this, one could identify the meaningful units by perturbing them individually (like in latent traversal (Kingma & Welling, 2014; Higgins et al., 2017), but not only is this computationally expensive and impractical, it is also biologically implausible.

The solution to the problem comes from the intuition that units that meaningfully code for relevant stimulus features have a substantially higher derivative $\Delta\tilde{\mathbf{x}} = \Delta\mathbf{z} * \frac{\delta\tilde{\mathbf{x}}}{\delta\mathbf{z}}$ in the neighborhood of the stimulus $\mathbf{x}$. This is because for the decoder to achieve its goal of low reconstruction loss, it should be sensitive to small changes in the activity of units that are coding for relevant features in $\mathbf{x}$, but not for irrelevant features. This means that we can perturb all units simultaneously with independent noise: training on a generic reconstruction loss encourages the network to code in a manner that allows independent noise to reveal the relevant derivative direction, making the representation clusterable for segmentation (Figure 1b). Indeed, the goal of independent noise is not to perform segmentation itself, but to reveal the local neighborhood of the input stimulus and thus enable segmentation, *from the perspective of the decoder*. We provide a more complete mathematical intuition of exactly why noise perturbation on latent units causes a meaningful object segmentation in the case of the VAE in Appendix A.1. In summary, we observe that noise induces the output pixels belonging to the same part of the image to co-vary with each other, while pixels belonging to different parts do not. Following a principal component argument (Zietlow et al., 2021; Rolínek et al., 2019), we show that the derivatives of the decoder's output with respect to the latent units are the local principal component directions of the training data. Together, these phenomena enable meaningful object segmentation.

2.3. The Good Gestalt (GG) Datasets

To evaluate our models, we developed the six datasets that we together call the Good Gestalt (GG) datasets. The notion of Gestalt comes from the study of perception (Wertheimer, 1923; Jäkel et al., 2016; Wagemans et al., 2012a;b), and is perhaps best summarized as 'the whole is greater than the sum of its parts'. The GG datasets aim to address a large variety of different rules of perceptual organization, otherwise known as Gestalt principles of grouping like continuity, closure and proximity — here, we out-

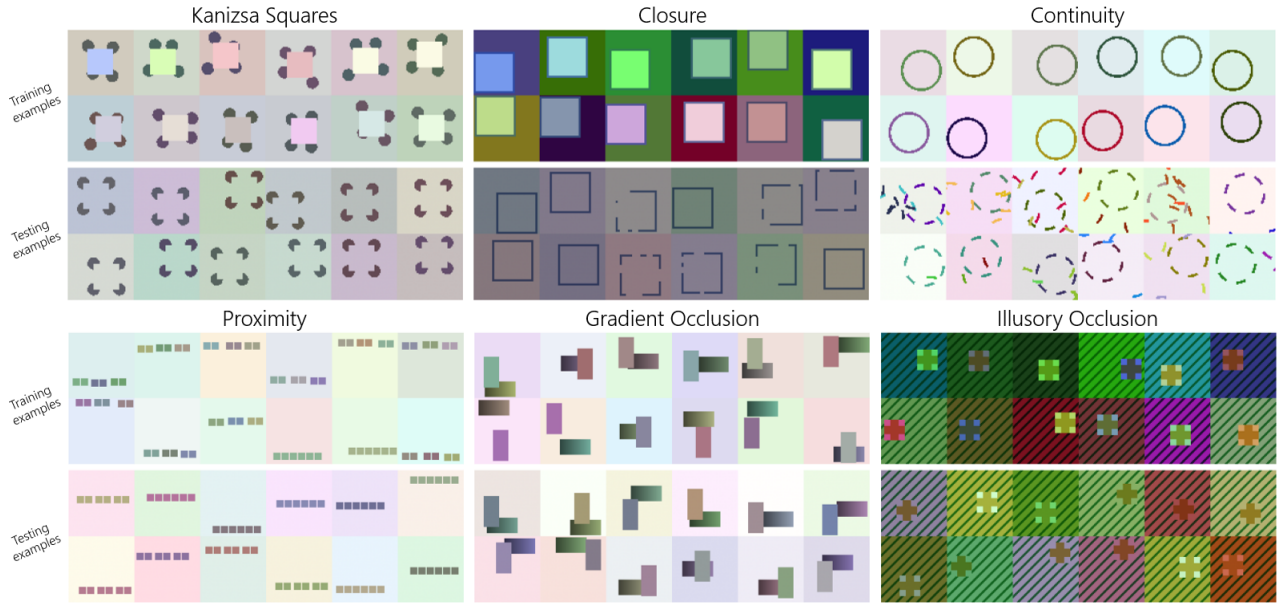


Figure 2: **The Good Gestalt (GG) Datasets.** Zoom in for an ideal viewing experience. The first two rows of images show training image examples, while the second two rows show images of testing examples.

line the reasons for studying machine perception using such datasets, and then describe the specifics of our datasets. Example stimuli of the training and testing samples are shown in Figure 2.

Reasons for studying Gestalt. In almost all the Gestalt stimuli we study in this work, we assume that in training the luminances (pixel values) of different objects in a visual scene are independent, while the luminances (pixel values) of the same object co-vary. This assumption follows a simplified view on optics (for details, see Appendix E.1), and is important as some principles of perceptual grouping have been shown to be consistent with the statistical structure of the natural environment (Elder & Goldberg, 2002; Geisler et al., 2001; Sigman et al., 2001). In testing, the model should generalize this information in ambiguous contexts, where the pixel values of the images are not fully informative of the object. For example, in the Kanizsa Squares dataset (Figure 2, *Kanizsa Squares, Testing examples*), the pixel value of the background is the same as the pixel value of the illusory square in the middle of the cornering circles.

We show target segmentation masks for the GG datasets in Figure 3. To determine what these segmentation masks should be, our focus here is not about *what* specifically humans perceive in a specific psychophysical experiment, but rather about what humans *can* perceive given the right instruction. This is part of what makes studying Gestalt so difficult: simple phrasing about what a participant should segment in such an image can potentially crucially affect how they segment the image. This is why in our GG datasets, we did not collect human data, but instead focus on the generic question of "which elements in the image belong together?", and defined the target segmentation masks for the test set evaluation on the basis of this question, focusing on evidence from human research in general from the past century (Wertheimer, 1923; Kimchi, 1992).

In total, the GG datasets are made up of six individual datasets, for which we explain the main premise individually below. More information on the datasets can be found in Appendix E.1.

Kanizsa Squares to study the perception of illusory figures. Humans are able to perceive illusory contours or illusory objects when cued by, for example, Pacman-like circles (Wang et al., 2012; Lesher, 1995).

Closure to study whether a model can combine individual image elements to form an object. The human ability to group objects together when they form a closed figure is well-documented (Elder & Zucker, 1993; Pomerantz et al., 1977; Ringach & Shapley, 1996; Marini & Marzi, 2016).

Continuity to study the integration of individual elements across the visual field. When elements are oriented such that they would form a continuous object, humans are able to group the elements together to form a whole figure (Kwon et al., 2016; Kovács & Julesz, 1993; Field et al., 1993).

Proximity to study the cue of intra-element distance for grouping. When multiple elements in a figure are

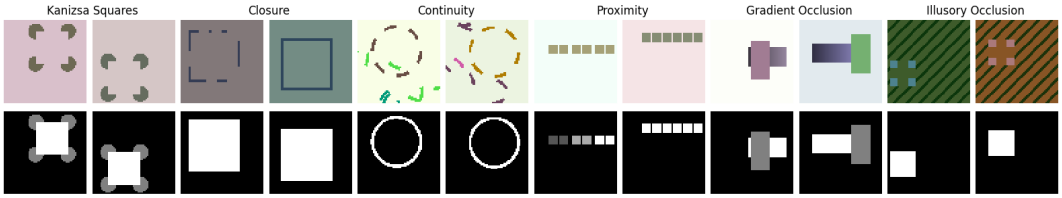


Figure 3: **Target segmentation mask examples of the GG datasets.** The first row of images shows training image examples, while the second row shows images of testing examples. Different colors indicate different objects. **Kanizsa Squares**: The model should segment a square in the center, and four background circles separately from the background. **Closure**: The model should segment a square. **Continuity**: The model should segment a circle traced by the relevant line segments to "complete the circle". **Proximity**: The model should segment a set of six squares together, or three sets of two squares when the proximity cue is given. **Gradient Occlusion**: The model should segment the two rectangles separately. **Illusory Occlusion**: The model should segment the static background and stripes together, and the foreground object parts together.

similar, humans can perceive many similar elements as belonging together with those that are closer in proximity (Kubovy & van den Berg, 2008; Quinlan & Wilton, 1998).

Gradient Occlusion to study the ability to handle full or partial occlusion of an element without a constant color value. The basis of objecthood in humans does not depend crucially on the exact color of the elements, and long-range integration and inference about objects and how they change under occlusion is possible (Kellman & Spelke, 1983; Spelke, 1990).

Illusory Occlusion to study grouping across the visual field while being occluded by an illusory object. Regular textures can be grouped together, and foreground objects that break the pattern can be separated from the complex textural background. A classic example is the Dalmatian Dog illusion by R. C. James, of which our Illusory Occlusion dataset is a simplification.

Together, the GG datasets comprise tests of a substantial number of the most major Gestalt laws of perceptual organization (for reviews, see Wagemans et al. (2012a;b); Jäkel et al. (2016); Todorovic (2008); Kanizsa (1979)), including figure-ground segregation, proximity, continuity, closure, and the general notion of Good Gestalt (the tendency of the perceptual system to organize objects according to the aforementioned laws).

2.4. Evaluation metrics and baseline models

Metrics. We evaluate all models using the Adjusted Rand Index (ARI) metric (Hubert & Arabie, 1985), which measures the above-chance similarity between two clusterings: the target segmentation mask, and the predicted segmentation mask. Here we choose to judge our models based on the ARI metric (instead of ARI minus background) because the point of the GG datasets (and of the study of Gestalt in general) is that of object-background separation, not of fine-grained object-specific details. We verified that in the GG datasets, an all-background assignment yields an ARI of ≈ 0, i.e. no better than random assignment. In addition, we plot randomly drawn examples of segmentation masks in Figure 4, and all segmentation masks for our highest-performing models in Appendix F.4, Figures 16 - 27.

Models and control experiments. We trained 5 Autoencoder (AE) and Variational Autoencoder (VAE) models with MSE and GECO (Rezende & Viola, 2018) losses, respectively. In addition, we computed two control experiments, one for each model variant (Noisy AE/VAE Reconstruction) with the goal of verifying whether adding noise at the latent layer is important. These control experiments compute a clustering on the AE/VAE reconstruction pixel values, adding noise to the reconstruction instead of the latent layer, keeping the rest of the model architecture and methodology the same (for more details, see Appendices E.2 and 14). We also evaluate two baseline models, Genesis (Engelcke et al., 2020a) and Genesis-v2 (Engelcke et al., 2022), for which details are shown in Appendix C.

3. Experimental Results

3.1. Quantitative Evaluation

Both the AE and VAE models achieve good quantitative performance in general perhaps with the exception of the Continuity dataset (Table 1). Adding noise in latent space as opposed to image space unlocks a substantial improvement in segmentation performance, improving ARI by 0.506 on average across all GG datasets. As described in Section 3.2, we do not see the somewhat lower quantitative score as a failure *per se*, but rather as related to the way in which the models fundamentally segment using LNS. Since the segmentation process seeks to vary the

Model	Kanizsa	Closure	Continuity	Proximity	Gradient Occ.	Illusory Occ.	Mean
AE	0.859 ± 0.008	0.766 ± 0.008	0.552 ± 0.008	**0.996*** ± 0.001	0.926 ± 0.009	**0.994*** ± 0.002	**0.849**
VAE	**0.871** ± 0.005	**0.795** ± 0.009	0.593 ± 0.012	0.943 ± 0.010	0.918 ± 0.002	0.974 ± 0.003	**0.849**
AE Rec.	0.246 ± 0.006	0.064 ± 0.001	0.570 ± 0.006	0.141 ± 0.001	**0.982** ± 0.000	-0.023 ± 0.001	0.333
VAE Rec.	0.343 ± 0.004	0.084 ± 0.000	**0.611** ± 0.005	0.144 ± 0.001	0.977 ± 0.001	-0.024 ± 0.001	0.356
Genesis	0.346	0.755	0.394	0.879	0.933	0.294	0.600
Genesis-v2	0.415	0.000	0.399	0.059	0.000	0.422	0.216

Table 1: Model results (ARI ± standard error of the mean). AE (Autoencoder) and VAE (Variational Autoencoder) model scores reported for the best noise level and number of samples. AE Rec. and VAE Rec. stand for noisy reconstruction control experiments, and (see Section 2.4 for details). Bold indicates top-performing model. For the AE and the VAE, a star (*) indicates a statistically significant difference in model performance (Bonferroni corrected). Our method improves performance over the previously published top-performing model Engelcke et al. (2020a) by 0.249%.

latent representation of the models, it introduces artifacts related to the model's representational space, such as small shifts of the stimulus in a given direction, or small changes to its size. The ARI metric is punishing to even single-pixel shifts of the target object — qualitatively, it is clear that even in cases where the model does not achieve an excellent ARI score, it captures the desired effect. This is in contrast to the Genesis and Genesis-v2 models, which fail at capturing the desired effect even when they do well quantitatively (Figure C). A similar effect of minor performance deterioration with respect to the baseline of clustering a reconstruction can be observed in the Gradient Occlusion dataset, where good reconstruction quality of the model is congruent with a good segmentation performance under output clustering. This is to be expected - the dataset by design is such that the pixel values are mostly informative of the correct segmentation label identities.

To understand whether a structured latent prior could improve model segmentation capability under LNS, we conducted a two-sample two-tailed t-test to compare the AE and VAE ($\beta > 1$) performance. The test indicated that the AE and VAE performance statistically significantly differed only for the Proximity ($t = 4.54$, $p < 0.05$) and Illusory Occlusion datasets ($t = 5.22$, $p < 0.05$) after the Bonferroni correction, while no statistically significant difference was found for the other datasets. Due to the additional loss term that places tension on the quality of the reconstruction, it is not surprising that the VAE would show small decreases in performance — indeed, we conclude that encouraging a structured latent prior on the model does not in general improve the model's capability to segment stimuli.

3.2. Qualitative Evaluation

To understand how the models perform qualitatively, we plot a random selection of examples in Figure 4, and all model outputs of the shown seed for all datasets in Appendix F.4. Important are two things: first, while model outputs are consistently good and generally reflect appropriate grouping of the correct object elements in the correct groups, segmenting by noise causes noisy artifacts in some cases (e.g. Figure 4: *Continuity*, *Gradient Occlusion*). Second, a qualitative inspection reveals how quantitative metrics fail to appropriately judge the goodness of model performance. Since segmentation is done by inference over a noisy set of outputs, and the noise affects the representation of those outputs, the outputs in some datasets (like *Continuity* and *Closure*) are often shifted or misshaped by a small number of pixels, resulting in a disproportionately large drop in quantitative performance, while qualitative performance remains relatively unchanged.

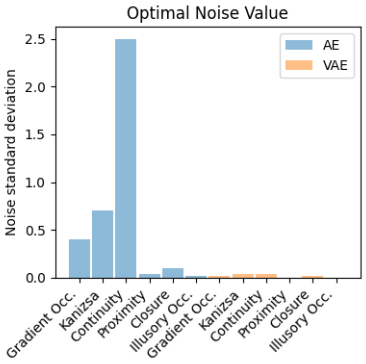


Figure 5: **The optimal noise level for the AE and the VAE across datasets.** The AE performs best with a larger range of noise values, while the VAE consistently prefers low noise values.

