Representation Learning with Hierarchical VAEs

Roshni Sahoo (rsahoo@mit.edu)
MIT 6.804 Computational Cognitive Science Final Project

Abstract

Variational autoencoders have thus far achieved success in generating disentangled feature representations. However, current VAE methods ignore high-entropy image statistics, such as textures. We propose a hierarchical VAE architecture for capturing low and high-level features of an image. We evaluate our representations qualitatively by generating reconstructions of the input image and visualizing traversals of the latent space and quantitatively by assessing how well we can regress global features from the latent representation. **Keywords:** hierarchical VAEs; compositional learning.

Introduction

Artificial intelligence can be distinguished from biological intelligence in many ways. Supervised machine learning models require large amounts of labeled training data to gain proficiency at a particular task. Furthermore, machine learning models often do not generalize well to data from outside their training distribution. In other words, these models can suffer from overfitting to the training distribution and specialization to a particular task (Marcus, 2018; Bengio, Courville, & Vincent, 2013). In contrast, humans are not only trained specifically for individual tasks but also accumulate knowledge without supervision. The benefit of the human learning approach is that acquired knowledge is reusable and generalizable to many different tasks.

The ultimate motivation for this work is to inform how to develop models that can learn efficiently and are robust to variation. As part of this goal, we focus on learning transferable representations. Learning transferable representations will allow models to learn more efficiently in the future because if models can identify salient features of perceptual inputs, there will be a reduced need for large, annotated datasets that are currently essential in supervised learning. Furthermore, we envision that learning transferable representations will allow machine learning models to be more robust to variation because the representation will be reusable and support reasoning in a changed context.

Variational autoencoders (VAEs) have thus far been a promising approach to learning feature representations from images. However, variational autoencoders often ignore texture, preventing them from generating meaningful feature representations on images containing textured patterns. To generate more informative feature representations, we propose a hierarchical variational autoencoder approach to capture low and high-level features of an image.

Background: Variational Autoencoders

Suppose that N observed i.i.d. images $\mathbf{x}$ are generated by some random process involving the unobserved random variable $\mathbf{z}$. A VAE is a generative approach that aims to learn the joint distribution of images $\mathbf{x}$ and their latent generating factor $\mathbf{z}$ (Kingma & Welling, 2014; Rezende, Mohamed, & Wierstra, 2014). In practice, the VAE consists of two main components, an encoder and a decoder. The encoder is a variational inference network that maps a datapoint $\mathbf{x}$ to a Gaussian distribution over the possible values of $\mathbf{z}$. The decoder, given a latent representation $\mathbf{z}$, generates a distribution over the possible corresponding values of $\mathbf{x}$. More precisely, the encoder learns a recognition model $q_\phi(\mathbf{z}|\mathbf{x})$, where ϕ denotes its parameters, to approximate the true posterior. The decoder learns a generative model $p_\theta(\mathbf{x}|\mathbf{z})$. See Figure 1 for a diagram of a VAE.

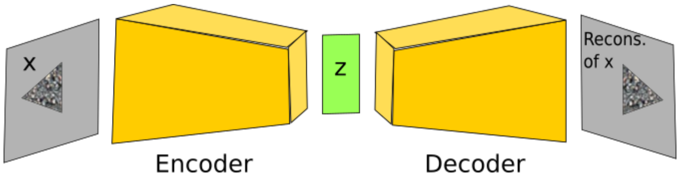

Figure 1: Components of a Variational Autoencoder (VAE)

As a proxy for optimizing the intractable marginal likelihood $p(\mathbf{x})$, we optimize the evidence lower bound (ELBO).

$$\begin{aligned}\log p(\mathbf{x}) &\geq \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - D_{KL}(q_\phi(\mathbf{z}|\mathbf{x})||p(\mathbf{z}))\\ &= \mathscr{L}(\theta,\phi;\mathbf{x})\end{aligned}$$

