VisionEngine, investigating natural color-patterns with machine learning

R. Ian Etheredge[*]
Max Planck Institute of Animal Behavior[†]
University of Konstanz[‡]
ietheredge@ab.mpg.de

Manfred Schartl
University Wuerzburg
phch1@biozentrum.uni-wuerzburg.de

Lyndon Alexander Jordan
Max Planck Institute of Animal Behavior[†]
University of Konstanz[‡]
ajordan@ab.mpg.de

February 14, 2020

Abstract

We present a framework built on methods from probabilistic machine learning that extends existing approaches for understanding complex natural color patterns and testing evolutionary hypotheses. VisionEngine captures hierarchical relationships between features missed by existing techniques, disentangles factors of variation in biologically-interpretable ways and generates photorealistic samples from complex image distributions. It also integrates evolutionary algorithms to formally test models of selection, combining analytical, virtual, and empirical approaches.

Keywords Deep Learning · Color-Pattern Spaces · Generative Models · Evolution · Virtual Reality

Looking through the lens of an animal's umwelt—the individual or species-specific sensory landscape, reveals a "picture book of invisible worlds" where, outside of our own perceptual biases, stimuli can take on new meaning. [1] Body coloration patterns are some of the most complex and difficult components of an animal umwelt to quantify, being tailored by selection to the exact sensory experience of the receiver. An explicit goal of evolutionary biology is to understand traits as a consequence of the selective pressures acting on them: the adaptive landscape. [2] In terms of visual stimuli, describing a color-pattern space is fundamental to our understanding of adaptive landscapes but the utility of these descriptions requires careful accounting of our own perceptual biases, and biases within the chosen analytical or experimental framework.

Computer vision and immersive virtual reality (VR) provide the basis for modern analytical and experimental approaches to investigating the connections between visual inputs and behavioral outputs. Currently, most VR studies use characteristics which have been predetermined by human observers presented virtually using images and 3D animations [3, 4, 5] Modern methods for quantifying these characteristics better account for the role of spatial vision, have improved imaging and calibration methods, and incorporate non-human models of photoreceptor stimulation. [6, 7, 8] In addition to color, these approaches employ local image features, such as textures or key points. Predefined filters, e.g. those used in Scale Invariant Feature Transformation (SIFT, [9]), which are combined with first- and second-order image statistics into a multivariate set of features, a color-pattern space. Still, how best to connect morphological features with behavioral observations remains an open question; and some researchers worry about the gap developing between analytical and experimental results. [10]

[*] corresponding author
[†] Department of Collective Behaviour
[‡] Centre for the Advanced Study of Collective Behaviour

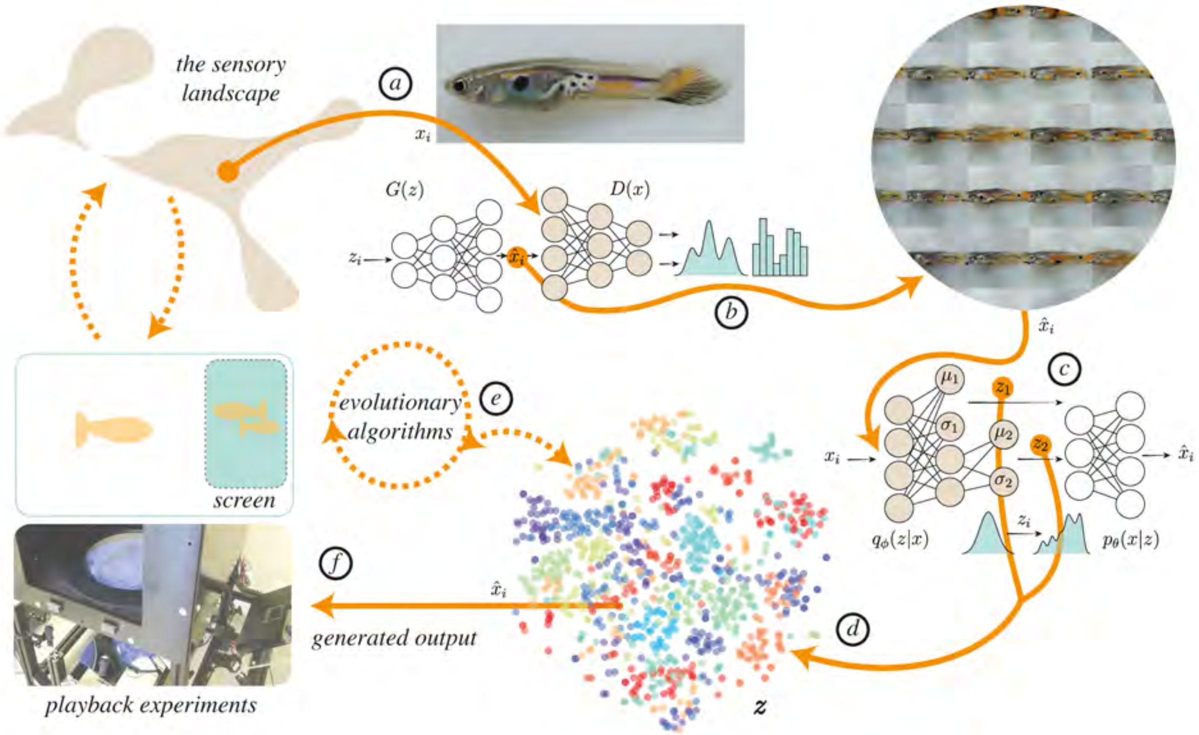

Figure 1: *Overview.* a) Many complex patterns (e.g. male guppy ornaments) consist of combinations of several elements which have distinct hierarchal relationships and, in addition to color contrast and size, may hold distinct biological importance. Sampling from the sensory landscape of these color patterns we input these to a generative (GAN) model which can b) produce an unlimited number of new samples. We use these samples as input to c) a variational auto encoder with a hierarchical structure designed to disentangle factors of variation across multiple scales of increasing abstraction (z_1 through z_n). We learn a distribution over these variables parameterized by a mean and variance term and use the embedding of samples in this distribution to define d) a color-pattern space. We can use this low dimensional representation as e) input to downstream models such as evolutionary algorithms and f) produce photo-realistic outputs to be used in playback experiments and immersive VR. Interpolations through the color-pattern space with animated models and VR allows researchers to manipulate generated output for experimental tests.