To further understand how elements are grouped in our model, we conducted two additional qualitative analyses: (1) Using UMAP visualization, we inspected model clustering and found that while the AE does not find the correct object binding, the VAE does (Appendix F.1). (2) To understand how well the segmentation method generalizes, we qualitatively evaluated the models on the CelebA dataset (Liu et al., 2015) and found that while the AE generally

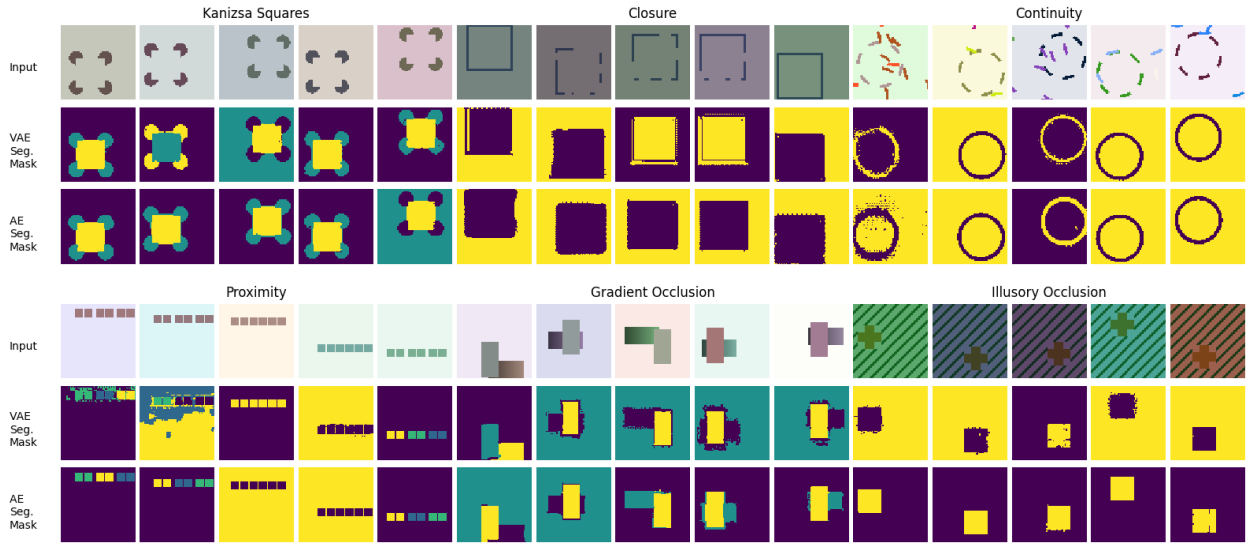


Figure 4: **Model Output Segmentation Mask Examples.** The first row shows inputs, the second row shows VAE segmentation masks, and the third row shows AE segmentation masks. Randomly selected examples from the **Kanizsa Squares**, **Closure**, **Continuity**, **Proximity**, **Gradient Occlusion**, and **Illusory Occlusion** datasets are shown using the best model hyperparameters shown in Table 1. Since the model segments by clustering noisy outputs, the specific color assignment to different object identities is arbitrary (for example, whether the model assigns the identity represented by yellow as the background, or purple, is not a meaningful distinction).

fails to find a semantically meaningful segmentation mask (and instead opts for more basic image-level features), the VAE often finds a semantically meaningful segmentation of face-hair-background (Appendix B).

3.3. Noise sensitivity

To understand how practically feasible LNS is for a system constrained by compute time or noise magnitude, we analyzed how the amount of noise added in latent space would affect the model's segmentation performance. While enforcing a prior distribution on the model's latent space (as is the case in the VAE) had no benefit in model performance (and perhaps a slight deterioration), it may have had a minor benefit in the across-task best-performing noise consistency. We tested the same models across different levels of noise used for segmentation and found that enforcing the prior $\mathcal{N}(0, \mathbf{I})$ caused the model to consistently perform best at very small levels of noise, while the AE optimal level of noise varied across datasets (Figure 5). This is intuitively not surprising, since the VAE enforces a specific activity magnitude for uninformative units in its prior ($\mathcal{N}(0, \mathbf{I})$), while the AE does not — as such, the AE has no reason to prefer a specific coding magnitude in any given task. To quantify the result, we performed an F-test for the equality of variances, and found that the level of noise that yielded the best results had higher variance for the AE than the VAE ($F = 2994.56, p = 1.11e^{-8}$).

The result suggests that enforcing a prior distribution on the representation may have an advantage in stabilizing the amount of noise needed to best segment stimuli across a variety of tasks. That being said, the absolute performance of the AE model did not substantially decrease as a function of noise (Figure 6 **left**), which means that it is possible that even the AE could yield meaningful segmentation results using a consistently low magnitude of noise.

3.4. Number of time steps needed

We evaluated model performance as a function of the number of samples collected from the model that were used for segmentation (Figure 6 **right**). We found that for an appropriate level of noise used in segmentation, both the AE and the VAE models' performance asymptotes quickly, with as few as a handful of time steps. Reducing the number of time steps used from 80 (Figure 6 **left**) down to as few as 5 barely reduces segmentation accuracy (Figure 6 **middle**).

4. Conclusion and Discussion

We present Latent Noise Segmentation (LNS) that allows us to turn a deep neural network that is trained on a generic image reconstruction task into a segmentation algorithm without training any additional parameters, using only independent neural noise. The intuition behind why independent neural noise works for seg-

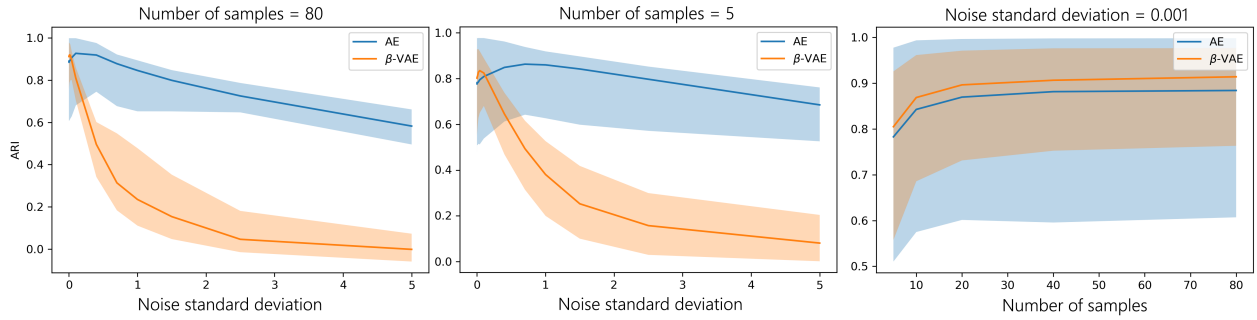


Figure 6: **Sensitivity to Noise Magnitude and Time Steps. Left:** AE and VAE segmentation performance as a function of noise standard deviation, evaluated with 80 time steps. **Middle:** AE and VAE segmentation performance as a function of noise standard deviation, evaluated with 5 time steps. **Right:** AE and VAE segmentation performance as a function of the number of samples for a fixed noise $\sigma = 0.001$. Note the change of y-axis scale. Shaded areas show 75th and 25th percentiles over all evaluated segmentation masks.

mentation is that independent noise affects neurons that are selective to a presented stimulus more than those that are not. This idea is not entirely new — noise has been shown to theoretically be able to *enhance* the signal of a stimulus when the signal is small (Benzi et al., 1981; Kitajo et al., 2003; Buchin et al., 2016); however, its application to segmentation and perceptual grouping, to the best of our knowledge, is novel.

Importantly, our goal is not to propose a state of the art segmentation algorithm (though our model beats other tested baseline models), but rather to understand the role of neural noise and how it can be applied to achieve segmentation and perceptual grouping. We implement LNS and find that it is possible to make pre-trained Autoencoders (AE) and Variational Autoencoders (VAE) segment images. We present the Good Gestalt (GG) datasets — a collection of datasets intended to investigate perceptual organization in humans — and show that LNS is able to explain the formation of 'Good Gestalt' in a neural network model. Our findings give a unified high-level account of the result of many perceptual grouping phenomena in humans: 1) learning encodes an approximation of the training dataset distribution, and 2) neural noise reveals the learned representation in out-of-distribution samples, resulting in certain hallmarks of perception, like effects of illusory contours, closure, continuity, proximity and occlusion. Establishing this link between training data (Appendix E.1), neural noise, and human perception is an important and timely step towards better models of visual representation (Bowers et al., 2022).

Interestingly, model learning rule plays a role in segmentation. We show that a very small amount of independent noise is optimal for segmentation in the case of the VAE, but not for the AE. Similarly, the VAE finds the correct number of objects in the image (Appendix F.1) even when the AE does not. This deviation between the VAE and the AE is interesting, as the VAE has connections with predictive coding (Marino, 2022; Boutin et al., 2020) and the free energy principle (Friston, 2010; Mazzaglia et al., 2022), and has been shown empirical support as a coding principle in primate cortex (Higgins et al., 2021). With both learning rules, segmentation performance asymptotes quickly, suggesting that the time scale of the segmentation in our models is practically feasible for constrained systems.

These results suggest a number of interesting questions for future research. First, our focus here has been on revealing a potential mechanism for perceptual grouping and segmentation. While we have demonstrated that the approach partially scales to the case of face segmentation (Appendix B), a relevant question is whether it is possible to scale the approach to improve unsupervised segmentation state-of-the-art (Bear et al., 2023; Engelcke et al., 2022). Second, while we have targeted a large collection of most relevant grouping principles, there are many that we have not tested here. As we have shown for the datasets we have tested, we speculate that most or all phenomena in Gestalt perception can be reproduced as a result of a combination of past experience (learning) and noise in the perceptual system. Of interest is also understanding how LNS interacts with perception under a motion stream, which we have not considered here. Finally, while we do not make any claims about the primate visual system, our results suggest a potential benefit of independent noise in the visual system. Understanding how noise interacts with perception in primate cortex is an active area of research (Destexhe & Rudolph-Lilith, 2012; Miller & Troyer, 2002; Stein et al., 2005; Guo et al., 2018; McDonnell & Ward, 2011), and we hope our results stimulate discussion about the potential benefits of noise for perceptual systems in general.

Reproducibility Statement

Mathematical details about our results are included in Appendix A.1. Full details on our GG datasets are included in Appendix E.1, and full details on model architecture and training are included in Appendix E.2. Code and data are public in our git repository: github.com/ZhengqingUUU/LatentNoiseSegmentation.

Impact Statement

The paper presents work whose goal is to advance the field of Machine Learning. Our work makes improvements in unsupervised segmentation, and our work could contribute to potential societal consequences in several ways. Our work enables the use of neural noise for segmentation without labels, which could be used for harmful applications such as misuse of face segmentation. Since our methodology is unsupervised, its performance depends on the underlying trained model on which it is applied, and as such our methodology could make mistakes or exhibit biases against underrepresented groups. Furthermore, the addition of noise in the network could cause further mistakes by the model. In total, mitigating these risks by de-biasing underlying networks used for segmentation, and by verifying outputs is important in outcome-critical applications.

Acknowledgments

We thank Jirko Rubruck, Martin Schrimpf, and Badr AlKhamissi for helpful discussions, and Marc Repnow for technical support. MHH and BL were supported by a grant from the Swiss National Science Foundation (N.320030_176153; "Basics of visual processing: from elements to figures").

References

Bear, D. M., Feigelis, K., Chen, H., Lee, W., Venkatesh, R., Kotar, K., Durango, A., and Yamins, D. L. K. Unifying (Machine) Vision via Counterfactual World Modeling. *ArXiv pre-print*, (arXiv:2306.01828), June 2023. arXiv:2306.01828 [cs] type: article.

Benzi, R., Sutera, A., and Vulpiani, A. The mechanism of stochastic resonance. *Journal of Physics A: Mathematical and General*, 14(11):L453, November 1981. ISSN 0305-4470. doi: 10.1088/0305-4470/14/11/006.

Biscione, V. and Bowers, J. S. Mixed Evidence for Gestalt Grouping in Deep Neural Networks. *Computational Brain & Behavior*, 6(3):438–456, September 2023. ISSN 2522-087X. doi: 10.1007/s42113-023-00169-2.

Boutin, V., Zerroug, A., Jung, M., and Serre, T. Iterative VAE as a predictive brain model for out-of-distribution generalization. *ArXiv pre-print*, (arXiv:2012.00557), December 2020. arXiv:2012.00557 [cs] type: article.

Bowers, J. S., Malhotra, G., Dujmović, M., Montero, M. L., Tsvetkov, C., Biscione, V., Puebla, G., Adolfi, F., Hummel, J. E., Heaton, R. F., Evans, B. D., Mitchell, J., and Blything, R. Deep Problems with Neural Network Models of Human Vision. *Behavioral and Brain Sciences*, pp. 1–74, December 2022. ISSN 0140-525X, 1469-1825. doi: 10.1017/S0140525X22002813. Publisher: Cambridge University Press.

Buchin, A., Rieubland, S., Häusser, M., Gutkin, B. S., and Roth, A. Inverse Stochastic Resonance in Cerebellar Purkinje Cells. *PLoS Computational Biology*, 12 (8):e1005000, August 2016. ISSN 1553-734X. doi: 10.1371/journal.pcbi.1005000.

Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Sengupta, B., and Bharath, A. A. Generative adversarial networks: An overview. *IEEE Signal Processing Magazine*, 35(1):53–65, 2018. doi: 10.1109/MSP.2017.2765202.

Destexhe, A. and Rudolph-Lilith, M. Neuronal Noise. *Springer US*, 2012.

Doerig, A., Bornet, A., Choung, O. H., and Herzog, M. H. Crowding reveals fundamental differences in local vs. global processing in humans and machines. *Vision Research*, 167:39–45, February 2020. ISSN 0042-6989. doi: 10.1016/j.visres.2019.12.006.

Elder, J. and Zucker, S. The effect of contour closure on the rapid discrimination of two-dimensional shapes. *Vision Research*, 33(7):981–991, May 1993. ISSN 0042-6989. doi: 10.1016/0042-6989(93)90080-G.

Elder, J. H. and Goldberg, R. M. Ecological statistics of Gestalt laws for the perceptual organization of contours. *Journal of Vision*, 2(4):5, August 2002. ISSN 1534-7362. doi: 10.1167/2.4.5.

Engelcke, M., Kosiorek, A. R., Parker Jones, O., and Posner, I. GENESIS: Generative Scene Inference and Sampling with Object-Centric Latent Representations. In *International Conference on Learning Representations (ICLR)*, 2020a.

Engelcke, M., Parker Jones, O., and Posner, I. Reconstruction Bottlenecks in Object-Centric Generative Models. *ICML Workshop on Object-Oriented Learning*, 2020b.

Engelcke, M., Jones, O. P., and Posner, I. GENESIS-V2: Inferring Unordered Object Representations without Iterative Refinement. *ArXiv pre-print*, (arXiv:2104.09958), January 2022. arXiv:2104.09958 [cs, stat] type: article.

Field, D. J., Hayes, A., and Hess, R. F. Contour integration by the human visual system: Evidence for a local "association field". *Vision Research*, 33(2):173–193, January 1993. ISSN 0042-6989. doi: 10.1016/0042-6989(93)90156-Q.

Friston, K. The free-energy principle: a unified brain theory? *Nature Reviews Neuroscience*, 11(2):127–138, February 2010. ISSN 1471-0048. doi: 10.1038/nrn2787. Number: 2 Publisher: Nature Publishing Group.

Geirhos, R., Temme, C. R. M., Rauber, J., Schütt, H. H., Bethge, M., and Wichmann, F. A. Generalisation in humans and deep neural networks. *Advances in Neural Information Processing Systems*, 31, 2018.

Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F. A., and Brendel, W. ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. *International Conference on Learning Representations*, 2019.

Geisler, W. S., Perry, J. S., Super, B. J., and Gallogly, D. P. Edge co-occurrence in natural images predicts contour grouping performance. *Vision Research*, 41(6):711–724, March 2001. ISSN 0042-6989. doi: 10.1016/S0042-6989(00)00277-7.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. 27, 2014.

Greff, K., van Steenkiste, S., and Schmidhuber, J. On the Binding Problem in Artificial Neural Networks. *ArXiv pre-print*, (arXiv:2012.05208), December 2020. arXiv:2012.05208 [cs] type: article.

Guo, D., Perc, M., Liu, T., and Yao, D. Functional importance of noise in neuronal information processing. *Europhysics Letters*, 124(5):50001, December 2018. ISSN 0295-5075. doi: 10.1209/0295-5075/124/50001. Publisher: EDP Sciences, IOP Publishing and Società Italiana di Fisica.

Hamilton, M., Zhang, Z., Hariharan, B., Snavely, N., and Freeman, W. T. Unsupervised Semantic Segmentation by Distilling Feature Correspondences. *ArXiv pre-print*, (arXiv:2203.08414), March 2022. arXiv:2203.08414 [cs, stat] type: article.

Herzog, M. H., Sayim, B., Chicherov, V., and Manassi, M. Crowding, grouping, and object recognition: A matter of appearance. *Journal of Vision*, 15(6):5, May 2015. ISSN 1534-7362. doi: 10.1167/15.6.5.

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. *International Conference on Learning Representations*, 2017.

Higgins, I., Chang, L., Langston, V., Hassabis, D., Summerfield, C., Tsao, D., and Botvinick, M. Unsupervised deep learning identifies semantic disentanglement in single inferotemporal face patch neurons. *Nature Communications*, 12(1):6456, November 2021. ISSN 2041-1723. doi: 10.1038/s41467-021-26751-5. Number: 1 Publisher: Nature Publishing Group.

Hinton, G. How to Represent Part-Whole Hierarchies in a Neural Network. *Neural Computation*, 35(3):413–452, February 2023. ISSN 0899-7667. doi: 10.1162/neco_a_01557.

Holcombe, A. O. and Cavanagh, P. Early binding of feature pairs for visual perception. *Nature Neuroscience*, 4(2):127–128, February 2001. ISSN 1546-1726. doi: 10.1038/83945. Number: 2 Publisher: Nature Publishing Group.

Hubert, L. and Arabie, P. Comparing partitions. *Journal of Classification*, 2(1):193–218, December 1985. ISSN 1432-1343. doi: 10.1007/BF01908075.

Jäkel, F., Singh, M., Wichmann, F. A., and Herzog, M. H. An overview of quantitative approaches in Gestalt perception. *Vision Research*, 126:3–8, September 2016. ISSN 0042-6989. doi: 10.1016/j.visres.2016.06.004.

Kanizsa, G. Organization in Vision: Essays on Gestalt Perception. *Praeger, New York*, 1979.

Kellman, P. J. and Spelke, E. S. Perception of partly occluded objects in infancy. *Cognitive Psychology*, 15(4):483–524, 1983. ISSN 1095-5623. doi: 10.1016/0010-0285(83)90017-8. Place: Netherlands Publisher: Elsevier Science.

Kimchi, R. Primacy of wholistic processing and global/local paradigm: a critical review. *Psychological Bulletin*, 112(1):24–38, July 1992. ISSN 0033-2909. doi: 10.1037/0033-2909.112.1.24.

Kingma, D. P. and Welling, M. Auto-Encoding Variational Bayes. *International Conference on Learning Representations*, December 2014. arXiv:1312.6114 [cs, stat] type: article.

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., Dollár, P., and Girshick, R. Segment Anything. *ArXiv pre-print*, (arXiv:2304.02643), April 2023. arXiv:2304.02643 [cs] type: article.

Kitajo, K., Nozaki, D., Ward, L. M., and Yamamoto, Y. Behavioral Stochastic Resonance within the Human Brain. *Physical Review Letters*, 90(21):218103, May 2003. doi: 10.1103/PhysRevLett.90.218103. Publisher: American Physical Society.

Koffka, K. Perception: An introduction to the Gestalt-theorie. *Psychological bulletin*, 19(10):531–585, 1922.

Kovács, I. and Julesz, B. A closed curve is much more than an incomplete one: effect of closure in figure-ground segmentation. *Proceedings of the National Academy of Sciences*, 90(16):7495–7497, August 1993. doi: 10.1073/pnas.90.16.7495. Publisher: Proceedings of the National Academy of Sciences.

Kramer, M. A. Nonlinear principal component analysis using autoassociative neural networks. *AIChE Journal*, 37 (2):233–243, 1991. ISSN 1547-5905. doi: 10.1002/aic.690370209.

Kubovy, M. and van den Berg, M. The whole is equal to the sum of its parts: A probabilistic model of grouping by proximity and similarity in regular patterns. *Psychological Review*, 115(1):131–154, 2008. ISSN 1939-1471. doi: 10.1037/0033-295X.115.1.131. Place: US Publisher: American Psychological Association.

Kwon, T., Agrawal, K., Li, Y., and Pizlo, Z. Spatially-global integration of closed, fragmented contours by finding the shortest-path in a log-polar representation. *Vision Research*, 126:143–163, September 2016. ISSN 0042-6989. doi: 10.1016/j.visres.2015.06.007.

Lesher, G. W. Illusory contours: Toward a neurally based perceptual theory. *Psychonomic Bulletin & Review*, 2 (3):279–321, September 1995. ISSN 1531-5320. doi: 10.3758/BF03210970.

Linsley, D., Kim, J., Veerabadran, V., Windolf, C., and Serre, T. Learning long-range spatial dependencies with horizontal gated recurrent units. *Advances in Neural Information Processing Systems*, 31, 2018.

Liu, Z., Luo, P., Wang, X., and Tang, X. Deep Learning Face Attributes in the Wild. *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

Locatello, F., Weissenborn, D., Unterthiner, T., Mahendran, A., Heigold, G., Uszkoreit, J., Dosovitskiy, A., and Kipf, T. Object-Centric Learning with Slot Attention. *Advances in Neural Information Processing Systems*, 33: 11525–11538, 2020.

Long, J., Shelhamer, E., and Darrell, T. Fully Convolutional Networks for Semantic Segmentation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

Löwe, S., Lippe, P., Rudolph, M., and Welling, M. Complex-Valued Autoencoders for Object Discovery. *Transactions on Machine Learning Research*, September 2022. ISSN 2835-8856.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards Deep Learning Models Resistant to Adversarial Attacks. *International Conference on Learning Representations*, 2018.

Manwani, A. and Koch, C. Signal Detection in Noisy Weakly-Active Dendrites. *Advances in Neural Information Processing Systems*, 11, 1998.

Marini, F. and Marzi, C. A. Gestalt Perceptual Organization of Visual Stimuli Captures Attention Automatically: Electrophysiological Evidence. *Frontiers in Human Neuroscience*, 10, 2016. ISSN 1662-5161.

Marino, J. Predictive coding, variational autoencoders, and biological connections. *Neural Computation*, 34(1):1–44, 2022.

Mazzaglia, P., Verbelen, T., Çatal, O., and Dhoedt, B. The free energy principle for perception and action: A deep learning perspective. *Entropy*, 24(2):301, feb 2022. doi: 10.3390/e24020301.

McDonnell, M. D. and Ward, L. M. The benefits of noise in neural systems: bridging theory and experiment. *Nature Reviews Neuroscience*, 12(7):415–425, July 2011. ISSN 1471-0048. doi: 10.1038/nrn3061. Number: 7 Publisher: Nature Publishing Group.

McInnes, L., Healy, J., and Melville, J. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *ArXiv pre-print*, (arXiv:1802.03426), 2018. arXiv:1802.03426 [cs, stat] type: article.

Miller, K. D. and Troyer, T. W. Neural Noise Can Explain Expansive, Power-Law Nonlinearities in Neural Response Functions. *Journal of Neurophysiology*, 87 (2):653–659, February 2002. ISSN 0022-3077. doi: 10.1152/jn.00425.2001. Publisher: American Physiological Society.

Milner, P. M. A model for visual shape recognition. *Psychological Review*, 81(6):521–535, 1974. ISSN 1939-1471. doi: 10.1037/h0037149. Place: US Publisher: American Psychological Association.

Moosavi-Dezfooli, S.-M., Fawzi, A., and Frossard, P. DeepFool: A Simple and Accurate Method to Fool Deep Neural Networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Plaut, E. From principal subspaces to principal components with linear autoencoders. *arXiv preprint arXiv:1804.10253*, 2018.

Pomerantz, J. R., Sager, L. C., and Stoever, R. J. Perception of wholes and of their component parts: Some configural superiority effects. *Journal of Experimental Psychology: Human Perception and Performance*, 3(3):422–435, 1977. ISSN 1939-1277. doi: 10.1037/0096-1523.3.3.422. Place: US Publisher: American Psychological Association.

Quinlan, P. T. and Wilton, R. N. Grouping by Proximity or Similarity? Competition between the Gestalt Principles in Vision. *Perception*, 27(4):417–430, April 1998. ISSN 0301-0066. doi: 10.1068/p270417. Publisher: SAGE Publications Ltd STM.

Rezende, D. J. and Viola, F. Taming VAEs. *ArXiv pre-print*, (arXiv:1810.00597), October 2018. arXiv:1810.00597 [cs, stat] type: article.

Ringach, D. L. and Shapley, R. Spatial and Temporal Properties of Illusory Contours and Amodal Boundary Completion. *Vision Research*, 36(19):3037–3050, October 1996. ISSN 0042-6989. doi: 10.1016/0042-6989(96)00062-4.

Roelfsema, P. R. Cortical Algorithms for Perceptual Grouping. *Annual Review of Neuroscience*, 29(1):203–227, 2006. doi: 10.1146/annurev.neuro.29.051605.112939. _eprint: https://doi.org/10.1146/annurev.neuro.29.051605.112939.

Rolínek, M., Zietlow, D., and Martius, G. Variational autoencoders pursue pca directions (by accident). *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12398–12407, 2019. doi: 10.1109/CVPR.2019.01269.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models. pp. 10684–10695, June 2022.

Ronneberger, O., Fischer, P., and Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pp. 234–241, 2015. doi: 10.1007/978-3-319-24574-4_28.

Sabour, S., Frosst, N., and Hinton, G. E. Dynamic Routing Between Capsules. *Advances in Neural Information Processing Systems*, 30, 2017.

Sigman, M., Cecchi, G. A., Gilbert, C. D., and Magnasco, M. O. On a common circle: Natural scenes and Gestalt rules. *Proceedings of the National Academy of Sciences*, 98(4):1935–1940, February 2001. doi: 10.1073/pnas.98.4.1935. Publisher: Proceedings of the National Academy of Sciences.

Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., and Ganguli, S. Deep unsupervised learning using nonequilibrium thermodynamics. 37:2256–2265, 07–09 Jul 2015.

Spelke, E. S. Principles of object perception. *Cognitive Science*, 14(1):29–56, January 1990. ISSN 0364-0213. doi: 10.1016/0364-0213(90)90025-R.

Stanić, A., Gopalakrishnan, A., Irie, K., and Schmidhuber, J. Contrastive Training of Complex-Valued Autoencoders for Object Discovery. *ArXiv pre-print*, (arXiv:2305.15001), May 2023. arXiv:2305.15001 [cs] type: article.

Stein, R. B., Gossen, E. R., and Jones, K. E. Neuronal variability: noise or part of the signal? *Nature Reviews Neuroscience*, 6(5):389–397, May 2005. ISSN 1471-0048. doi: 10.1038/nrn1668. Number: 5 Publisher: Nature Publishing Group.

Todorovic, D. Gestalt principles. *Scholarpedia*, 3(12):5345, December 2008. ISSN 1941-6016. doi: 10.4249/scholarpedia.5345.

Wagemans, J., Elder, J. H., Kubovy, M., Palmer, S. E., Peterson, M. A., Singh, M., and von der Heydt, R. A century of Gestalt psychology in visual perception: I. Perceptual grouping and figure–ground organization. *Psychological Bulletin*, 138(6):1172–1217, 2012a. ISSN 1939-1455. doi: 10.1037/a0029333. Place: US Publisher: American Psychological Association.

Wagemans, J., Feldman, J., Gepshtein, S., Kimchi, R., Pomerantz, J. R., van der Helm, P. A., and van Leeuwen, C. A century of Gestalt psychology in visual perception: II. Conceptual and theoretical foundations. *Psychological Bulletin*, 138(6):1218–1252, 2012b. ISSN 1939-1455. doi: 10.1037/a0029334. Place: US Publisher: American Psychological Association.

Wang, L., Weng, X., and He, S. Perceptual Grouping without Awareness: Superiority of Kanizsa Triangle in Breaking Interocular Suppression. *PLOS ONE*, 7(6):e40106, June 2012. ISSN 1932-6203. doi: 10.1371/journal.pone.0040106. Publisher: Public Library of Science.

Wang, X., Yu, Z., Mello, S. D., Kautz, J., Anandkumar, A., Shen, C., and Alvarez, J. M. Freesolo: Learning to segment objects without annotations. 2022.

Wertheimer, M. Laws of Organization in Perceptual Forms. *Psycologische Forschung*, 4:301–350, 1923.

Zietlow, D., Rolinek, M., and Martius, G. Demystifying inductive biases for β-vae based architectures. *ArXiv*, abs/2102.06822, 2021.

A. Appendix

A.1. Object separability

The Latent Noise Segmentation approach proposed in this work is capable of separating pixels into different objects regardless of their actual values in the input. This fact is a necessary result yielded by a Variational Autoencoder (VAE) pursuing its optimization objective. Here, we formalize the mathematical intuition behind why this is the case.

A.1.1. Background

(Rolínek et al., 2019; Zietlow et al., 2021) showed that VAE-based Deep Neural Networks (DNNs) learn to represent local Principal Component (PC) axes. The training process of a VAE aims to minimize the reconstruction loss, part of which can be re-assembled as the stochastic reconstruction loss:

$$\hat{L}_{\text{rec}}\left(\mathbf{x}^i\right) = \mathop{\mathbb{E}}_{\mathbf{z}^i \sim \mathcal{N}(\boldsymbol{\mu}(\mathbf{x}^i), \text{diag}(\boldsymbol{\sigma}^2(\mathbf{x}^i)))} \| \operatorname{Dec}_\theta\left(\boldsymbol{\mu}\left(\mathbf{x}^i\right)\right) - \operatorname{Dec}_\theta\left(\mathbf{z}^i\right) \|^2, \tag{2}$$

where the superscript i denotes the index of the training samples.

We further define a Jacobian matrix to denote the decoded output's derivative with respect to the latent variables:

$$J_i = \left.\frac{\partial \operatorname{Dec}_\theta(\mathbf{z})}{\partial \mathbf{z}}\right|_{\mathbf{z}=\boldsymbol{\mu}(\mathbf{x}^i)} \in \mathbb{R}^{D \times d}, \tag{3}$$

where $D = img_x \times img_y \times C$ is the dimension of the images, and d is the number of latent nodes. (Rolínek et al., 2019) proved that optimizing equation 2 will promote pairwise orthogonality on the columns of equation 3, which means that the latent variables locally encode orthogonal features in the image space in $\mathbb{R}^D$. Furthermore, Zietlow et al. (2021) analyzed what those pairwise orthogonal directions are. With experiments, they empirically showed that those directions correspond to local PC directions. More concretely, those directions are the PC directions yielded by a Principal Component Analysis (PCA) performed on a subset of all the training samples that lie close to each other in the data space.