The first term on the right hand side of the inequality can be interpreted as a reconstruction cost: Typically the likelihood $p_\theta(\mathbf{x}|\mathbf{z})$ is parameterized as a product of independent Gaussians with fixed variance σ^2 and mean $\tilde{\mathbf{x}}$, in which case it becomes $||\mathbf{x}-\tilde{\mathbf{x}}||^2/2\sigma^2$. Here we choose $\sigma = \frac{1}{\sqrt{2}}$ so that the term becomes mean squared error, as is common in the literature (Higgins et al., 2016; Kim & Mnih, 2018). The KL divergence term regularizes the approximate posterior $q_\phi(\mathbf{z}|\mathbf{x})$ by pressuring it towards the prior, which is typically a standard Gaussian distribution. From a cost function perspective, we aim to minimize $-\mathscr{L}(\theta,\phi;\mathbf{x})$. We will refer to the VAE loss function defined below, and we can understand it as the sum of the reconstruction loss (the first term) and the KL loss (the second term)

$$L_{VAE} = ||\mathbf{x}-\tilde{\mathbf{x}}|| + D(q_\phi(\mathbf{z}|\mathbf{x})||p(\mathbf{z})).$$

Investigative Questions

We seek to understand whether creating a hierarchy of variational autoencoders yields more informative feature representations on textured images than a standard variational autoencoder. These features may be high-level or low-level. An example of a high-level feature in the image in Figure 2 is the triangle shape; a low-level feature of the image is the cloth

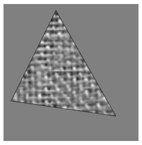

Figure 2: Textured Image with Low and High-Level Features

texture in the interior of triangle. We hypothesize that a VAE will struggle to learn meaningful feature representations on images that contain textured patterns and background color equal to the mean color of the texture. The reconstruction loss for a VAE is a pixel-based loss (often the Mean Squared Error between the initial image and the reconstruction). We hypothesize that the standard variational autoencoder will not have enough latent capacity to reconstruct the texture at the pixel level so will instead output a monochromatic reconstruction, namely the average color of the input image.

We propose a Hierarchical VAE architecture where we use a Texture VAE pretrained on small patches of the texture to generate a latent representation of the image and a Global VAE to study high-level features of the image. The Texture VAE aims to capture local statistics of the image. After generating this latent representation, we train the Global VAE on the latent representations with the goal of capturing the higher level features of the image. We aim to see whether this allows for better reconstructions of the original image and a more informative latent representation.

Hierarchical VAE

A Hierarchical VAE can be viewed as a series of VAEs stacked on top of each other. For our purposes, we have a hierarchy of two latent variables $\mathbf{z} = \{\mathbf{z}_1, \mathbf{z}_2\}$ in addition to the observed variable $\mathbf{x}$.

A schematic diagram of our hierarchical model is depicted in Figure 3. The main components of our model include a Texture VAE, which takes in 7×7 patches of the input image $\mathbf{x}$. The Texture VAE is pretrained on textured images so that it learns a representation of local features. To train the Hierarchical VAE, we pass patches of our input image $\mathbf{x}$ to the encoder of the Texture VAE and infer the latent representation of the patches of $\mathbf{x}$. We concatenate the outputted latent representations of the patches across space to form a spatial feature map $\mathbf{z}_1$. We generate these latent representations for every image in our training set. Next, we train the Global VAE on the latent representations $\mathbf{z}_1$. We use the decoder of the Texture VAE to generate $\tilde{\mathbf{x}}$ from the reconstruction $\tilde{\mathbf{z}}_1$. We inspect whether the Global VAE can learn a latent representation of the image that encodes salient high-level features of the image, such as the position of the triangle.

Methods

To investigate whether we can learn high-level features by imposing hierarchy, we use a simple task: reconstructing images containing a textured triangle where the background color is the mean color of the texture. We create a dataset for this specific task, train a Standard VAE and a Hierarchical VAE on this dataset, and evaluate the results qualitatively and quantitatively.

Dataset Generation

We generate a dataset of cropped textured images using the ETHZ Synthesizability Dataset (Dai, Riemenschneider, & Van Gool, 2014). The images in the ETHZ Synthesizbility Dataset are 300×300 images, which we resize to 80×80 and take random 7×7 crops from each texture. These images will be used to pretrain the Texture VAE. For our main experiment, we select 3 high contrast textures and generate random crops from these images. We also generate the Textured Triangle Dataset, which consists of 64×64 RGB images containing a textured triangle and the background of the image is the mean color of the texture. An example image is shown in Figure 2. These images are generated by creating triangle mask images using the Spriteworld Reinforcement Learning Environment (Watters, Matthey, Borgeaud, Kabra, & Lerchner, 2019) and multiplying these masks by a texture from the ETHZ Synthesizability Dataset (Dai et al., 2014). The position and size of the triangle varies in each image. After that we set the background of the image to the mean color of the textured triangle.