Many of the metrics used to describe color-pattern spaces stem from a distinctly Newtonian tradition for understanding color that divorces light as a physical object from the physiological processes of perception. [11] In the visual cortex and its homologues, top-down mechanisms abound and high level representations heavily influence the perception of low level features. [12] However, existing quantitative approaches do not reconcile physical theories with the hierarchical mechanisms of visual processing and our quantitative understanding of aesthetics outside of the color-spectrum remains limited.

Current approaches characterize many of the regular patterns found in nature [8] but miss higher-level interactions and impose strict assumptions in terms of rigidly defined filters. Meanwhile, in the field of artificial intelligence, deep convolutional neural networks (CNNs) are used for natural image processing with increasing success. [13] In CNNs, handcrafted features are discarded in favor of filters optimized directly on the sample data, freeing them from the biases imposed by predefined features. Using sequential stacks of convolutional layers, CNNs combine information across spatial scales. While CNNs are not explicitly modeled on the brain, like the visual cortex, representations made at higher convolutional layers capture more abstract features as well as influence the learned weights and biases at lower layers. CNNs have previously been used as a model for hidden preferences in evolutionary signals [14] but as they become more and more powerful, confusion persists about how to properly leverage them and interpret their output. [8] Going from recognizing the potential of deep networks to realizing their practical application to research depends on building interpretability across disciplines outside of computer science.

Here, we combine approaches from unsupervised, generative models (i.e. variational auto encoders, VAEs, [15] and generative adversarial networks, GANs [16]) with techniques from representation learning [17] into a framework that compresses complex image data to produce informative variable distributions—latent representations, eliminating the need to interpret individual network layers which are often uninformative (see Methods Figure 3). Our framework 1: disentangles factors of variation and 2: captures a hierarchy of features across scales. [18]) Allowing us to build quantitative comparisons of complex hierarchal features. Moreover, because this framework is generative, it 3: provides crucial direct connections between analytical and experimental approaches that do not currently exist, including interpolation between samples and traversals of the latent representation. It also 4: provides a strategy for leveraging small datasets, a common limiting factor for many biological studies, by generating novel samples from the image distribution. [19] In addition to compressing information in meaningful and interpretable ways and generating realistic outputs, we can freely combine these representations with a range of downstream models, e.g. as part of VR experiments; effectively removing the need for human-in-the-loop approaches.

We demonstrate this framework on an example dataset of male guppy images. Guppies are a model organism for investigating the selective forces which produce complex, irregular color-patterns (see Supplemental Material for an additional application to butterfly images). Our latent representation consists of 10 variables across four latent codes (Z.1 - Z.4) with increasing capacity to capture abstract features (see Methods). After training, we embed 13000 generated samples (Figure 1b) into this 40-dimensional space and find distinct clusters which correspond to the generated sample categories (Figure 2b). Latent variables across the four increasingly abstract encodings reveal scale-dependent factors of variation in the reconstructed output (Figure 2a, Supplemental Figure 3). These latent variables affect samples in consistent and interpretable ways (Figure 2a). Z.1, the latent code with the lowest capacity captures local traits such as the color and intensity of discrete local patches, e.g. Z.1.1 encodes variation in the intensity of an orange spot in the caudal fin (Figure 2a left). At higher levels (Z.2 - Z.4), latent variables encode traits which combine multiple elements, e.g. latent variable Z.4.1 decreases the distance between a cluster of anterior black spots (Figure 2a right). This allows us to compare complex feature sets directly via the low-dimensional latent representation without imposing human-level descriptions and investigate the relative contribution of features across scales. We can calculate likelihood estimates for each data sample and identify those with low probability given the model parameters (Supplemental Figure 1a). Rare samples of guppy ornament patterns, such as the "Tr5" strain in our sample data which are distinctly melanated, cluster together and have a low sample likelihood (Supplemental Figure 1a). Likelihood estimates have direct connections to fitness outcomes. For example, in guppies rare male phenotypes may be preferred by females. [20]

Typically, biological datasets are several orders of magnitude smaller than those used in machine learning tasks. Our strategy to overcome this limitation is to first use a modified GAN model [19] to generate a large number of out-of-sample images. For training the above VAE model, we used only a small subset of the original data as input to the GAN model and incorporated previous knowledge of our sample data via a multi-class categorical latent code. By generating samples conditioned on this variable we capture inherent characteristics while also disentangling the covariance of features in the sample data (see Methods, Supplemental Figure 1b). Samples from each category possess unique features, e.g. a distinct black bar and orange stripe which characterizes one guppy species, P. wengei (Figure 2b, category number 11,)

We can also leverage our latent representation as input to evolutionary algorithms (Figure 1e). For evolutionary biologists and ecologists, this provides a transformative new technique to formally test models of selection. For example, we can manipulate the latent code, based on the expected fitness, to test established hypotheses from Sexual Selection; e.g. using a simple heuristic from the guppy literature: oranger, higher contrast males are preferred by females. [20] Using this fitness function in our evolutionary model, we "evolved" the latent code to produce sample outputs with these traits exaggerated to produce qualitatively "fitter" generated samples after simulating many generations of selective pressure directly from a randomly initialized parent population (Figure 2c).

Using an interpretable latent representation which captures a hierarchy of features allows us to quantify important higher-level features current approaches miss. This better captures the true topography of color-pattern spaces and adaptive landscapes. By learning these features directly from the data, instead of imposing them in our choice of analytical tool, we can ensure our own perceptual biases do not undermine the way we summarize our sample data. And, because our framework can be easily extended, we can simulate multiple fitness landscapes simultaneously while being explicit about the parameters of an evolutionary model and test these results empirically using generated output as part of immersive VR playback experiments (see Supplemental Movie 1). As the latency between input and output decreases in VR experiments, future work will integrate these model extensions to enable a holistic framework that also incorporates instantaneous behavioral feedback and even richer in-the-loop methods for hypothesis testing.