Built on the theoretical framework of Rolínek et al. (2019) and Zietlow et al. (2021), we are able to link these results to our context, and provide intuition for why the VAE is able to separate the object and the background.

One relevant technical detail is that the above statement regarding the latent nodes pursuing local local PC directions is derived from a VAE structure with a standardized objective as formulated by Higgins et al. (2017). However, it is easy to check the proof from Rolínek et al. (2019) to see that the same conclusion applies to our case, where the GECO (Rezende & Viola, 2018) loss is used.

Furthermore, although the above-mentioned argument from (Rolínek et al., 2019; Zietlow et al., 2021) is for the Jacobians (equation 3) calculated at training samples, we posit that the same conclusion holds for the testing samples. After all, in our setting, the training samples and the testing samples are close to each other. Henceforth, i will be used to denote the index of testing samples.

A.1.2. Mathematical Intuition Underpinning Gestalt Perception

In the following, we will use p_1 to denote a pixel belonging to one object of the image, and p_2 to denote a pixel belonging to a different object of the image.

In our segmentation algorithm, we first repeatedly apply noise to the latent units and observe the resultant changes in output pixels. We denote the added noise, at the time step of n, to be $\boldsymbol{\xi}_n = (\xi_{n,j})^\intercal_{j \in \{1,2,\cdots,d\}} \in \mathbb{R}^d$, where the second subscript j denotes the indices of the latent units to which noise is added.

Based on multivariate Taylor expansion, the reconstruction yielded by the latent affected by the noise can be written as:

$$\tilde{\mathbf{x}}_n^i = \operatorname{Dec}\left(\boldsymbol{\mu}\left(\mathbf{x}^i\right) + \boldsymbol{\xi}_n\right) = \operatorname{Dec}\left(\boldsymbol{\mu}\left(\mathbf{x}^i\right)\right) + \sum_{j=1}^{d} \xi_{n,j} \left.\frac{\partial \operatorname{Dec}(\mathbf{z})}{\partial z_j}\right|_{\mathbf{z}=\boldsymbol{\mu}(\mathbf{x}^i)} + o\left(\boldsymbol{\xi}_n\right), \tag{4}$$

in which the superscript i of $\mathbf{x}$ denotes the index of the input image, the subscript j of z denotes the latent index. $\left.\frac{\partial \operatorname{Dec}(\mathbf{z})}{\partial z_j}\right|_{\mathbf{z}=\boldsymbol{\mu}(\mathbf{x}^i)}$ is the jth column of the Jacobian in equation 3. $o\left(\boldsymbol{\xi}_n\right)$ denotes the residual terms of the Taylor expansion.

In our segmentation algorithm, we subtract the reconstruction affected by one set of noise from the reconstruction affected by the previous set of noise. If we use $\boldsymbol{\delta}_n = \left(\delta_{n,j}\right)^{\intercal}_{j\in\{1,2,\cdots,d\}} \in \mathbb{R}^d$ to represent $(\boldsymbol{\xi}_{n+1} - \boldsymbol{\xi}_n)$, then we can write the difference between the two consecutive noisy reconstructions as:

$$\Delta\tilde{\mathbf{x}}^i_n = \tilde{\mathbf{x}}^i_{n+1} - \tilde{\mathbf{x}}^i_n = \sum_{j=1}^{d} \delta_{n,j} \left.\frac{\partial \operatorname{Dec}\left(\mathbf{z}\right)}{\partial z_j}\right|_{\mathbf{z}=\boldsymbol{\mu}(\mathbf{x}^i)} \in \mathbb{R}^D, \tag{5}$$

where we omit the $o(\cdot)$ term, as they are negligible when the perturbing noise is sufficiently small.

Furthermore, we use $\Delta\tilde{\mathbf{x}}^i_{n,c,p_1} \in \mathbb{R}$ to denote the value of the component of $\Delta\tilde{\mathbf{x}}^i_n$ corresponding to the value of the pixel p_1 in the channel c and $\Delta\tilde{\mathbf{x}}^i_{n,c,p_2} \in \mathbb{R}$ to denote the value of the component of $\Delta\tilde{\mathbf{x}}^i_n$ corresponding to the value of the pixel p_2 in the channel c.

Our visual segmentation pipeline separates pixel p_1 from pixel p_2 by separating the vector $\Delta\tilde{\mathbf{X}}^i_{p_1} = \left(\Delta\tilde{\mathbf{x}}^i_{n,c,p_1}\right)^{\intercal}_{(n,c)\in\{1,2,\cdots,N-1\}\times\{1,\cdots,C\}} \in \mathbb{R}^{(N-1)\times C}$ from the vector $\Delta\tilde{\mathbf{X}}^i_{p_2} = \left(\Delta\tilde{\mathbf{x}}^i_{n,c,p_2}\right)^{\intercal}_{(n,c)\in\{1,2,\cdots,N-1\}\times\{1,\cdots,C\}} \in \mathbb{R}^{(N-1)\times C}$.

More concretely, as indicated in Appendix D, the separation between the two vectors are based on the angles they point toward in the $[(N-1)\times C]$-dimensional space. In other words, the case in which the two pixels cannot be separated with our pipeline can only be caused by the case that no matter how we sample our noise, the following condition holds with probability 1.

Condition for Inseparability. There exist some constant $r_{p_1,p_2} > 0$ such that, for the given pixel pair (p_1, p_2) and any channel c and any time step n: $\Delta\tilde{\mathbf{x}}^i_{n,c,p_1} = r_{p_1,p_2}\Delta\tilde{\mathbf{x}}^i_{n,c,p_2}$.

The equation in the condition for inseparability can be expanded as:

$$\sum_{j=1}^{d} \delta_{n,j} \left.\frac{\partial \operatorname{Dec}\left(\mathbf{z}\right)_{c,p_1}}{\partial z_j}\right|_{\mathbf{z}=\boldsymbol{\mu}(\mathbf{x}^i)} = r_{p_1,p_2} \sum_{j=1}^{d} \delta_{n,j} \left.\frac{\partial \operatorname{Dec}\left(\mathbf{z}\right)_{c,p_2}}{\partial z_j}\right|_{\mathbf{z}=\boldsymbol{\mu}(\mathbf{x}^i)} \tag{6}$$

The fact that condition for inseparability holding with probability 1 implies inseparability is intuitive, and we also provide a proof for it in A.1.3.

In the following, we will argue why the condition for inseparability is unlikely to hold with probability 1 given the background in Section A.1.1.

For the condition of inseparability to hold, the Jacobian (in equation 3) and the noise must satisfy one of the following two cases.

The **first case** is when the Jacobian satisfies:

$$\left.\frac{\partial \operatorname{Dec}\left(\mathbf{z}\right)_{c,p_1}}{\partial z_j}\right|_{\mathbf{z}=\boldsymbol{\mu}(\mathbf{x}^i)} = r_{p_1,p_2} \left.\frac{\partial \operatorname{Dec}\left(\mathbf{z}\right)_{c,p_2}}{\partial z_j}\right|_{\mathbf{z}=\boldsymbol{\mu}(\mathbf{x}^i)} \tag{7}$$

for some fixed r_{p_1,p_2} for all $j \in \{1, 2, \cdots, d\}$ and all channels c. Here $\operatorname{Dec}\left(\mathbf{z}\right)_{c,p_2/p_2} \in \mathbb{R}$ is the component of the D-dimensional vector $\operatorname{Dec}\left(\mathbf{z}\right)$ that corresponds to the p_1 or p_2 pixel in channel c. In this case, the condition for inseparability trivially holds with probability 1.

The **second case** is where equation 7 does not hold for any fixed r_{p_1,p_2} for some $j \in \{1, 2, \cdots, d\}$ and some channel c. In this case, the condition for inseparability might still hold.

In the following, we will explain why both cases cannot hold with probability 1 when p_1 and p_2 are pixels belonging to different objects of the image, which means that they have independent pixel values in the training dataset (see Appendix E.1 for more details).

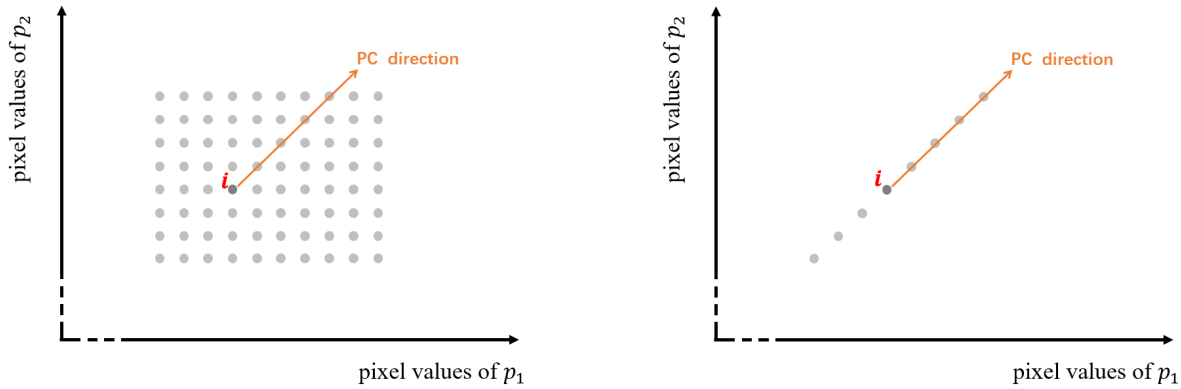
(1) WHY IS THE FIRST CASE UNLIKELY TO HOLD?

The argument of the first case boils down to a characterization of the Jacobian in equation 3. A complete theoretical understanding of the Jacobian has yet to be established in the past literature to the best of our knowledge. However, from the conclusions derived from Rolínek et al. (2019) and Zietlow et al. (2021), which we briefly mentioned in Appendix A.1.1, the first case is unlikely to happen as it intuitively goes against the behavior of PCA.

In more detail, as demonstrated in (Zietlow et al., 2021), the columns of the Jacobian (equation 3) should point into directions in the D-dimensional space that are principal components extracted from a compact cluster of training images in the D-dimensional space. In addition, equation 7 implies that all the principal components characterize the pixel values (in each channel) of p_1 and p_2 to have a strict linear relationship with the same ratio $r_{p_1 p_2}$. However, as introduced in Appendix E.1, the training dataset is generated by assuming the pixel values of p_1 and p_2 (when they do not belong to the same object) are independent. Thus, it is unlikely to have all local PC directions satisfying equation 7.

For clarity, we provide a diagram in the left panel of Figure 7. Here we project the entire training dataset (denoted by the gray dots) to a 2-dimensional space. Each dimension stands for the pixel value of p_1 or p_2. The grey dots are arranged in a way to demonstrate the independence of the two pixel values. The orange arrow denotes a potentially learned local PC direction, moving along which the variation of the two pixel values follows a strict linear ratio. Since there are data variances in other directions, it is highly unlikely that all the local PC directions learned from local data points only capture directions of such kind.

On the other hand, we can also develop intuition into why any two pixels belonging to the same object can be inseparable with the right panel of Figure 7. If p_1 and p_2 are pixels belonging to the same object, the way the dataset is generated (discussed in Appendix E.2) dictates that the training samples projected onto the pixel value space will only stretch in a direction where there exists a strict linear ratio between the values of the two objects. In this case, the learned local PC directions are highly likely to point only in the direction on this 2-dimensional plane, as there exists no data variance along other directions. Such learned local PC directions guarantee that the two pixels of the same object will not be separated in the segmentation process, for it implies the Jacobian satisfies the condition described in equation 7 in the first case.



(a) p_1 and p_2 belong to different parts

(b) p_1 and p_2 belong to the same part

Figure 7: **(Left)** An example of a direction that all the local PC directions should follow to have p_1 and p_2 become inseparable. **(Right)** An example of how the local PC directions might look like when they are projected into the pixel value space of two pixels of the same part of the images.

(2) WHY CAN THE SECOND CASE NOT HOLD WITH PROBABILITY 1?

To understand this case, we first rewrite equation 6 as:

$$\sum_{j=1}^{d} \left[\frac{\partial \operatorname{Dec}(\mathbf{z})_{c,p_1}}{\partial z_j}\bigg|_{\mathbf{z}=\boldsymbol{\mu}(\mathbf{x}^i)} - r_{p_1,p_2} \frac{\partial \operatorname{Dec}(\mathbf{z})_{c,p_2}}{\partial z_j}\bigg|_{\mathbf{z}=\boldsymbol{\mu}(\mathbf{x}^i)} \right] \delta_{n,j} = \sum_{j=1}^{d} C_{c,j}(r_{p_1,p_2})\delta_{n,j} = 0, \tag{8}$$

in which $C_{c,j}(r_{p_1,p_2})$ is used to summarize the term in the square bracket $[\ \cdot\ ]$. As we are considering the second case, we know that $C_{c,j}(r_{p_1,p_2}) \neq 0$ for at least one $j \in \{1, 2, \cdots, d\}$ and one channel c. Thus, in the second case, the condition

for inseparability can be interpreted as: $\boldsymbol{\delta}_n = (\delta_{n,1}, \delta_{n,2}, \cdots, \delta_{n,d})^\intercal \in \mathbb{R}^d$ lies in a hyperplane as described by equation 8, and the hyperplane has less than d dimensions. Such an event cannot happen with probability 1. This is because $\boldsymbol{\delta}_n$ has a Gaussian distribution in the d-dimensional space, implying its probability mass on a hyperplane with lower dimensions is 0.

A.1.3. Why Probability 1 Implies Inseparability

Here, we give a formal derivation of why the condition for inseparability must hold with probability 1 to make p_1 and p_2 inseparable. To simplify notation, let us denote the event that equation 6 holds for some time step n some channel c and some positive constant r_{p_1,p_2} to be $\mathcal{A}_{n,c}(r_{p_1,p_2})$. It would require $\mathcal{A}_{n,c}(r_{p_1,p_2})$ to hold for all $n \in \{1, 2, \cdots, N-1\}$ and all $c \in \{1, \cdots, C\}$ to make p_1 and p_2 inseparable, otherwise $\Delta\tilde{\mathbf{X}}^i_{p_1}$ and $\Delta\tilde{\mathbf{X}}^i_{p_2}$, both belonging to $\mathbb{R}^{(N-1)\times C}$, do not point into the same direction, making the two pixels separable.

Intuitively, if the condition for inseparability does not hold with probability 1, then the events $\{\mathcal{A}_n(r_{p_1,p_2})\}_{n\in\{1,2,\cdots,N-1\}}$ are less and less likely to happen simultaneously as N increases, meaning p_2 and p_1 are always separable given N sufficiently large. Here below, we characterize this intuition rigorously.

Note that $\mathcal{A}_{n,c}(r_{p_1,p_2})$ and $\mathcal{A}_{n+1,c}(r_{p_1,p_2})$ may be dependent on each other, as we can show that $\boldsymbol{\delta_n}$ and $\boldsymbol{\delta_{n+1}}$ are dependent of each other. $\boldsymbol{\delta_n}$ involves the noise pair of $(\boldsymbol{\xi}_n, \boldsymbol{\xi}_{n+1})$ and $\boldsymbol{\delta_{n+1}}$ involves the noise pair of $(\boldsymbol{\xi}_{n+1}, \boldsymbol{\xi}_{n+2})$. Both are affected by $\boldsymbol{\xi}_{n+1}$ and are thus dependent (a more formal explanation of which will be given in the last part of this section). Nonetheless, we have that $\mathcal{A}_{n,c}(r_{p_1,p_2})$ and $\mathcal{A}_{n+2,c}(r_{p_1,p_2})$ are independent of each other, since the former concerns the noise pair of $(\boldsymbol{\xi}_n, \boldsymbol{\xi}_{n+1})$ and the latter concerns the noise pair of $(\boldsymbol{\xi}_{n+2}, \boldsymbol{\xi}_{n+3})$, which are unaffected by each other.