Pretraining the Texture VAE

We pretrain the Texture VAE on cropped textured images and train until the loss stabilizes. Figure 4 depicts the pretraining process. We set the latent dimension be 16 for the pretrained VAE. This means that a 7×7 patch is encoded in a 1×16 dimensional vector.

Training the Global VAE

We train the Global VAE on the Textured Triangle Dataset. We use the pretrained Texture VAE to generate the latent representation $\mathbf{z}_1$ of the 64×64 RGB input image $\mathbf{x}$. We generate a latent representation of the 7×7 patch that centered at each pixel; we extract patches of the images of kernel size 7×7 with a stride of 1 and a padding of 3. Since the latent dimension of the Texture VAE is 16, each patch of the image yields a 1×16 dimensional vector that contributes to the latent variable $\mathbf{z}_1$. In total, $\mathbf{z}_1$ has dimension $16 \times 64 \times 64$ for a single image.

The Global VAE encoder takes $\mathbf{z}_1$ as input and encodes it into a 1×16 dimensional vector, $\mathbf{z}_2$. The Global Decoder takes in $\mathbf{z}_2$ and generates a reconstruction of $\mathbf{z}_1$, $\tilde{\mathbf{z}}_1$. We note that this implies that the reconstruction loss for the Global VAE is computed between $\mathbf{z}_1$ and $\tilde{\mathbf{z}}_1$, and not the original image $\mathbf{x}$. We train the Global VAE for 40 epochs on latent representations $\mathbf{z}_1$.

We also note that our Hierarchical VAE is not trained end-to-end; we train the Texture VAE first, use it to generate the latent representations of the inputs, and solely train the Global VAE on these latents.

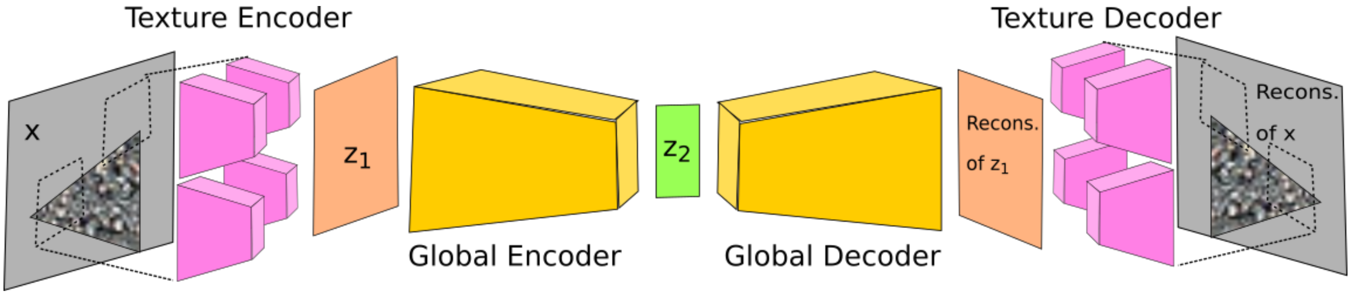

Figure 3: A Hierarchical VAE model for learning low-level and high-level features.

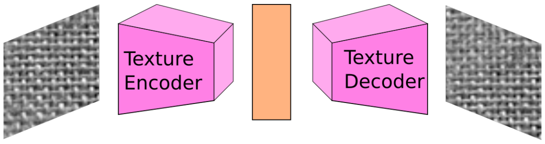

Figure 4: Pretraining the Texture VAE

In our evaluation of the Hierarchical VAE, we inspect $\mathbf{z}_2$ to see if the hierarchical VAE is capturing high-level features.

Training the Standard VAE

Training the Standard VAE is very similar to training the Global VAE, except that the Standard VAE takes in the RGB 64×64 image $\mathbf{x}$. Nevertheless, we use similar architectures for training the Standard VAE and the Global VAE. The latent dimension of the Standard VAE is also a 1×16 vector as in the Global VAE.