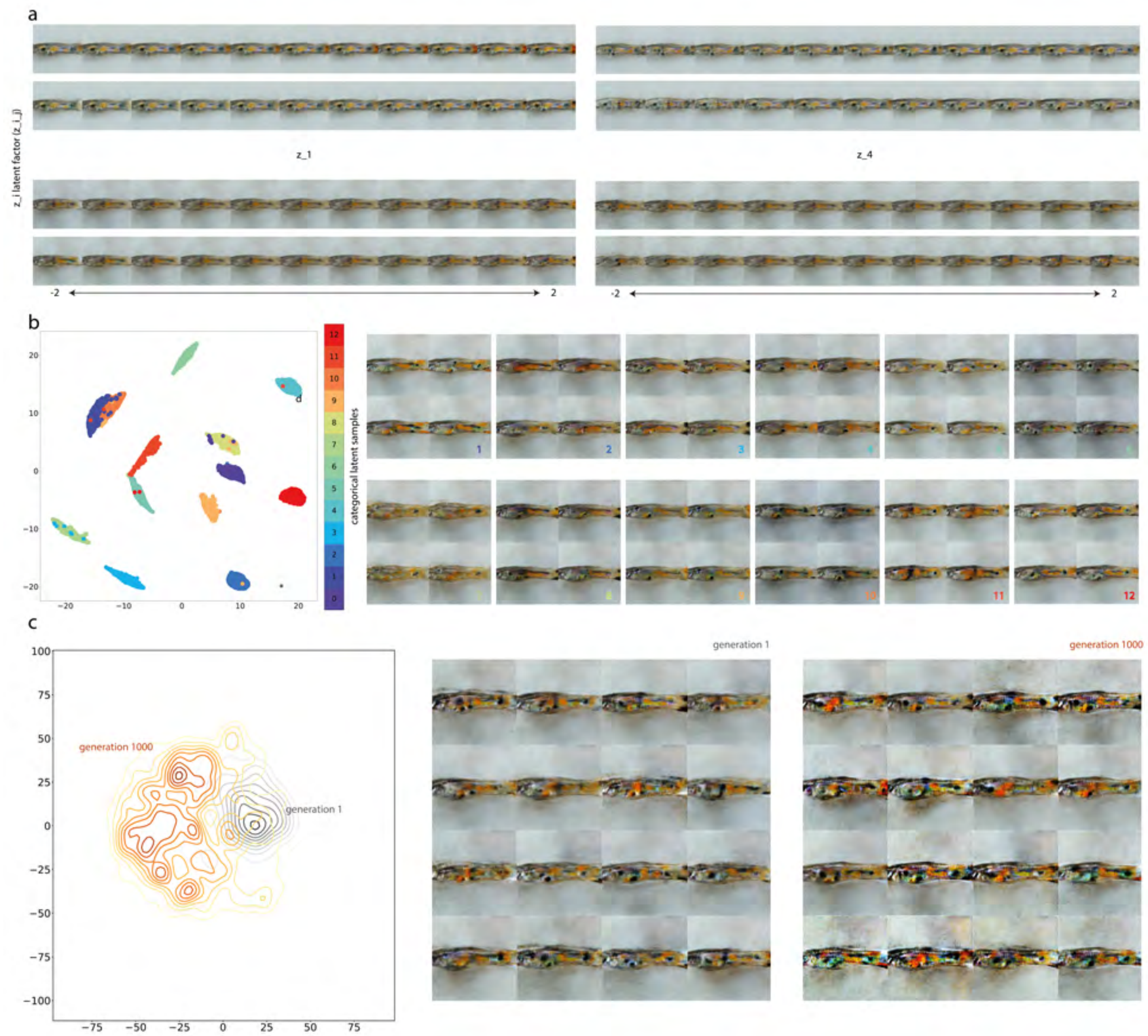

Figure 2: *Guppy ornaments*. a) Latent traversal of 2 latent variables from z_1 and z_4. Top is an embedded sample and bottom a latent code initialized at zero. For each latent variable (rows) we traverse values between -2 and 2 for the generated output. We see that each latent code has consistent effects; e.g. Z.1.1 (top rows, left) encodes variation in the intensity of an orange spot in the caudal fin, Z.4.1 (top rows, right) decreases the distance between a cluster of anterior black spots b) categorical labels of input samples in the same space and examples from each category of the trained generative model, categorical labels. We incorporated knowledge of our sample data by providing a 13-class categorical latent code and were able to generate examples from distinct classes learned by the model which capture meaningful combinations of features in our sample data. c) Kernel density plot of samples over generations, selecting for percent orange and percent black. After 1000 generations (orange) the population has shifted from it's initial distribution, filling the space and finding multiple peaks which maximize the fitness function. Sample of initial parent population, middle, randomly sampled from the distribution of latent variable compared to a sample of the population after 1000 generations of selection on percent orange and black, right. Samples that have undergone selection show noticeable larger and brighter orange spots as well as higher contrast and larger black spots as would be predicted.

References

[1] Jakob Von Uexküll. A stroll through the worlds of animals and men: A picture book of invisible worlds. *Semiotica*, 89(4):319–391, 1992.

[2] John Maynard Smith. Natural selection and the concept of a protein space. *Nature*, 225(5232):563–564, 1970.

[3] Spencer J Ingley, Mohammad Rahmani Asl, Chengde Wu, Rongfeng Cui, Mahmoud Gadelhak, Wen Li, Ji Zhang, Jon Simpson, Chelsea Hash, Trisha Butkowski, et al. anyfish 2.0: an open-source software platform to generate and share animated fish models to study behavior. *SoftwareX*, 3:13–21, 2015.

[4] John R Stowers, Maximilian Hofbauer, Renaud Bastien, Johannes Griessner, Peter Higgins, Sarfarazhussain Farooqui, Ruth M Fischer, Karin Nowikovsky, Wulf Haubensak, Iain D Couzin, et al. Virtual reality for freely moving animals. *Nature methods*, 14(10):995, 2017.

[5] Hemal Naik, Renaud Bastien, Nassir Navab, and Iain Couzin. Animals in Virtual Environments. *arXiv e-prints*, page arXiv:1912.12763, Dec 2019.

[6] Eleanor M. Caves, Nicholas C. Brandley, and Sönke Johnsen. Visual acuity and the evolution of signals. *Trends in Ecology Evolution*, 33(5):358 – 372, 2018.

[7] Julien P Renoult, Almut Kelber, and H Martin Schaefer. Colour spaces in ecology and evolutionary biology. *Biological Reviews*, 92(1):292–315, 2017.

[8] Mary Caswell Stoddard and Daniel Osorio. Animal coloration patterns: Linking spatial vision to quantitative analysis. *The American Naturalist*, 193(2):164–186, 2019.