Thus, suppose we arbitrarily choose a channel c_0:

$$\mathbb{P}(\text{inseparability}) \leq \mathbb{P}\left(\mathcal{A}_{1,c_0}(r_{p_1,p_2}), \mathcal{A}_{2,c_0}(r_{p_1,p_2}), \cdots, \mathcal{A}_{N-1,c_0}(r_{p_1,p_2})\right) \tag{9a}$$

$$\leq \mathbb{P}\left(\mathcal{A}_{1,c_0}(r_{p_1,p_2}), \mathcal{A}_{3,c_0}(r_{p_1,p_2}), \cdots, \mathcal{A}_{2\lceil\frac{N-1}{2}\rceil-1,c_0}(r_{p_1,p_2})\right) \tag{9b}$$

$$= \prod_{i=1}^{\lceil\frac{N-1}{2}\rceil} \mathbb{P}\left(\mathcal{A}_{2i-1,c}(r_{p_1,p_2})\right) \tag{9c}$$

The independence between $\mathcal{A}_{n,c_0}(r_{p_1,p_2})$ and $\mathcal{A}_{n+2,c_0}(r_{p_1,p_2})$ enables the fully factorized representation in equation 9c. Note that every probability term in equation 9c has the same value, and if they are all smaller than 1 (for all $r_{p_1,p_2} > 0$), the product will decay to zero exponentially (for all $r_{p_1,p_2} > 0$). Thus, we require equation 6 to happen with the probability 1 to achieve inseparability.

The last part of this section demonstrates why $\boldsymbol{\delta}_n$ and $\boldsymbol{\delta}_{n+1}$ are not independent. If $\boldsymbol{\delta}_n$ and $\boldsymbol{\delta}_{n+1}$ are independent, then we must have that the variance of their sum should be equal to the sum of their variances. However,

$$\begin{aligned}&\text{Var}\left(\boldsymbol{\delta}_{n+1} + \boldsymbol{\delta}_n\right) = \text{Var}\left(\boldsymbol{\xi}_{n+2} - \boldsymbol{\xi}_{n+1} + \boldsymbol{\xi}_{n+1} - \boldsymbol{\xi}_n\right) = \text{Var}\left(\boldsymbol{\xi}_{n+2} - \boldsymbol{\xi}_n\right) = 2\boldsymbol{\sigma}^2_{\boldsymbol{\xi}}\\ &\neq 4\boldsymbol{\sigma}^2_{\boldsymbol{\xi}} = \text{Var}\left(\boldsymbol{\xi}_{n+2} - \boldsymbol{\xi}_{n+1}\right) + \text{Var}\left(\boldsymbol{\xi}_{n+1} - \boldsymbol{\xi}_n\right) = \text{Var}\left(\boldsymbol{\delta}_{n+1}\right) + \text{Var}\left(\boldsymbol{\delta}_n\right),\end{aligned} \tag{10}$$

resulting in a contradiction. With slight abuse of notation, all the terms in equation 10 are d-dimensional vectors, and all the operations are component-wise.

A.1.4. Discussion

With the above, we demonstrated how the underlying local PCA enables VAE to segment the images as we desired. In our experiments, we also show that the Autoencoders can also manage to segment the images similarly, but that their preferred number of objects is different to the VAEs'. Indeed, qualitatively, the Autoencoder generalizes worse than the VAE (Appendix B). Previous works have drawn a link between PCA and Linear Autoencoders (Plaut, 2018). However, a full theoretical account for the connection between PCA and the more complicated non-linear autoencoder to the best of our knowledge does not yet exist. With the intuitions provided in the experimental observation in this paper, we are at a better place to formulate the connection, which we consider as a meaningful future direction. Furthermore, we have provided a mathematical link between perception in humans, representation in Deep Neural Networks, and semantic segmentation. We hope our work encourages future work in forming rigorous links between these established fields.

A.1.5. APPLICABILITY TO OTHER GENERATIVE ALGORITHMS

In this work, we have focused primarily on the Variational Autoencoder (VAE) in lieu of other generative models, such as Diffusion models (Sohl-Dickstein et al., 2015; Rombach et al., 2022) or Adversarial Generative (Goodfellow et al., 2014; Creswell et al., 2018) models. There are two primary reasons behind this choice: 1) The VAE formulation allows us to neatly follow the argumentation in (Zietlow et al., 2021) and (Rolínek et al., 2019) to arrive at a more firm understanding of why Latent Noise Segmentation (LNS) works; 2) the mathematical and computational links between the VAE formulation and other biologically plausible coding and learning schemes such as predictive coding (Boutin et al., 2020; Marino, 2022) and the Free Energy Principle (Friston, 2010; Mazzaglia et al., 2022) are clearer than for other generative algorithms to the best of our knowledge.

That being said, the fact that the Autoencoder works relatively well for segmentation using LNS is interesting, and raises the question of whether other model architectures could also perform well. We suggest that future work could empirically evaluate LNS on other model classes, such as Latent Diffusion models (Rombach et al., 2022) to understand whether the mathematical intuition behind why LNS works also generalizes beyond the simple Autoencoder model class to any model class that learns meaningful representations about images.

It is important to also note that the noise in Latent Noise Segmentation (LNS) acts in a fundamentally different way than noise in other generative models, such as diffusion models. While diffusion models are trained to denoise data in a stepwise manner, noise in LNS is not a part of the training process, and is only added at inference to stochastically explore the neighborhood of the latent representation. While both models are stochastic and add noise, noise serves a functionally different role in each: a key role in training in diffusion models; and an easy way to stochastically explore the local neighborhood of the input stimulus in LNS.

B. Generalization to More Naturalistic Stimuli

To test the generalization and potential scalability of our models, we evaluated them qualitatively on the CelebA dataset (Liu et al., 2015). The dataset consists of a large variety of faces in different contexts, lighting conditions, and backgrounds. We trained the model on the CelebA training dataset as provided, and tested on the first 100 samples of the testing dataset.

We found that the VAE model successfully finds a desired face-hair-background segmentation mask relatively often, when prompted on 3 output clusters (Figure 8). In some cases the clustering is coherent but not semantically aligned with the desired output — we suspect that in the CelebA dataset, lighting condition forms one of the strongest local PC components, and that this leads to a semantically less meaningful cluster on occasion. This effect is seen more strongly in the case of the AE, which appears to primarily follow a lighting-related clustering (Figure 9).

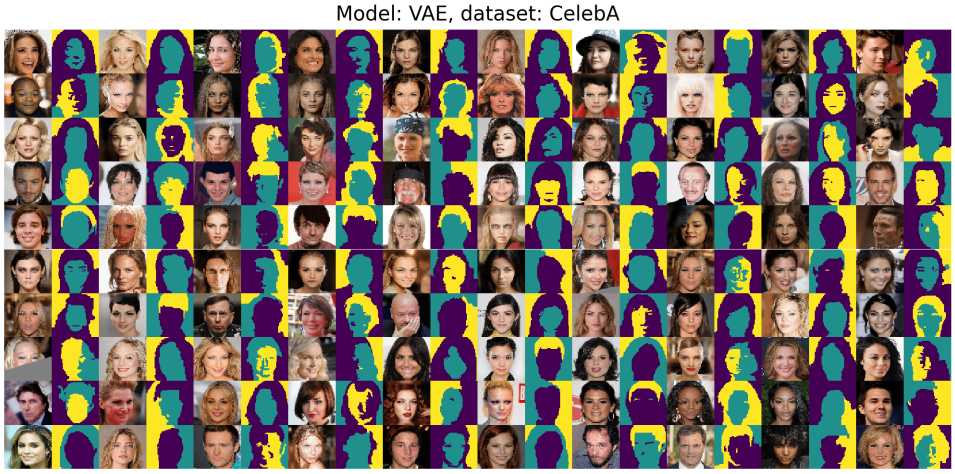


Figure 8: **All tested Variational Autoencoder outputs for a single seed of the CelebA dataset.** Inputs and outputs are displayed as horizontal pairs, with the input being the left image, and the model output being the right image.

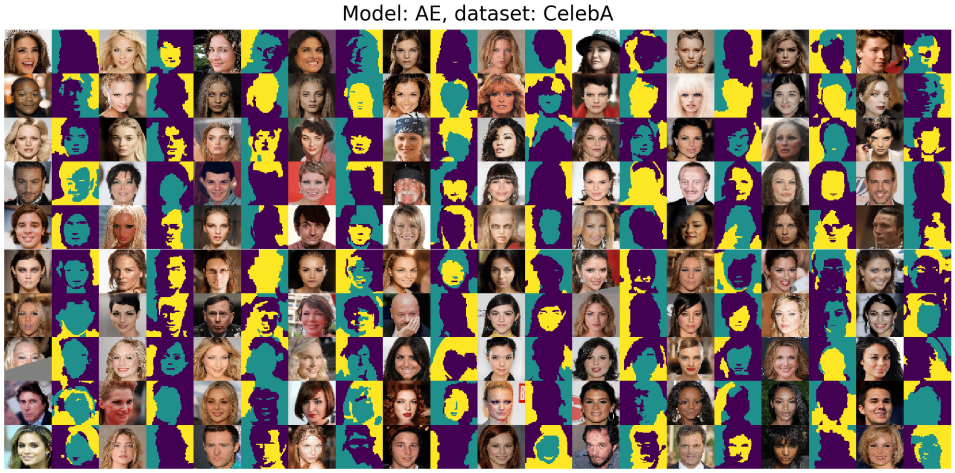


Figure 9: **All tested Autoencoder outputs for a single seed of the CelebA dataset.** Inputs and outputs are displayed as horizontal pairs, with the input being the left image, and the model output being the right image.

C. Control Experiments

C.1. Additional Control Experiment Results

We tested two high-performing unsupervised object segmentation models from the literature: Genesis (Engelcke et al., 2020a) and Genesis-v2 (Engelcke et al., 2022). We show quantitative results in Table 1. While Genesis performs quantitatively relatively well, a closer inspection of the actual segmentation performance of the model shows that with the exception of the Closure and Continuity datasets, it fails to find the correct Gestalt grouping of the stimuli 10, often opting to follow the color value cues in the testing dataset. The Genesis-v2 performs generally worse than Genesis, failing on all datasets both quantitatively and qualitatively with perhaps the exception of the Continuity dataset.

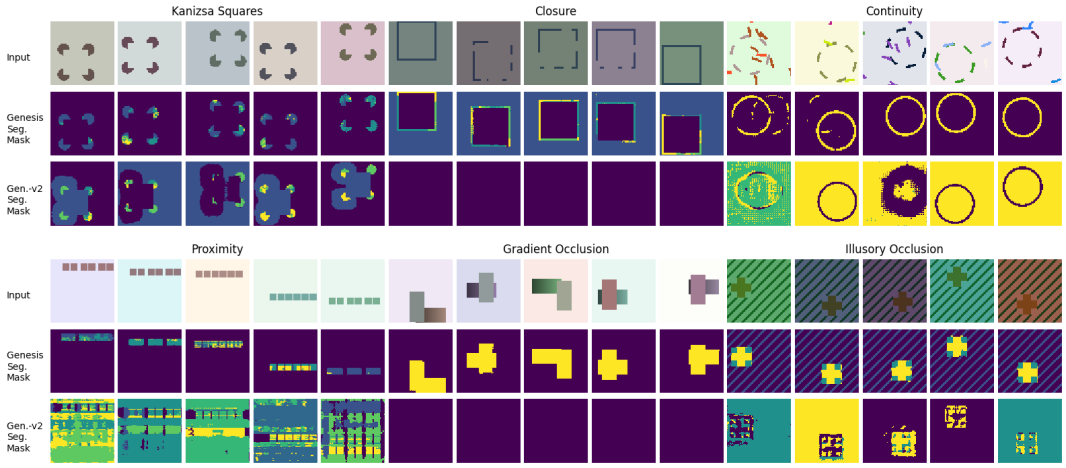


Figure 10: **Control Model Segmentation Mask Examples.** The first row shows inputs, the second row shows Genesis segmentation masks, and the third row shows Genesis-v2 segmentation masks. Randomly selected examples from the **Kanizsa Squares**, **Closure**, **Continuity**, **Proximity**, **Gradient Occlusion**, and **Illusory Occlusion** datasets. The specific color assignment to different object identities is arbitrary (for example, whether the model assigns the identity represented by yellow as the background, or purple, is not a meaningful distinction).

C.2. Implementation Details And Convergence

To ensure that the baseline failure is not due to an implementation error or a failure in convergence, we take two precautions: 1) We follow the official implementation of Genesis and Genesis-v2 (Engelcke et al., 2022; 2020a;b) (GPL v3 license); 2) We verify convergence by visualizing model reconstructions on a separate validation set with samples that were not shown during training 11. Furthermore, we follow the hyperparameter choices reported in the original papers ($5e^5$ training iterations with batch size 64, $K = 5$). Models were trained using an RTX 4090 GPU. Genesis-v2 trained for approximately 30 GPU-hours per dataset, while Genesis trained for approximately 20 GPU-hours per dataset, for a combined total of approximately 300 GPU-hours.

C.3. Noisy AE/VAE Control Experiment Implementation Details

The noise standard deviation applied to the reconstructions is 0.3235, the mean of the best noise values for the AE and VAE. For clustering we used Agglomerative Clustering with the Euclidean metric and Ward linkage.

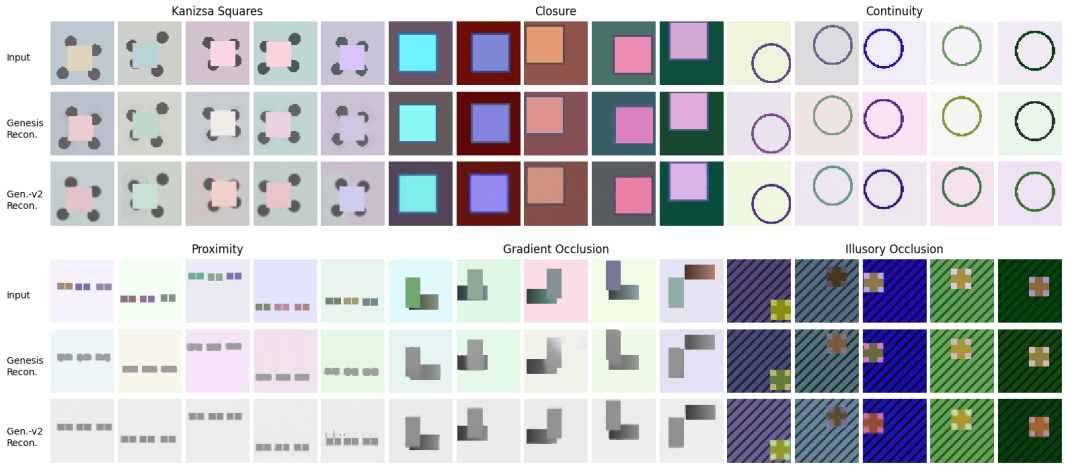


Figure 11: **Control Model Reconstruction Examples.** The first row shows inputs, the second row shows Genesis reconstructions, and the third row shows Genesis-v2 reconstructions. Randomly selected examples from the **Kanizsa Squares**, **Closure**, **Continuity**, **Proximity**, **Gradient Occlusion**, and **Illusory Occlusion** validation datasets. The validation samples come from the same distribution as the training data, but the specific samples are never seen in training.