Evaluation

Our goal is to see whether the Hierarchical VAE architecture captures more structural information than a Standard VAE architecture. We can evaluate the methods qualitatively in two ways, by inspecting the reconstructions $\tilde{\mathbf{x}}$ of the input and the latent traversals. We can quantitatively assess whether the Standard VAE and Hierarchical VAE can encode high-level structural information by evaluating how accurately the position and size of the triangle can be regressed from their encodings of the input.

Reconstruction Quality

The simplest qualitative analysis is to check the reconstructions $\tilde{\mathbf{x}}$ created by the Standard VAE and the Hierarchical VAE. We see that the Hierarchical VAE can construct defined reconstructions of the textured triangle after 40 epochs of training, whereas the reconstructions of the Standard VAE appear to capture the average color of the image. Nevertheless, the reconstructions made by the Hierarchical VAE are not perfect– despite capturing the local texture, shape, location of the triangles, the background of all the images produced by the Hierarchical VAE are tinted red. We hypothesize that the bias toward red-colored backgrounds is due to imbalance in the color distribution of the training set.

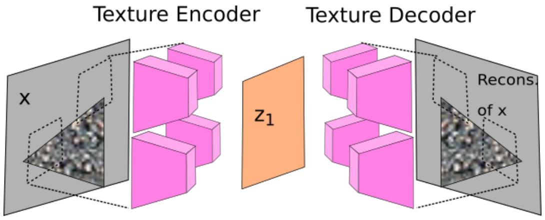

Figure 5: Reconstructing the textured triangle images using only the Texture VAE allows us to understand what our Hierarchical VAE reconstructions would look like if we had an error of 0 in reconstructing $\mathbf{z}_1$.

In Figure 6, we compare the input images to the Standard VAE reconstructions. In Figure 7, we compare the input images to the reconstructions made by the Hierarchical VAE in the second row; these images are made by passing an image $\mathbf{x}$ through the Hierarchical VAE pipeline depicted in Figure 3. In addition, in Figure 7, we also show the reconstructions made solely by using the Texture VAE. These reconstructions are made by passing patches of $\mathbf{x}$ to the Texture Encoder and then directly decoding the latent representation $\mathbf{z}_1$ using the Texture Decoder. The pipeline for this process is depicted in 5. We display the Texture VAE reconstructions to demonstrate what the best quality reconstruction of the original image $\mathbf{x}$ would resemble given a perfect reconstruction of $\mathbf{z}_1$ by the Global VAE. The quality of the Texture VAE reconstructions can be thought of as an upper bound on the quality of the reconstructions of the Hierarchical VAE.

Latent Traversals

To generate a latent traversal, we pass an input image $\mathbf{x}$ to the encoder of the VAE and generate the latent representation of the image. We can create a visualization of a traversal by adjusting entry i of $\mathbf{z}$ and keeping the other entries fixed. After that, we decode all of the variants of $\mathbf{z}$ to visualize the effect of changing the value at that particular dimension. For the Standard VAE, the latent representation is the 1×16 vector $\mathbf{z}$ that is shown in green in Figure 1. For the Hierarchical VAE, we use the 1×16 vector $\mathbf{z}_2$ shown in green in Figure 3, and we decode the variants of $\mathbf{z}_2$ by passing it through the Global Decoder and the Texture Decoder. The visualizations of the latent traversals were made by following the Medium