[9] D. G. Lowe. Object recognition from local scale-invariant features. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, volume 2, pages 1150–1157 vol.2, Sep. 1999.

[10] Anselm Brachmann and Christoph Redies. Computational and experimental approaches to visual aesthetics. *Frontiers in Computational Neuroscience*, 11:102, 2017.

[11] Isaac Newton. *Opticks*. Dover Press, 1704.

[12] David H Hubel and Torsten N Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of physiology*, 160(1):106–154, 1962.

[13] Yann LeCun, Koray Kavukcuoglu, and Clément Farabet. Convolutional networks and applications in vision. In *Proceedings of 2010 IEEE international symposium on circuits and systems*, pages 253–256. IEEE, 2010.

[14] Anthony Arak and Magnus Enquist. Hidden preferences and the evolution of signals. *Philosophical Transactions of the Royal Society of London. Series B: Biological Sciences*, 340(1292):207–213, 1993.

[15] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

[16] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Networks. *arXiv e-prints*, page arXiv:1406.2661, Jun 2014.

[17] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, Aug 2013.

[18] Shengjia Zhao, Jiaming Song, and Stefano Ermon. Learning hierarchical features from deep generative models. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 4091–4099. JMLR. org, 2017.

[19] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in neural information processing systems*, pages 2172–2180, 2016.

[20] Anne E Houde. Mate choice based upon naturally occurring color-pattern variation in a guppy population. *Evolution*, 41(1):1–10, 1987.

VisionEngine: Online Methods

February 14, 2020

1 Summary

Many complex patterns consist of combinations of several elements which have distinct hierarchal relationships and, in addition to color contrast and size, may hold distinct biological importance. Sampled data are first input to a generative (GAN) model (see section 1.1) which learns to generate samples parametrized by a set of categorical latent codes. This allows us to produce an unlimited number of new samples from the image distribution while imposing some prior knowledge about our sample data (the number of species, distinct groups, etc.). We use these generated outputs, along with our original samples as input to a variational auto encoder (VAE) model (see section 1.2) with a hierarchical structure designed to disentangle factors of variation across multiple scales of increasing abstraction, which is a limitation of typical VAE model architectures. This learns a distribution over latent variable distributions parameterized by a mean and variance term. Embedding our sample in this distribution we define a color-pattern space which captures the factors of variation in the data for comparison across samples. Additionally, by manipulating variables within the latent code we can interpolate between samples and generate out-of-sample examples. We may use our latent-representation as input to downstream models such as evolutionary algorithms we incorporate theoretical approaches. And, because this representation can produce photo-realistic outputs, we can combine these manipulations on the latent code with playback experiments and immersive VR (Figure 1, main text).

In terms of the color-pattern space, our latent representation, we want to identify key traits and, because many color patterns have scale dependent relationships between features, we want to distinguish between features at different scales. To make this representation useful for analysis, we would like to map these features to a low-dimensional space which we can investigate which show consistent effects across samples. [1, 2] Ultimately, achieving these characteristics relies on the way we represent the underlying data and by applying key meta-prior enforcement strategies [3] from machine learning we can ensure our representation has characteristics which make them interpretable and useful for studying biological phenomenon, table 1, Supplemental Figure 3 (see Appendix Section 3 for a larger discussion).

All models were implemented using Tensorflow 2.0 and can be accessed at https://www.github.com/ietheredge/VisionEngine. We provide notebooks for evaluating our findings and training novel datasets.

Table 1: Our framework

Desired Characteristic	Representation Learning Meta-Prior [3]	Example Approach
Disentangling factors of variation	Limited number of shared factors of variation	Latent regularization [4]
Capturing spatial relationships	Hierarchical organization of representation	Hierarchical model architecture [5]
Incorporating existing knowledge	Local manifold variation	Structured latent codes [6]
Connect analyses and experiments	Local variation on manifolds	Generative models [7, 8]
Perform statistical inference	Probability mass collects on manifold	Variational Bayesian inference [7]