D. Clustering

We used Agglomerative Clustering (Pedregosa et al., 2011) with the Euclidean metric and Ward linkage. We selected the desired number of clusters on an image-by-image basis. The desired number of clusters is the number of co-varying parts in the visual scene. For example, for an image in the GG Proximity dataset, if it contains one group of 6 squares, then the desired number of clusters is 2, corresponding to the co-varying squares and the background; if it contains 3 groups of 2 squares, then the desired number of clusters is 4, corresponding to the three groups of co-varying squares and the background.

Prior to applying the agglomerative clustering algorithm, we normalize $\Delta\tilde{\mathbf{X}}$ along its last dimension, so the vectors (with a size of $[\mathrm{C} \times (N-1)]$) have the same norm.

D.1. Comparison of Different Clustering Algorithms

We conducted a control experiment to test whether the specific clustering we used is a critical part of why LNS works. We tested three clustering algorithms: Agglomerative Clustering ((Pedregosa et al., 2011) with the Euclidean metric and Ward linkage; Agglomerative Clustering with the Euclidean metric and Complete linkage; and K-means clustering. We find that while the choice of clustering algorithm can affect results (Ward linkage outperforms Complete linkage), the specific algorithm is not a critical part of LNS (K-means achieves similar or better performance than Agglomerative Clustering with the Ward linkage). This result supports the conclusion that the key intuition behind LNS is that it allows the model to extract a meaningful clusterable representation, that can then be clustered using various clustering algorithms.

Model	Kanizsa	Closure	Continuity	Proximity	Gradient Occ.	Illusory Occ.
AE+Agg	0.859 $\pm$ 0.008	0.766 $\pm$ 0.008	0.552 $\pm$ 0.008	**0.996** $\pm$ 0.001	**0.926** $\pm$ 0.009	0.994 $\pm$ 0.002
VAE+Agg	**0.871** $\pm$ 0.005	0.795 $\pm$ 0.009	**0.593** $\pm$ 0.012	0.943 $\pm$ 0.010	0.918 $\pm$ 0.002	0.974 $\pm$ 0.003
AE+KMeans	0.854 $\pm$ 0.003	0.803 $\pm$ 0.001	0.553 $\pm$ 0.004	0.995$\pm$ 0.000	0.922 $\pm$ 0.006	**0.998** $\pm$ 0.000
VAE+KMeans	0.866 $\pm$ 0.003	**0.852** $\pm$ 0.006	0.589 $\pm$ 0.006	0.921 $\pm$ 0.005	0.924 $\pm$ 0.003	0.986 $\pm$ 0.001
AE+Agg'	0.557 $\pm$ 0.007	0.296 $\pm$ 0.006	0.322 $\pm$ 0.004	0.929$\pm$ 0.004	0.702 $\pm$ 0.004	0.520 $\pm$ 0.027
VAE+Agg'	0.457 $\pm$ 0.008	0.324 $\pm$ 0.004	0.302 $\pm$ 0.004	0.752$\pm$ 0.012	0.594$\pm$ 0.009	0.438$\pm$ 0.006

Table 2: Model results for different clustering algorithms (ARI $\pm$ standard error of the mean). Agg = the agglomerative clustering algorithm from `sklearn` with `metric` being "euclidean" and `linkage` being"ward". Agg' = the agglomerative clustering algorithm from `sklearn` with `metric` being "euclidean" and `linkage` being "complete". KMeans = the K-means algorithm from `sklearn`. These scores are obtained for the optimal noise level and number of samples found for Agg.

E. LNS Implementation Details

E.1. GG Datasets

Here we introduce the design of the GG datasets. Each image of the datasets contains several parts including one or several (groups of) objects and a background. In the training dataset, the pixel values belonging to the same part co-vary with each other in the same channel while the pixel values belonging to different parts take independent values in the same channels. Such a scheme is expected to abstract, at a high level, the statistical structure of realistic visual scenes, which is based on basic optics:

$$L = I \times R, \tag{11}$$

where L is luminance, I is illuminance, and R is reflectance. The rationale behind the above-mentioned relationship of different pixels is explained below.

The images in the datasets all have 3 channels. For simplicity, let us only consider one of the channels. The pixel values of one channel are the luminances of the reflected light that has the same wavelength. If two pixels belong to the same object, the ratio between their reflectances is fixed. This is because, in real life, the reflectance ratio between the different parts of the same object is fixed, as it is determined by the material. On the other hand, the reflectances of two pixels belonging to two different objects that can appear in a visual scene are independent. Furthermore, we assume that all the pixels of one object are exposed to the same lighting condition, namely the same illuminance, which is a reasonable simplifying assumption for simple visual scenes in real life. With the above, one can draw the conclusion that the luminances (pixel values) of different pixels belonging to the same object should keep a fixed ratio between each other throughout the training dataset, the luminances (pixel values) of different pixels belonging to different objects should be independent. Moreover, the pixel values of the same pixel in different channels are independent. Because the object might be in different lighting conditions in different visual scenes, and the illuminances of different channels under different lighting conditions should be independent.

Here, we describe how the different GG datasets are constructed. We encourage the reader to revisit Figures 2 and 3 while reading the following for a better understanding.

Kanizsa Squares contains three parts: a square, four circles, and a background. The four circles always have the same pixel values. Their relative positions are fixed in the testing dataset but not in the training dataset. In the testing dataset, although the square and the background have the same color, the model is expected to separate these two parts since it is primed by the training samples to do so. Such expectation is reflected in the segmentation masks (Figure 3).

Closure contains two parts, one outlined square and a background. In the testing dataset, the interior of the outlined square and the background share the same pixel values, and the outlines of the squares are sometimes occluded partially. The pixel value ratio of the outline and the square interior is kept the same as in the training dataset. The model is expected to perceive the squares together with the outlines.

Continuity contains two parts in the training dataset, a circle and a background. In the testing dataset, the circle is

fragmented, and several randomly generated fragments are placed at arbitrary positions. The model should perceive the existence of a circle in the testing dataset despite these disturbances.

Proximity contains either 2 parts or 4 parts. In the former case, it contains a background one group of 6 squares with an interspace of 1 pixels. In the latter case, it contains a background and 3 groups of 2 squares. The interspaces between different squares within the same group are 1 pixel wide, and the interspaces between different groups are at least 3 pixel wide so that the two cases cannot be confounded. In the testing dataset, all the squares have the same color. However, the model is expected to group different squares together correctly given the interspaces.

Gradient Occlusion contains three parts, one tall rectangle with homogeneous color, one wide rectangle with gradient color, and a background. In the testing dataset, the homogeneously-colored rectangle always occludes the gradient-colored rectangle. However, the model is expected to group together the pixels belonging to the gradient-color rectangle, even if they do not have the same pixel values and are sometimes separated by the homogeneously-colored rectangle.

Illusory Occlusion contains 2 parts, one striped background, and one striped square. In the testing dataset, part of the background and the foreground object have the same pixel value. However, the model is expected to identify the foreground-background structure.

For code to generate the GG datasets, please refer to `create_datasets.py` in our git repository: https://github.com/ZhengqingUUU/LatentNoiseSegmentation. The training set in our experiments contained $30,000$ images, the validation set contained 300 images that were not in the training set, and the test set contained 100 images.

E.2. Deep Neural Networks

Hyperparameter	Value
Optimizer	Adam
Learning rate	$5e^{-5}$
Adam betas	(0.9, 0.999)
Batch size	64
Model latent dim	15(GG)/50(CelebA)
Training iterations	$1.1e^{6}$
AE loss function	MSE
VAE loss function	GECO* (Rezende & Viola, 2018)]
GECO g_goal	0.0006(GG)/0.0125 (CelebA)

Table 3: Autoencoder and VAE hyperparameters used in the experiments. * Adapted from the implementation by Engelcke et al. (2022).

	Layer	Output Dimension	Kernel size	Stride	Padding	Activation function
Enc	Conv2d	32	3	1	1	ReLU
	Conv2d	64	4	2	1	ReLU
	Conv2d	64	4	2	1	ReLU
	Conv2d	128	4	2	1	ReLU
	Conv2d	128	4	2	1	ReLU
	Conv2d	256	4	1	0	ReLU
	Linear	128	-	-	-	ReLU
	Linear	15	-	-	-	-
Dec	Inverse of the encoder					

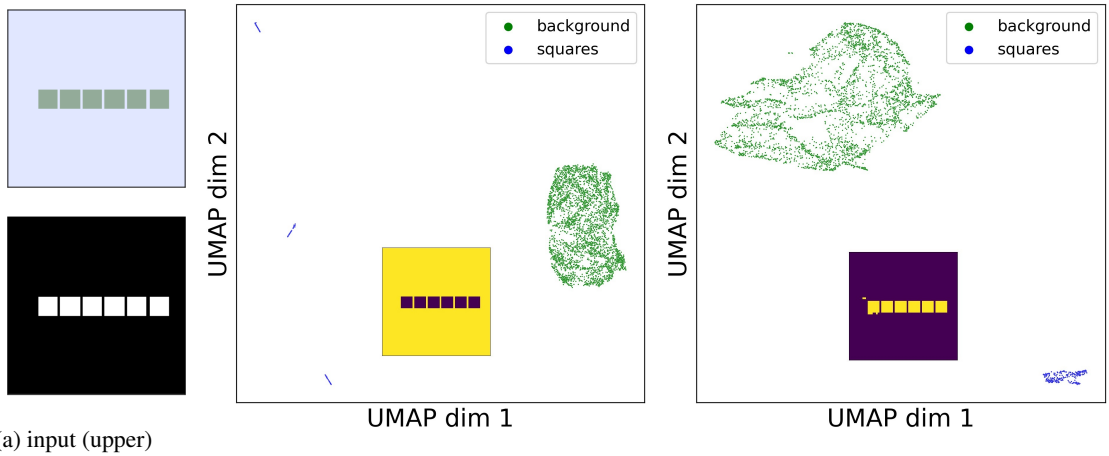
Table 4: Autoencoder and VAE architecture.

F. Clusterability of Representations

F.1. AEs, but not VAEs, Perceive the Wrong Number of Clusters

In our Segmentation algorithm, the expected number of clusters is provided on an image-by-image basis as *a priori* information for the model. In most cases, the expected number of clusters coincides with the actual clusters spontaneously formed by $\Delta\tilde{\mathbf{X}}_i$, which can be qualitatively verified with a UMAP visualization (McInnes et al., 2018). However, there are exceptions where $\Delta\tilde{\mathbf{X}}_i$ is not readily grouped into the expected numbers of clusters. We observe such exceptions in segmentation using AE on the GG Proximity dataset (when there only exists 1 group of 6 squares) and the GG Illusory Occlusion dataset. Here, we present the UMAP visualization for them using one example per dataset. Note that while not shown here, we observe that the qualitative pattern of the AE being unable to capture the expected number of clusters generally holds for these two cases.

In Figures 12b and 12c, we show the distribution of the vectors corresponding to each pixel in the $[C \times (N-1)]$-dimensional space where clustering is performed, when the models are confronted with an image from the GG Proximity dataset with 1 group of 6 squares. The image and the expected target mask are shown in Figure 12a. In this case, the AE tends to separate the pixels belonging to the squares into 3 clusters, perhaps as a result of being affected by training samples where there exist 3 groups of 2 squares; while the VAE tends to perceive the right number of clusters. However, when given the expected number of clusters, which is 2 in this case, the AE managed to nevertheless yield the correct segmentation mask (indicated by the inset in Figure 12b).



(a) input (upper) target(lower)

(b) UMAP for AE

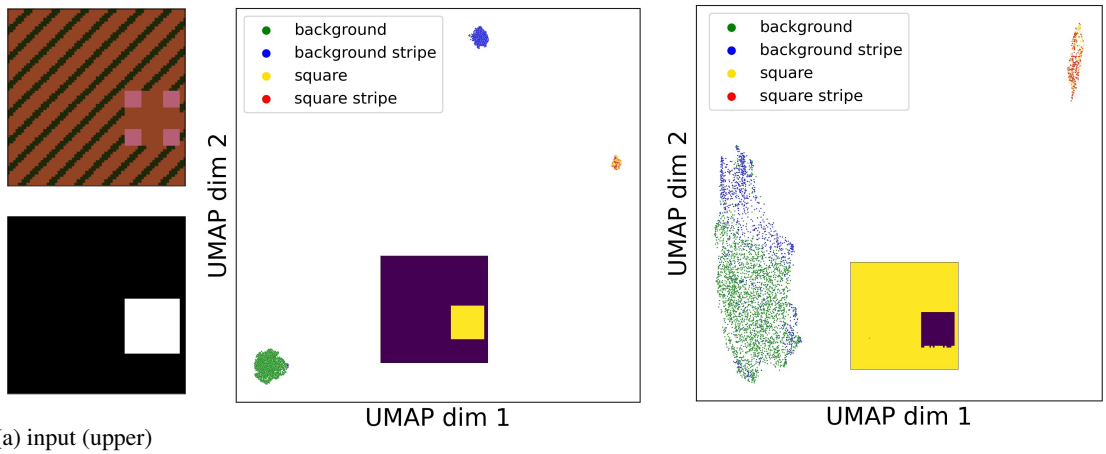
(c) UMAP for VAE

Figure 12: The AE perceives the wrong numbers of clusters for the GG Proximity dataset with 1 group of 6 squares. (a) contains the input image and the expected mask. (b) and (c) show the visualization of $\Delta\tilde{\mathbf{X}}^i$. Each dot corresponds to a pixel in the image. The visualization is made by using UMAP to reduce the $[C \times (N-1)]$-dimensional vector corresponding to each pixel to 2 dimensions. The color of the dots indicates which part the pixels correspond to. The insets are the corresponding segmentation masks generated by the AE/VAE model.

A similar situation happens also for the GG Illusory Occlusion dataset. As shown in Figure 13, the AE tends to separate the background from the background stripes. We would expect these two parts to be grouped together, since they always co-vary with each other. Nonetheless, given the expected number of clusters, the AE manages to ultimately segment correctly.

F.2. Latent Noise Segmentation Forms Clusterable Representations

We further show the UMAP visualization of $\Delta\tilde{\mathbf{X}}_i$ to demonstrate that LNS forms a meaningful clusterable representation of the pixels in general. $\Delta\tilde{\mathbf{X}}_i$ is first reduced to $imag_x \times imag_y \times 2$ with UMAP performed on its last two dimensions, which contains $[C \times (N-1)]$-dimensional representations for each pixels. The resulting UMAP representation, accordingly, contains 2-dimensional representations for all the pixels, which we visualize in Figure 15. The position of each dot shows the 2-dimensional representation for each pixel, and we differentiate between different parts of the images with different colors. The UMAP results in in Figure 15 are generated while segmenting the samples in Figure 4.



(a) input (upper) target(lower)

(b) UMAP for AE

(c) UMAP for VAE

Figure 13: The AE perceives the wrong number of clusters for the GG Illusory Occlusion Dataset. Please refer to the caption of Figure 12.

F.3. Model Test Reconstructions

To support our control experiment, we also visualize the model output reconstructions (Figure 14). In accordance with the control experiments, we find that the outputs in some cases contain some information about the identity of the objects, even when it does not veridically exist in the input image. However, as reported in Table 1, the amount of information in the reconstructions is not as high as it is when using Latent Noise Segmentation. This is because of the tension between reconstructing an image well (which would lead to poor segmentation without LNS in test samples), and segmenting the image components based on their color values. In other words, the fact that the model is still able to segment in some cases can be understood as an artefact caused by the small size of our model and imperfect reconstruction performance. The training process implicitly optimizes toward the success of LNS segmentation, but against the success of direct reconstruction segmentation.