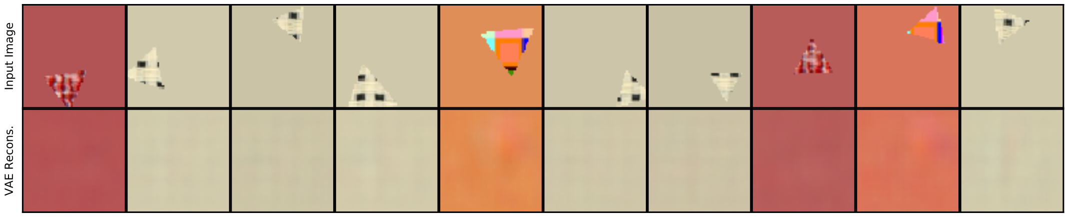

Figure 6: Reconstructions from Standard VAE after 40 Epochs of Training

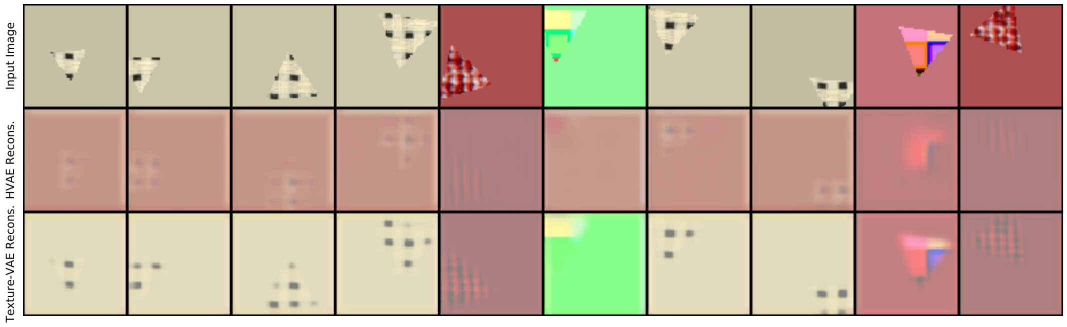

Figure 7: Reconstructions from Hierarchical VAE after 40 Epochs of Training

tutorial ”Learning Disentangled Features” and the associated code (Morton, 2018).

In Figure 8, we have the latent traversals of the Standard VAE. We observe that they show changes in the average color of the image as we vary particular dimensions of z. In Figure 9, we have the latent traversals of $\mathbf{z}_2$ of the Hierarchical VAE. When inspecting the latent traversals, we can qualitatively observe that the traversal from the Hierarchical VAE depicts more variation in the shape and size of the triangle. For example, when adjusting dimension 3 of $\mathbf{z}_2$ of the Hierarchical VAE, we see that the position of the fuzzy shape changes. Similarly, when adjust dimension 2 of $\mathbf{z}_2$ we see that the size of the textured part of the image changes.

Regression of Shape Position and Size

While our reconstructions and traversals qualitatively show that the Hierarchical VAE is encoding more meaningful information about the image and the Standard VAE, we also quantitatively assess this improvement by measuring and comparing how much information about the textured object is represented in the models' latent spaces. We train a simple regression model, a neural network with one hidden layer, to predict the position and size of the triangle given the latent representation of the image. Position and size of the triangle are high-level features of the image.

We create a training dataset $\{z_i, f_i\}_{i=1}^N$ of latent representations z and labels f that are tuples of the x-position, y-position, and scale. We note that the x-position and y-position lie in the range from $[0,64]$ and the scale is in the range $[0,100]$. A scale factor of 100 means that the shape takes up 100% of the image.

We encode an image and store the outputted the latent representation z_i. We aim to predict f_i from z_i. We use the MSE loss function to train the regression model, so we aim to minimize the squared difference between the predicted position and the target position and the predicted scale and the target scale. A diagram of the inference pipeline for regressing high-level features of an image with the Hierarchical VAE is shown in Figure 10. In Figure 11, we show the results of this experiment. After training the Hierarchical VAE and the Standard VAE for 40 epochs, we see that the Hierarchical VAE achieves approximately half error in position and size compared to the Standard VAE. This suggests that the Hierarchical VAE can better capture high-level features.

Challenges

We highlight some of the challenges we came across during the completion of this project.

1. *Developing a way of representing $\mathbf{z}_1$ of the hierarchical model.* We attempted many different techniques of building a latent representation of a Textured Triangle image from the outputs of the Texture VAE. Initially, we created a representation that consisted of non-overlapping patches of the original image. So, from a 64×64 image, we extracted patches with a kernel size of k and a stride of k.