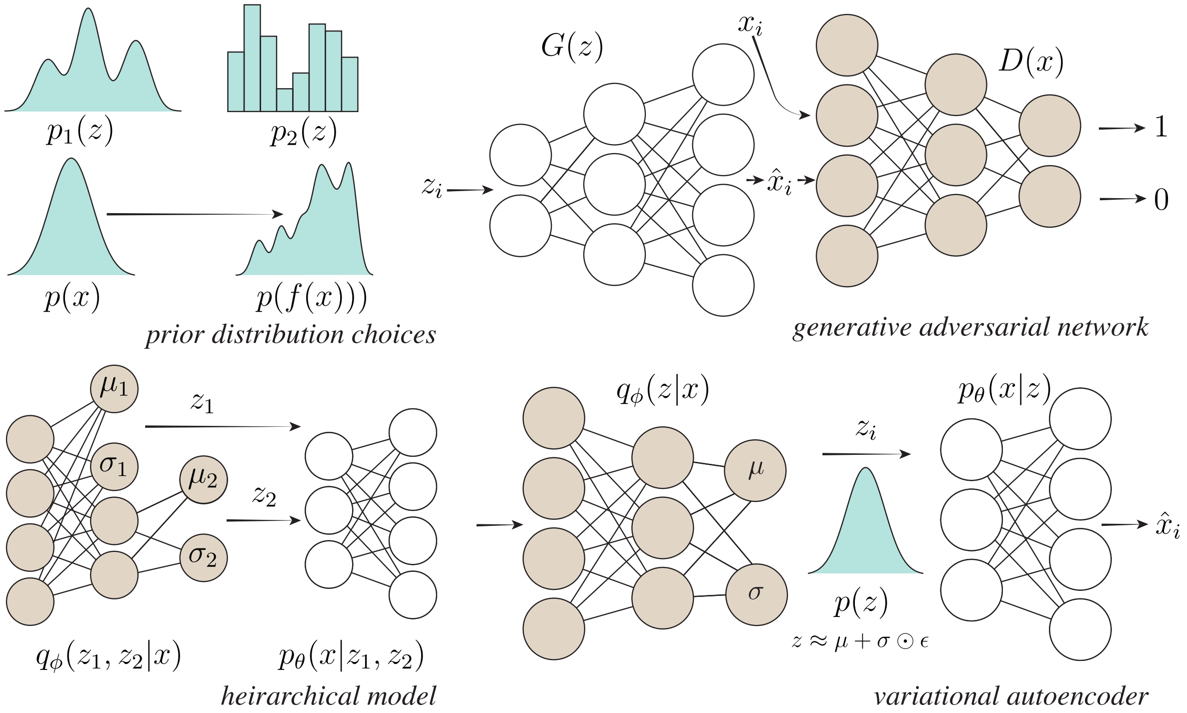

Figure 1: *Key Methods Top left*: the distributions of our latent representation may be parameterized by continuous or discrete distributions or there may be multiple latent distributions. In infoGAN a categorical distribution is combined with two continuous latent codes which allows for semi-supervised approaches, for example by setting the number of categories in the discrete distribution to correspond to a number of known classes (e.g. 10 categories of digits).*Top right*, example structure of a generative adversarial network. Here, a noise vector, z_i is input to the generator network $G(z)$ which produces a reconstructed output $\hat{x_i}$. A real sample, x_i, and generated sample $\hat{x_i}$ are subsequently passed through a separate discriminator network $D(x)$ which determines if the sample is real (1) or generated (0). Known as an adversarial loss, $G(z)$ is optimized such that generated inputs are harder and harder for $D(x)$ to distinguish from real. *Bottom left*, architecture of a variational ladder autoencoder (VLAE). Multiple latent spaces $(z_1, z_2, ..., z_k)$ are learned from multiple layers of a convolutional encoder and different levels of feature abstraction, mitigating the explain away problem and allowing for bottom-up and top-down feedback. *Bottom right*, structure of a variational autoencoder (VAE). x_i and $\hat{x_i}$ are an example input and its reconstructed output, the probabilistic encoder or inference model, $q_\phi(z|x)$, performs posterior inference learning shared model parameters, ϕ, across samples, approximating the true posterior distribution. The probabilistic decode, $p_\theta(Z|X)$, $p_\theta(X|Z)$, learns a joint distribution of the encoded space, Z, and the data space x. The low dimensional bottleneck, Z is a distribution of latent variables capable of reconstructing sample inputs, parameterized by a vector of means μ and standard deviations σ the noise term ϵ allows for the parameters of this multivariate distribution to be optimized using back propagation, known as the reparametrization trick.

1.1 Addressing small datasets

We adapt a GAN-based approaches for generative modeling which incorporates a categorical and continues latent code. [6] Typically in GANs, the generator network $G(z)$ learns to manipulate a noise vector to reproduce outputs (as judged by the discriminator $D(z)$, Figure 1, top right). The primary contribution of [6] was that they were able to introduce latent codes which produce predictable and consistent effects across generated samples. We substitute the original generator and discriminator models from [6] with DarkNet architectures from YOLOv3. [9] which have been shown to produce more photo-realistic outputs. We also increase the flexibility of the latent code, providing K continuous and discrete latent codes, e.g. in our guppy sample data, we have 13 different lines of guppies and use a K=13 class categorical code with 2 additional continuous latent codes. However, we could have alternatively used three latent codes, representing the three different guppy species (*P. wengei P. reticulata, P. obscura*) or any other number of categorical latent codes which represent known groupings we wish to compare in our analyses. Our generator uses a 100-unit latent code as input. The model was trained for 200000 training steps with a learning rate of 2e-4 and a decay of 6e-8 per epoch using an RMSprop optimizer (Supplemental Figure 3, top)

By using generated data as input to the VAE model (below) we gain additional advantages for encouraging disentanglement of the generative factors in our sample data. This relates to the restrictive mean-field assumption of the objective function which emphasizes the local feature context for understanding non-linear relationships and local dependence on the relative placement of other features at the cost of large scale spatial relationships (See Supplemental Figure 1). By using these generated outputs to our hierarchal model, below (where we wish to preserve these relationships), it allows the network to better disentangle the range of scale-dependent feature dependencies.

1.2 Uncovering hierarchal feature relationships

Variational Autoencoders (VAEs) [7] learn a representation which maps inputs to a compressed latent code which contains the information needed for reconstruction. The basic architecture of VAEs consist of two models, an encoder which performs posterior inference on the sample and a decoder which learns a joint distribution and reconstructs outputs from this encoded representation. For more background on VAEs.

Typically, VAEs use amortized approximate inference and the evidence lower bound (ELBO) objective. [10, 7] However, inherent properties of the ELBO objective, create a trade off between inference and data fit and amortized inference may fail; in particular when samples are high dimensional relative to the latent representation which is often the case with natural image data. When using a complex decoder $p_\theta(X|Z)$, as we do here, the mutual information between X and Z can become small—the "information preference" problem, which undermines the primary goal of creating a meaningful latent representation. [11, 5] To address this issue, we use the maximum mean discrepancy approach (MMD) [12] which modifies the ELBO objective to address these shortcomings by maximizing the similarity between the moments of $p_\theta(x|z)$ and $q_\phi(z|x)$. This reduces the quality of output sample, due to the rate distortion tradeoff, but ensures that our latent encoding is informative.

VAEs typically rely on a restrictive mean-field assumption to perform training, emphasizing the local feature context for understanding non-linear relationships and local dependence on the relative placement of other features at the cost of large scale spatial relationships.To impose a hierarchical spatial structure on our latent code we split out encoder across multiple embedding, each with increasing capacity where shallow networks express simple low-level features and deeper networks express more high-level complex ones (Supplemental Figure 3, bottom) . [4]

We adapt the architecture presented in [5] to include K latent codes of increasing expressivity with N latent variables each (for both guppy and butterfly datasets we set K=4 and N=10). Each latent code stacks convolutional layers with batch normalization and leaky ReLU activation with a kernel size of four, gaussian kernel initializers with a standard deviation of 0.02 and L2 kernel regularization. Each latent variable is paramterized by a multivariate normal distribution with mean of zero and standard deviation of one. We measure the discrepancy between this prior using either one or four distribution moments. [5, 6] For the latent code we utilize a mean maxim discrepancy (MMD [12]) loss weighted evenly across the four latent codes weighted by a hyperparameter $\Lambda = 5000$, chosen to balance the magnitude of the latent loss with the reconstruction loss. We use a denoising training paradigm, in addition to the noise added in the variational layer of the VAE, we add additional Gaussian and salt and pepper noise to inputs during training. [13] This approach has been shown to both encourage disentanglement and better utilize limited sample sizes. Mean squared error reconstruction loss between input and output was weighted by half of the number of pixel values in the images ((256*256*3)/2). The VAE model was trained with an linearly increasing learning rate between 10e-5 and 10e-8 using an Adam optimizer. Larger learning rates made training unstable. We used an early stopping criterion until the validation accuracy improved less that 0.0001 for at least 15 epochs