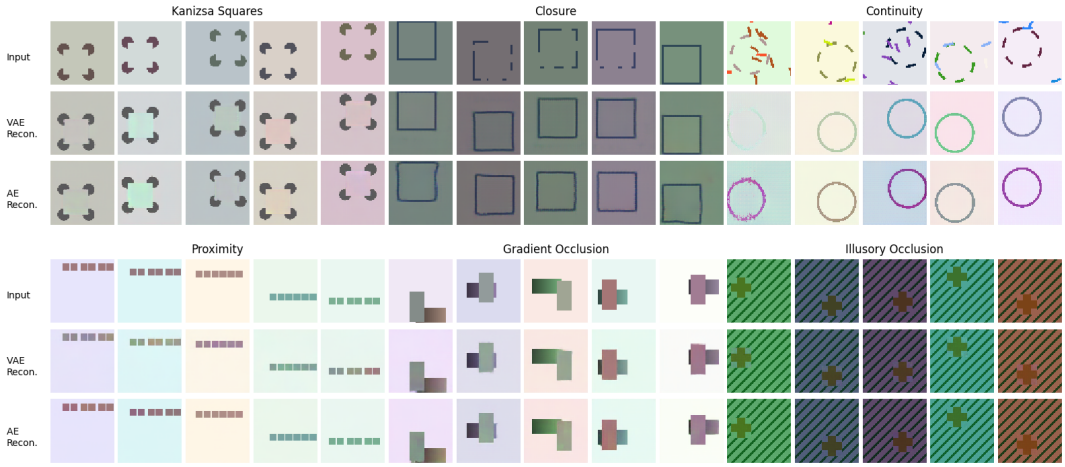
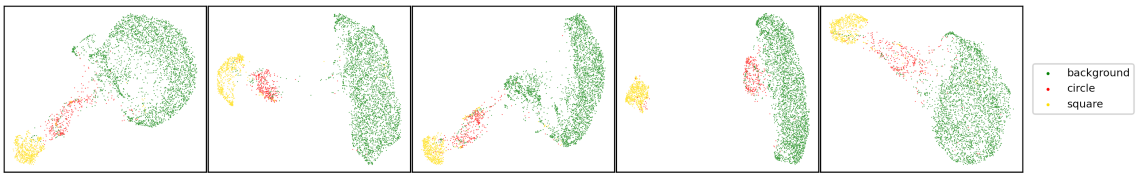
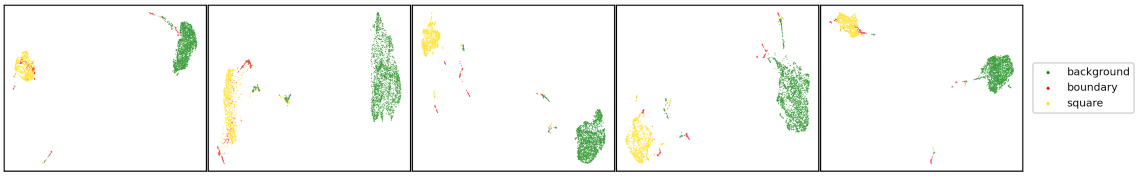


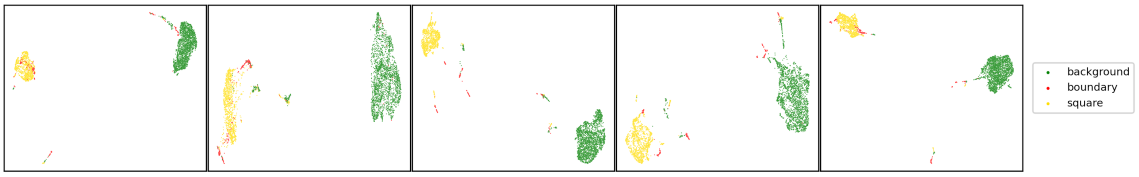
Figure 14: **AE and VAE Model Reconstruction Examples.** The first row shows inputs, the second row shows VAE reconstructions, and the third row shows AE reconstructions. Randomly selected examples from the **Kanizsa Squares**, **Closure**, **Continuity**, **Proximity**, **Gradient Occlusion**, and **Illusory Occlusion** test datasets.



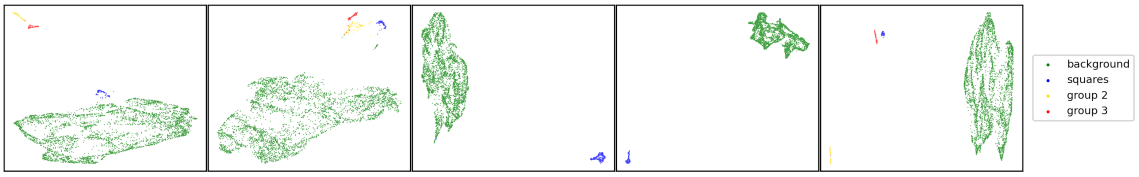
(a) Model: VAE, dataset: Kanizsa Squares



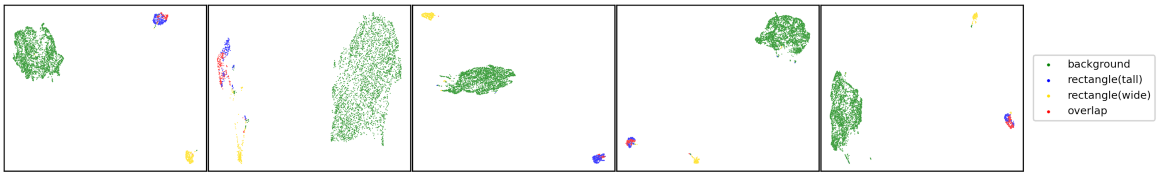
(b) Model: VAE, dataset: Closure



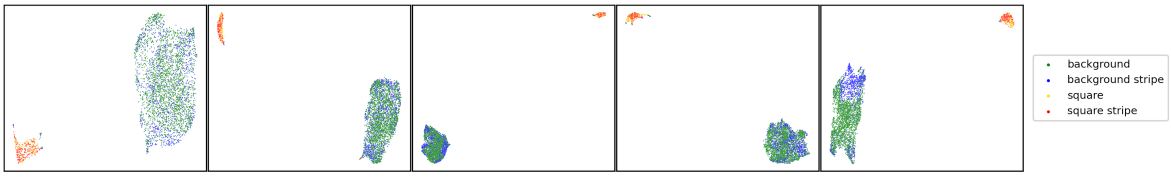
(c) Model: VAE, dataset: Continuity



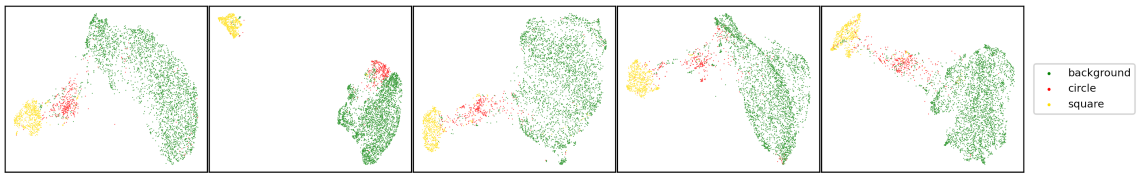
(d) Model: VAE, dataset: Proximity



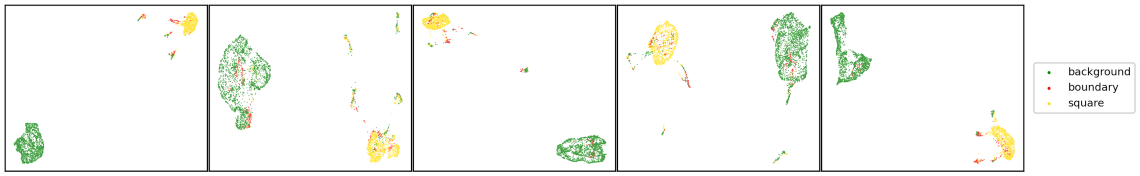
(e) Model: VAE, dataset: Gradient Occlusion



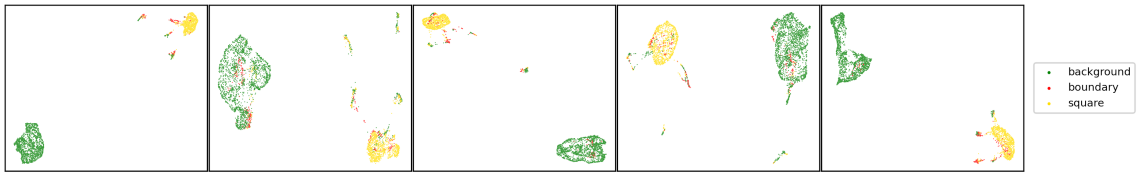
(f) Model: VAE, dataset: Illusory Occlusion



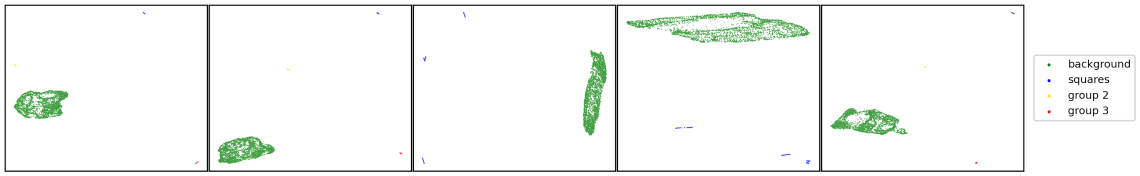
(g) Model: AE, dataset: Kanizsa Squares



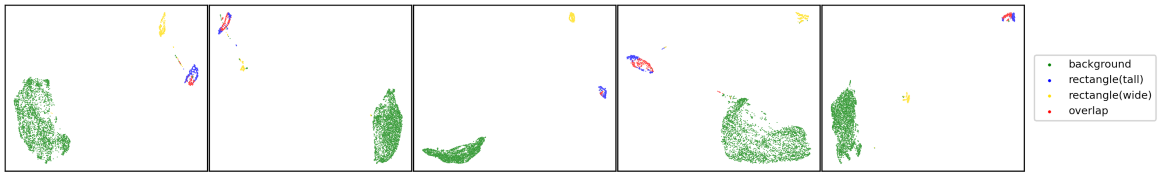
(h) Model: AE, dataset: Closure



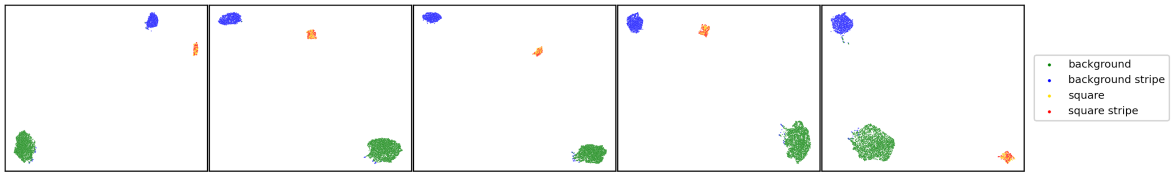
(i) Model: AE, dataset: Continuity



(j) Model: AE, dataset: Proximity



(k) Model: AE, dataset: Gradient Occlusion



(l) Model: AE, dataset: Illusory Occlusion

Figure 15: Visualization of $\Delta\tilde{\mathbf{X}}^i$ with UMAP.

F.4. Additional model outputs

Here, we plot all model outputs for our testing set for a single seed for both model architectures (AE and VAE) to qualitatively confirm that our models capture the meaningful and relevant desired properties in our datasets, and that our qualitative model performance holds in general, and not only in a few hand-picked examples.

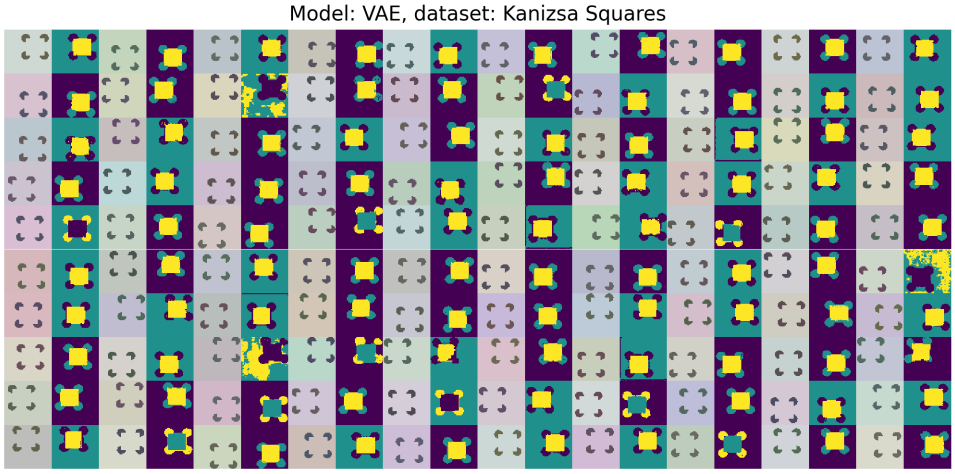


Figure 16: **All Variational Autoencoder outputs for a single seed of the GG Kanizsa Squares dataset.** Inputs and outputs are displayed as horizontal pairs, with the input being the left image, and the model output being the right image.

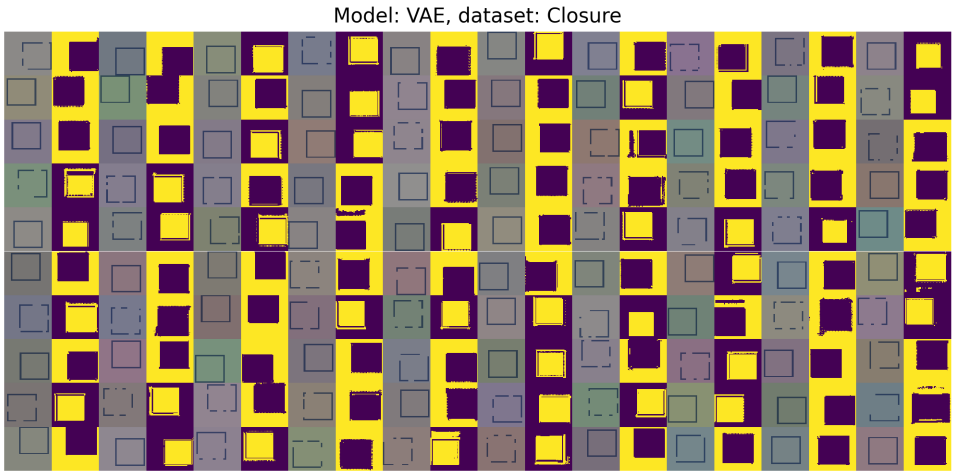


Figure 17: **All Variational Autoencoder outputs for a single seed of the GG Closure dataset.** Inputs and outputs are displayed as horizontal pairs, with the input being the left image, and the model output being the right image.

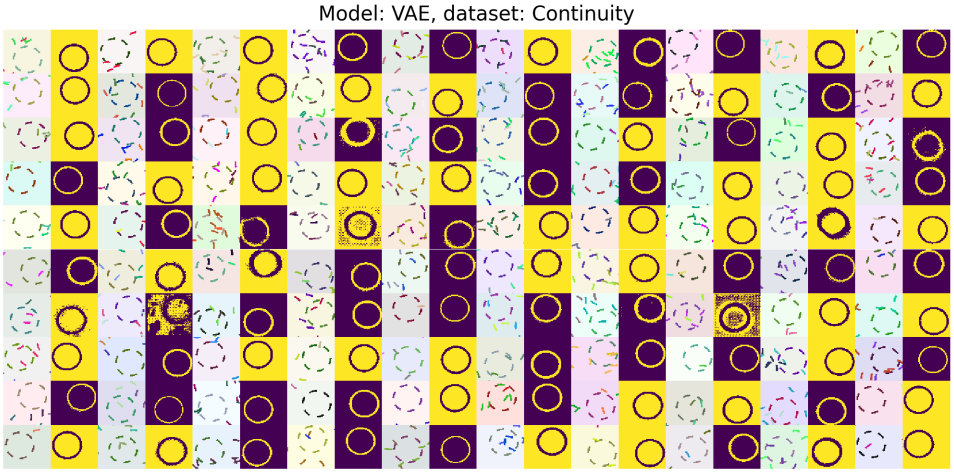


Figure 18: **All Variational Autoencoder outputs for a single seed of the GG Continuity dataset.** Inputs and outputs are displayed as horizontal pairs, with the input being the left image, and the model output being the right image.