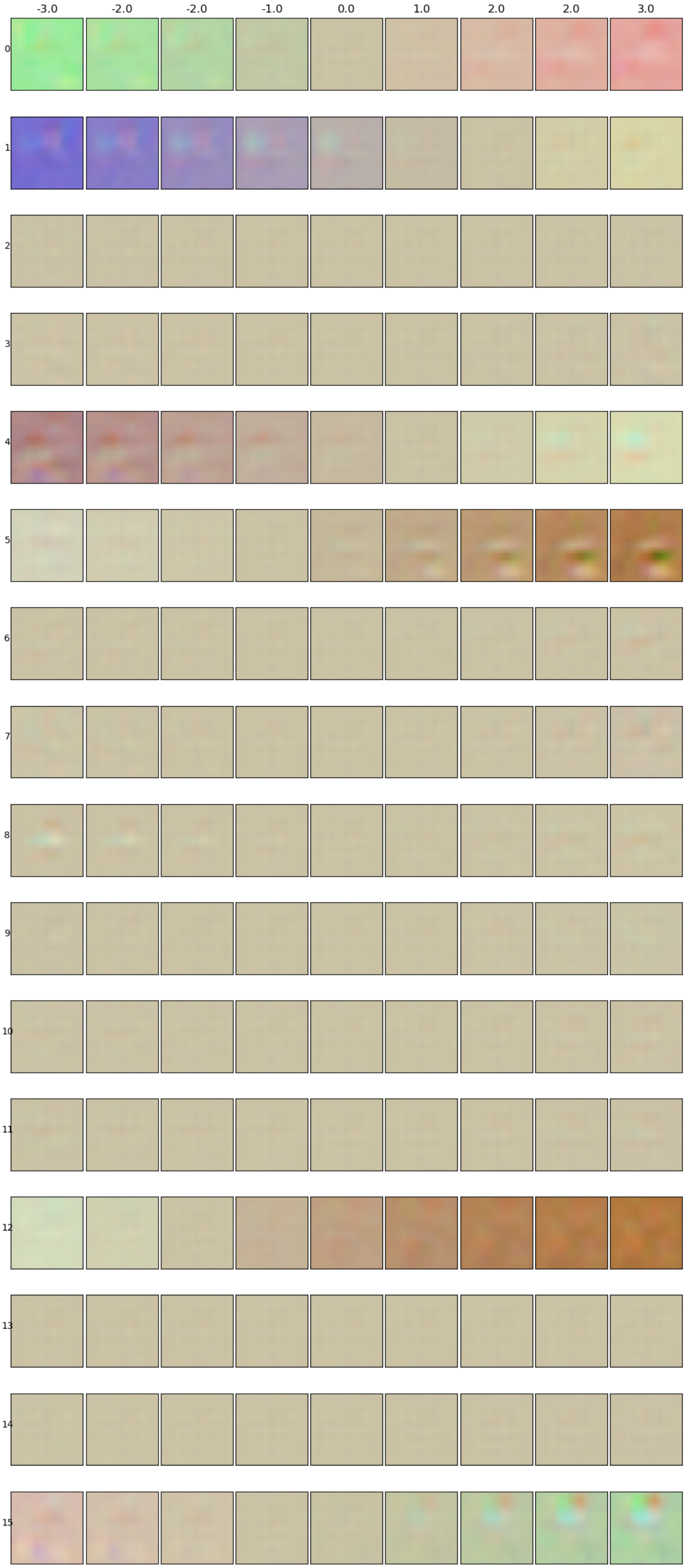

Figure 8: Latent Traversal of Standard VAE after 40 Epochs of Training

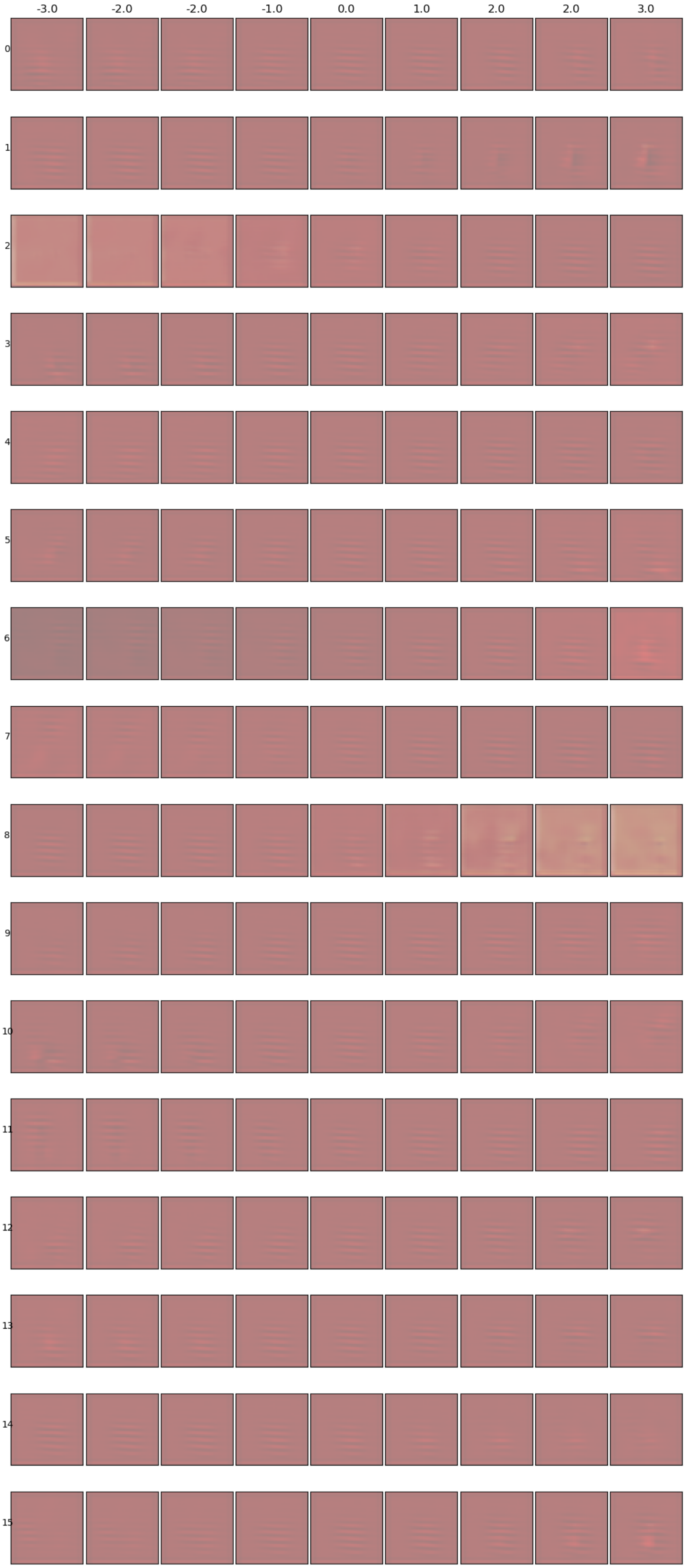

Figure 9: Latent Traversal of Hierarchical VAE after 40 Epochs of Training

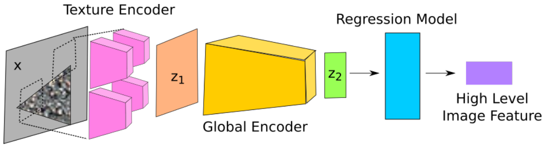

Figure 10: Pipeline for Regressing High-Level Features from the Hierarchical VAE Latent Representation

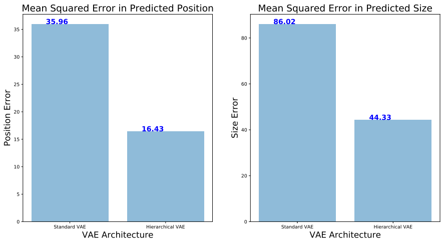

Figure 11: Error in Regressed Shape Position and Size From Latent Representations

However, we had difficulty regressing high-level features using this setup. This was likely ineffective because most of the patches contained no information about the triangle. We found that using a stride of 1 was more effective, likely because parts of the triangle were captured in multiple patches.

2. *Identifying the appropriate training dataset for learning high-level features.* Initially, we trained the Hierarchical VAE and the Standard VAE on Textured Triangle images from a single texture. We found that they either both failed to identify salient features of the image or they both succeeded in identifying high-level features. When the texture was very simple, such as a checkerboard pattern of colored squares, both VAEs generated reasonable reconstructions after sufficient training (although the Standard VAE required many more epochs than the Hierarchical VAE). However, if we selected a low contrast texture, both the Hierarchical VAE and the Standard VAE generated poor reconstructions. We found that introducing more variation into the training set resulted in better reconstructions for the Hierarchical VAE. This indicates that the model may continue to improve with more textures, and hence we hypothesize that it should scale well to more naturalistic images.