1.3 Evolving new samples

We use a simple genetic algorithm where an initial population of 1000 parent samples are selected up over 1000 generations. Parent samples are random initialized a random normal initialization across the the latent variables of each latent coder. Fitness was calculate as an equally weighted sum of the total percentage of pixels within two ranges (orange rgb(0.9, 0.55, 0.) > rgb(1., 0.75, 0.1) and black rgb(0., 0., 0.) < rgb(0.2, 0.2, 0.2)) measured on the generated output, a simplification of empirical results from the literature. [14, 15] We created a table of 10000 even drawn samples along with the calculated fitness scores to decrease processing time. During each generation the value of each sample was compared to its predicted fitness, measured by the fitness of the nearest neighboring value in the reference table. During selection, we drew 500 random subsamples weighted by the proportional fitness of each sample. An additional 200 samples were drawn, without the proportional fitness weighting. From the 700 subsamples in each generation we drew 300 random pairs, the "alleles" from each sample (the specific latent variable values) were chosen randomly with equal probability to create a combined offspring between the two samples. Each combined offspring then had two alleles mutated, one by drawing from a random normal distribution and the other by replacing an existing value with zero (similar to destabilizing and stabilizing mutations). The next generation of 100 samples, 700 parent samples + 300 offspring, repeated this process for 1000 generations.

2 Sample data

Guppy images were collected from a maintained stock at the University of Wuerzburg under authorization 568/300-1870/13 of the Veterinary Office of the District Government of Lower Franconia, Germany, in accordance with the German Animal Protection Law (TierSchG). Individuals were imaged on a white background with fixed lighting conditions [16] using a Cannon D600 digital camera. Images were not linearized and down sampled and center cropped to final size of 256 x 256 pixels. The dataset consists of 989 standardized RGB images across three species and 13 individual strains.

Butterfly images were downloaded from the Natural History Museum, London under a creative commons license (DOI: https://doi.org/10.5519/qd.gvq3p7xg). This dataset consists of 1991 RGB images and spans 8 Families.

References

[1] Mary Caswell Stoddard and Daniel Osorio. Animal coloration patterns: Linking spatial vision to quantitative analysis. *The American Naturalist*, 193(2):164–186, 2019.

[2] Julien P Renoult, Almut Kelber, and H Martin Schaefer. Colour spaces in ecology and evolutionary biology. *Biological Reviews*, 92(1):292–315, 2017.

[3] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, Aug 2013.

[4] Shengjia Zhao, Jiaming Song, and Stefano Ermon. Infovae: Information maximizing variational autoencoders. *arXiv preprint arXiv:1706.02262*, 2017.

[5] Shengjia Zhao, Jiaming Song, and Stefano Ermon. Learning hierarchical features from deep generative models. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 4091–4099. JMLR. org, 2017.

[6] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in neural information processing systems*, pages 2172–2180, 2016.

[7] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

[8] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Networks. *arXiv e-prints*, page arXiv:1406.2661, Jun 2014.

[9] Joseph Redmon, Santosh Kumar Divvala, Ross B. Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 779–788. IEEE Computer Society, 2016.

[10] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In Eric P. Xing and Tony Jebara, editors, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 1278–1286, Bejing, China, 22–24 Jun 2014. PMLR.

[11] Tian Qi Chen, Xuechen Li, Roger B Grosse, and David K Duvenaud. Isolating sources of disentanglement in variational autoencoders. In *Advances in Neural Information Processing Systems*, pages 2610–2620, 2018.

[12] Arthur Gretton, Karsten Borgwardt, Malte J Rasch, Bernhard Scholkopf, and Alexander J Smola. A kernel method for the two-sample problem. *arXiv preprint arXiv:0805.2368*, 2008.

[13] Daniel Im Jiwoong Im, Sungjin Ahn, Roland Memisevic, and Yoshua Bengio. Denoising criterion for variational auto-encoding framework. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[14] Anne E Houde. Mate choice based upon naturally occurring color-pattern variation in a guppy population. *Evolution*, 41(1):1–10, 1987.

[15] John A Endler and Anne E Houde. Geographic variation in female preferences for male traits in poecilia reticulata. *Evolution*, 49(3):456–468, 1995.

[16] Darrell J Kemp. Female mating biases for bright ultraviolet iridescence in the butterfly eurema hecabe (pieridae). *Behavioral Ecology*, 19(1):1–8, 2008.