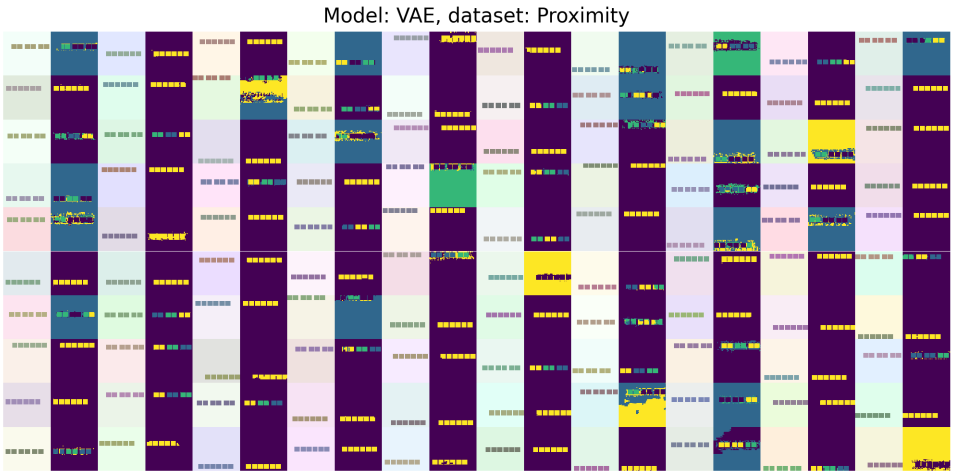


Figure 19: **All Variational Autoencoder outputs for a single seed of the GG Proximity dataset.** Inputs and outputs are displayed as horizontal pairs, with the input being the left image, and the model output being the right image.

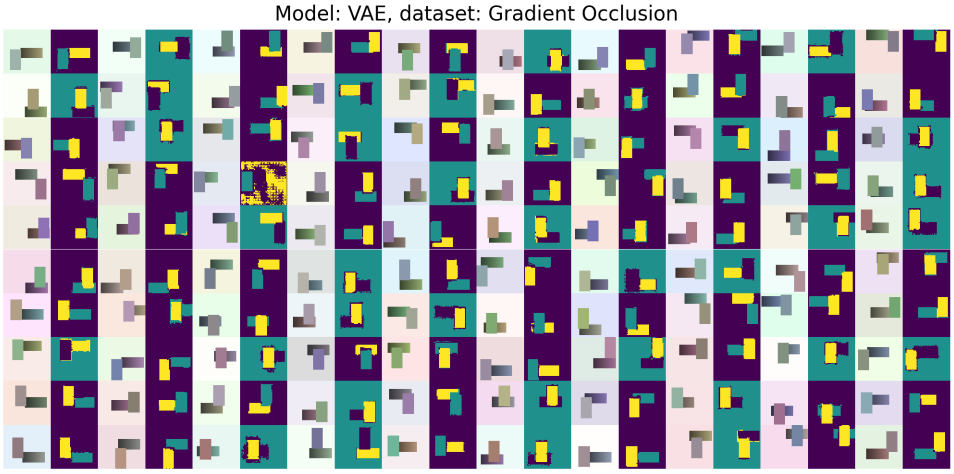


Figure 20: **All Variational Autoencoder outputs for a single seed of the GG Gradient Occlusion dataset.** Inputs and outputs are displayed as horizontal pairs, with the input being the left image, and the model output being the right image.

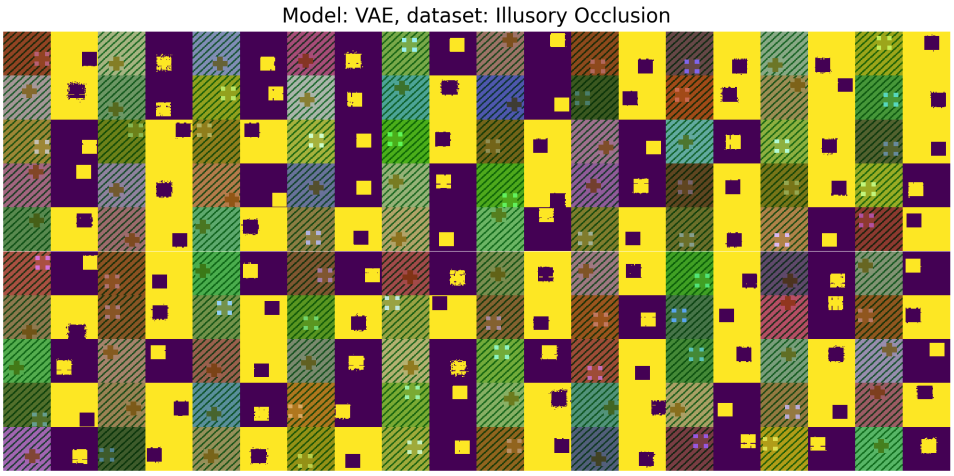


Figure 21: **All Variational Autoencoder outputs for a single seed of the GG Illusory Occlusion dataset.** Inputs and outputs are displayed as horizontal pairs, with the input being the left image, and the model output being the right image.

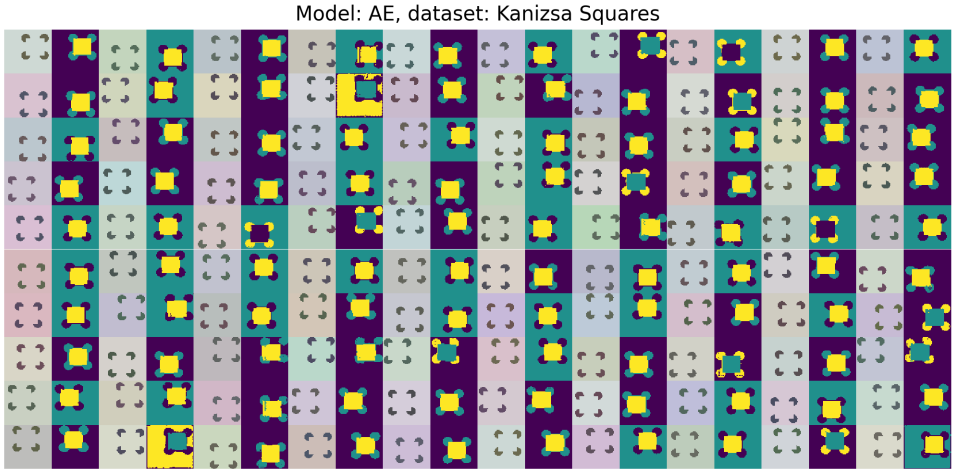


Figure 22: **All Autoencoder outputs for a single seed of the GG Kanizsa Squares dataset.** Inputs and outputs are displayed as horizontal pairs, with the input being the left image, and the model output being the right image.

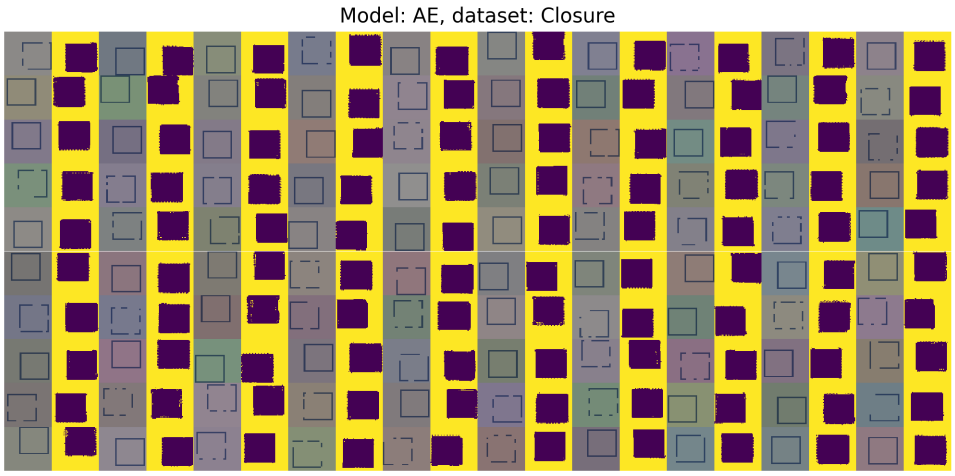


Figure 23: **All Autoencoder outputs for a single seed of the GG Closure dataset.** Inputs and outputs are displayed as horizontal pairs, with the input being the left image, and the model output being the right image.

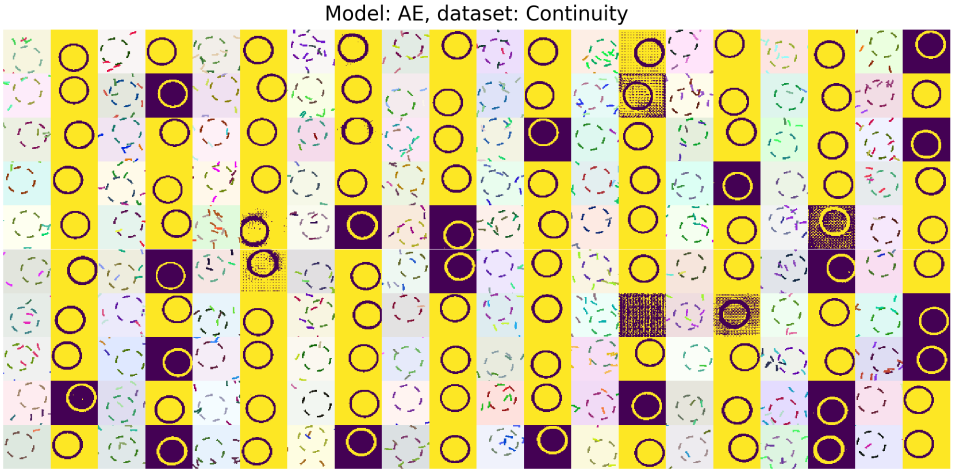


Figure 24: **All Autoencoder outputs for a single seed of the GG Continuity dataset.** Inputs and outputs are displayed as horizontal pairs, with the input being the left image, and the model output being the right image.

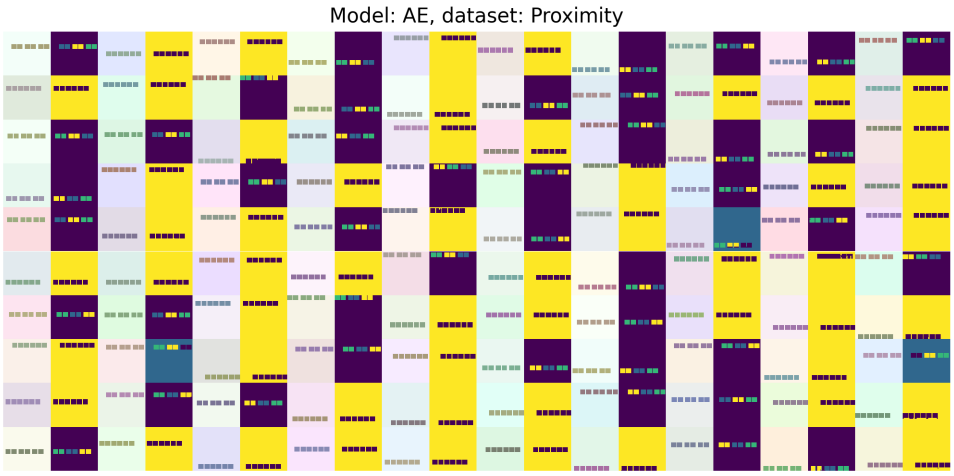


Figure 25: **All Autoencoder outputs for a single seed of the GG Proximity dataset.** Inputs and outputs are displayed as horizontal pairs, with the input being the left image, and the model output being the right image.

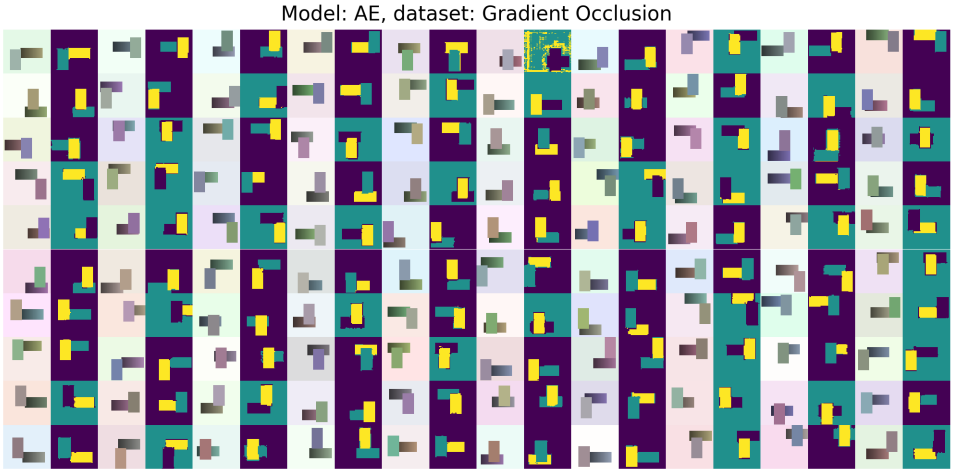


Figure 26: **All Autoencoder outputs for a single seed of the GG Gradient Occlusion dataset.** Inputs and outputs are displayed as horizontal pairs, with the input being the left image, and the model output being the right image.

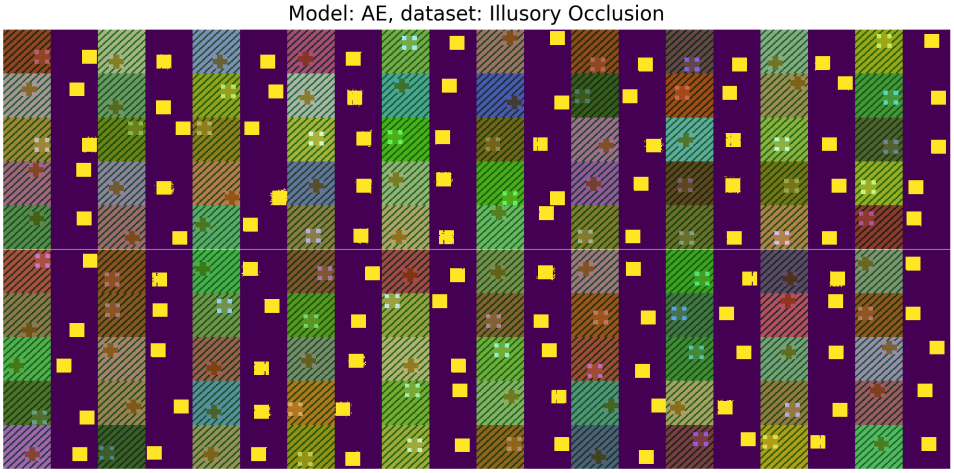


Figure 27: **All Autoencoder outputs for a single seed of the GG Illusory Occlusion dataset.** Inputs and outputs are displayed as horizontal pairs, with the input being the left image, and the model output being the right image.