3. *End-to-end training causes the Hierarchical VAE to degenerate.* When the Hierarchical VAE is trained end-to-end with a loss that includes the reconstruction error $\mathbf{z}_1$, we observed that the training process quickly degenerates. We observed that the KL loss and the reconstruction loss quickly became 0 but did not produce meaningful representations of the input $\mathbf{x}$. We hypothesize that training end-to-end allows for changing the weights of the Texture VAE so that $\mathbf{z}_1$ contains no meaningful information but is easy to reconstruct by the Global VAE.

Related Work

Hierarchical VAEs were first proposed in the original work on variational autoencoders (Kingma & Welling, 2014). Since then, they have been applied to tasks such as modeling of sequences with long-term structure, as in the case of MusicVAE (Roberts, Engel, Raffel, Hawthorne, & Eck, 2018).

Hierarchical VAEs are an active area of study, and a recent work has a highlighted their advantages and limitations (Zhao, Song, & Ermon, 2017). Hierarchical VAEs are advantageous because they can improve the Evidence Lower Bound (which is used as a proxy for optimizing the intractable log-marginal likelihood) and decrease reconstruction error (Rezende et al., 2014). However, some argue that it is challenging to learn a meaningful hierarchy when there are many layers of latent variables (Sø nderby, Raiko, Maalø e, Sø nderby, & Winther, 2016). As a result, an alternative VAE architectures have been proposed, such as Ladder VAE.

Conclusion

The higher quality reconstructions, the more informative latent traversals, and the decodability of the latent space to learn high-level features reveal that imposing hierarchy aids in learning high-level features from an image. Nevertheless, imposing a hierarchy did not allow us to learn a disentangled feature representation, which could be an area of future work.

Acknowledgments

This work was completed at MIT in the class 6.804: Computational Cognitive Science, taught by Professor Josh Tenenbaum. We would like to thank Professor Tenenbaum for introducing us to computational theories of human cognition and presenting computational frameworks for human-like intelligence. We would also like to thank Nick Watters, who developed the initial idea for this project and provided guidance throughout the process. The code for this project is publicly available at here.

References

Bengio, Y., Courville, A., & Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence, 35(8).*

Dai, D., Riemenschneider, H., & Van Gool, L. (2014). The synthesizability of texture examples. In *Ieee conference on computer vision and pattern recognition (cvpr).*

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

Kim, H., & Mnih, A. (2018). Disentangling by factorising. *arXiv 1802.05983.*

Kingma, D. P., & Welling, M. (2014). Auto-encoding variational bayes. In *2nd international conference on learning* *representations, ICLR 2014, banff, ab, canada, april 14-16, 2014, conference track proceedings.* Retrieved from http://arxiv.org/abs/1312.6114

Marcus, G. (2018). Deep learning: A critical appraisal. *arXiv*.

Morton, D. (2018, Jun). *Learning disentangled representations.* Medium. Retrieved from https://medium.com/@davidlmorton/learning-disentangled-representations-part-1-simple-dots-c5553ecc995b

Rezende, D. J., Mohamed, S., & Wierstra, D. (2014). Stochastic backpropagation and approximate inference in deep generative models. *arXiv preprint arXiv:1401.4082*.

Roberts, A., Engel, J., Raffel, C., Hawthorne, C., & Eck, D. (2018). A hierarchical latent vector model for learning long-term structure in music. In *International conference on machine learning (icml).* Retrieved from http://proceedings.mlr.press/v80/roberts18a.html

Sø nderby, C. K., Raiko, T., Maalø e, L., Sø nderby, S. r. K., & Winther, O. (2016). Ladder variational autoencoders. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, & R. Garnett (Eds.), *Advances in neural information processing systems 29* (pp. 3738–3746). Curran Associates, Inc. Retrieved from http://papers.nips.cc/paper/6275-ladder-variational-autoencoders.pdf

Watters, N., Matthey, L., Borgeaud, S., Kabra, R., & Lerchner, A. (2019). *Spriteworld: A flexible, configurable reinforcement learning environment.* https://github.com/deepmind/spriteworld/. Retrieved from https://github.com/deepmind/spriteworld/

Zhao, S., Song, J., & Ermon, S. (2017, 06–11 Aug). Learning hierarchical features from deep generative models. In D. Precup & Y. W. Teh (Eds.), *Proceedings of the 34th international conference on machine learning* (Vol. 70, pp. 4091–4099). International Convention Centre, Sydney, Australia: PMLR. Retrieved from http://proceedings.mlr.press/v70/zhao17c.html