VISIONENGINE: SUPPLEMENT

February 14, 2020

1 Embedding and Likelihoods

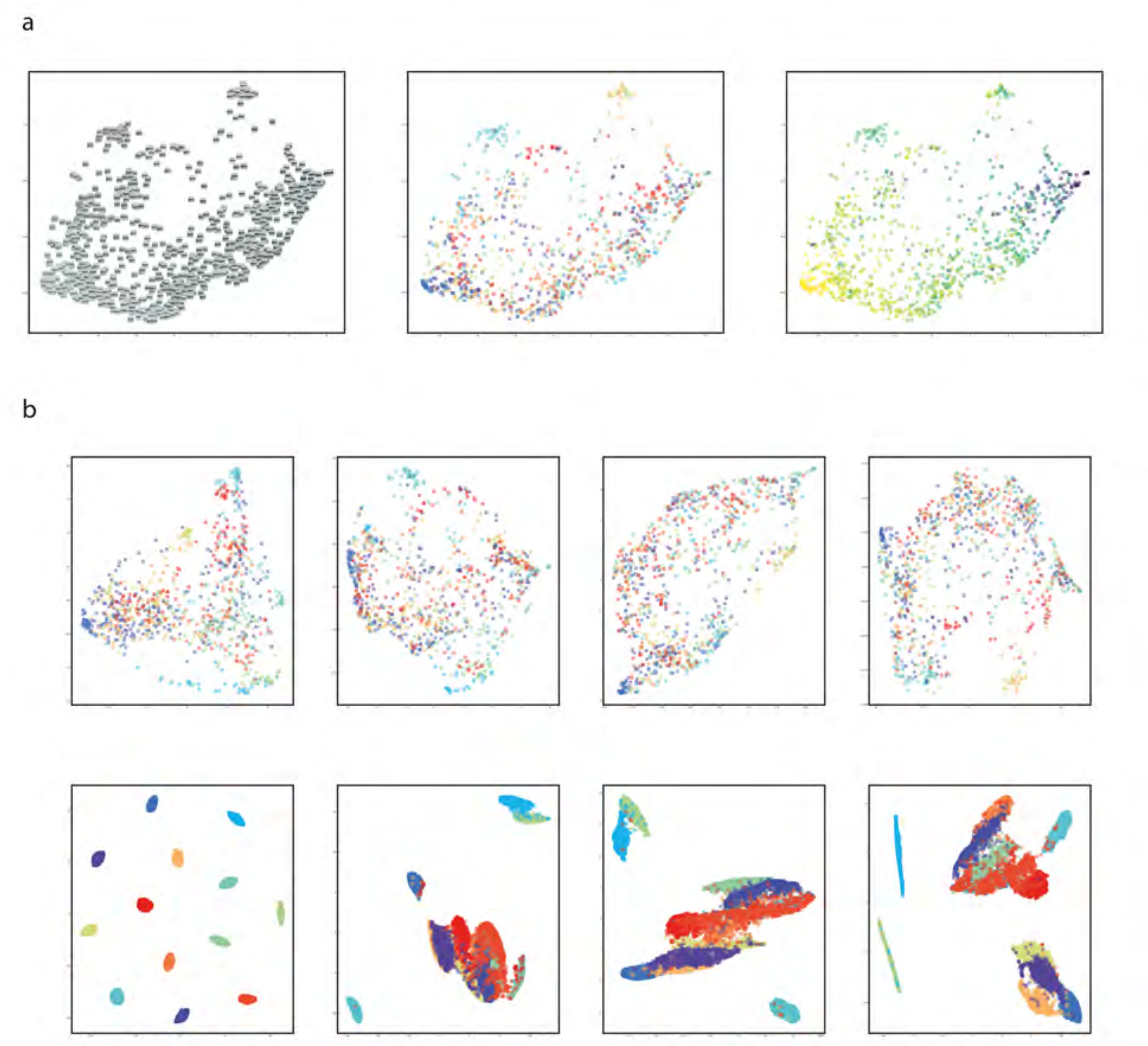

Figure 1: *Guppy ornament color-space* a) Left, embedded input samples images (we project the 40-dimensional latent code to 2D for visualization). Middle, colors indicate the 13 distinct guppy strains in the sample data. Right, sample likelihood estimates likely (yellow) and unlikely (blue) phenotype based on the model parameters. b) 2D embedding of each latent code (z_1 through z_4) for real sample images, and generated output. Colors indicate guppy strain (top) and categorical code (bottom).

2 Latent Traversal

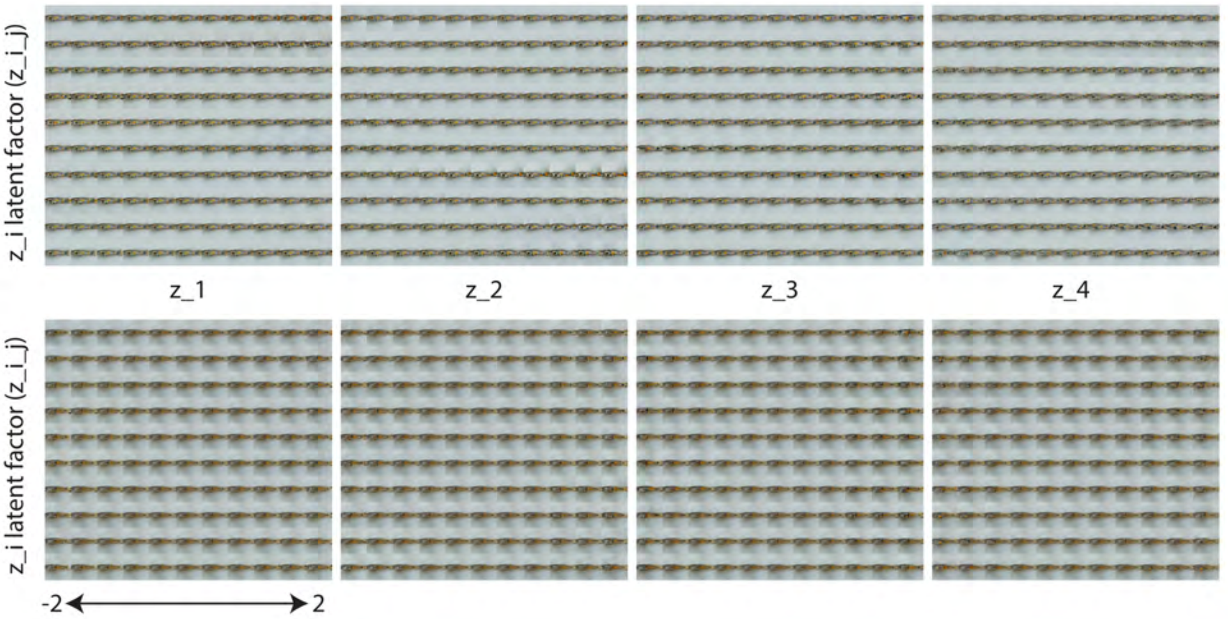

Figure 2: *Latent Traversal* From left to right, the four latent codes with increasing capacity. Top, an embedded sample, bottom latent code initialized at zero. We traverse values of each variable (from -2 to 2) while holding all others at their initial values.

3 NHML Butterflies

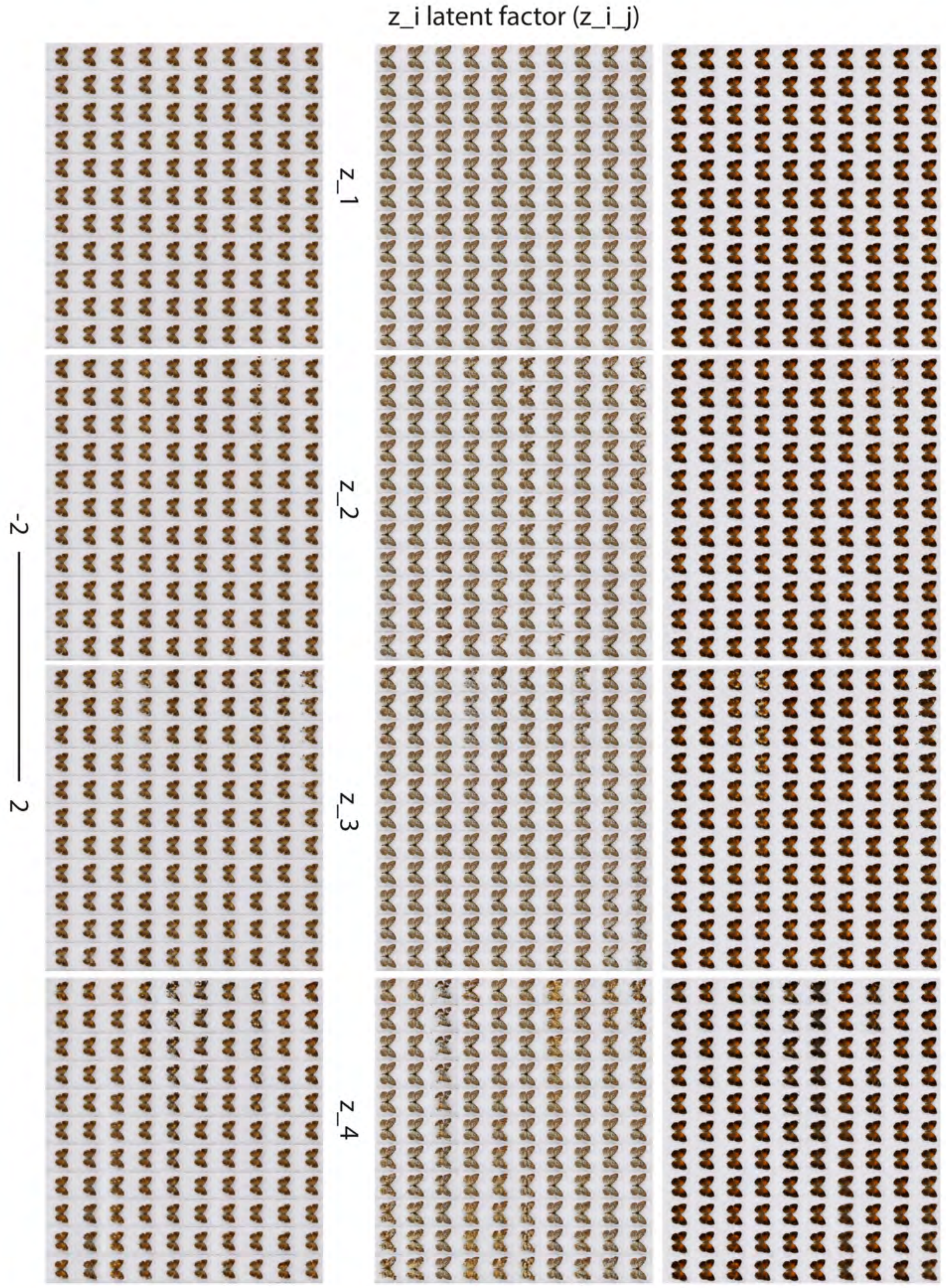

Figure 3: *Latent Traversal* From left to right, the four latent codes with increasing capacity. Three embedded samples from the original data set. We traverse values of each variable (from -2 to 2) while holding all others at their initial values.

4 Background on GANs and VAEs

Generative Adversarial Networks (GANs) [1] use a randomly initialized vector as input to a generative model for reproducing example input. Unlike likelihood based methods, GANs use adversarial learning where a generator network produces fake samples from these noise vectors which resemble real sample data, and is fine tuned by the classification output of the discriminator network (real or fake).The structure within these noise vectors is sufficient to produce consistent and meaningful effects on the output from trained generators and can be manipulated to produce samples which combine attributes from across the sample data.

Variational Autoencoders (VAEs) [2] learn a representation which maps inputs to a compressed latent code which contains the information needed for reconstruction. The basic architecture of VAEs consist of two models, an encoder which performs posterior inference on the sample and a decoder which learns a joint distribution and reconstructs outputs from this encoded representation. Here, our encoder and decoder models are directed graphical models, CNNs, which are the inverse of each other. VAEs learn an encoding which maps a distribution p_θ such that the relationships between inputs X and the encoding Z is described by a prior $p_\theta(Z)$, likelihood $p_\theta(X|Z)$, and posterior distribution $p_\theta(Z|X)$. From this distribution we can generate a novel example x_i by first sampling from $p_\theta(Z)$ and generating from the conditional likelihood distribution $p_\theta(X|Z = z_i)$. In most case a simple distribution is provided for the latent variables, e.g. a multivariate normal distribution parameterized by a vector of means μ and standard deviations σ the noise term ϵ allows for the parameters of this multivariate distribution to be optimized using back propagation. The inputs conditioned on these latent variables are modeled as a deep neural network. The posterior distribution $p_\theta(Z|X)$ is approximated by our probabilistic encoder $q_\phi(Z|X)$ which, in addition to the decoder $p_\theta(X|Z)$, may have tens of millions of parameters. Computing the posterior distribution would require integrating over all possible K_θ^n configurations making inference computationally intractable. Originally, VAEs have used amortized approximate inference and the evidence lower bound (ELBO) objective. [3, 2] However, inherent properties of the ELBO objective, create a trade off between inference and data fit and amortized inference may fail; in particular when X is high dimensional relative to Z which is often the case with natural image data. Even worse, when using a complex decoder $p_\theta(X|Z)$, the mutual information between X and Z can become small, the "information preference" problem, which undermines the primary goal of creating a meaningful latent representation. [4, 5] Here we use the maximum mean discrepancy approach (MMD) [6] which modifies the ELBO objective to address these shortcomings by maximizing the similarity between the moments of $p_\theta(x|z)$ and $q_\phi(z|x)$.

5 Meta Priors

The hesitancy of some researchers to adopt deep learning as a standard part of their toolset stems from characteristics of the representations learned by supervised learning tasks. Understanding complex color patterns and bridging the gap between analytical and experimental approaches requires us to think about the nature of the representations we work with. In typical *supervised* discriminative models, the objective being optimized is well defined, e.g. accurate classification or localization. As such, the representations provided by the downstream convolutional layers of deep networks take on characteristics optimized for performance on this task. At higher and higher network layers, the boundaries between classes can become complex and specialized to this objective because of the usefulness of such representations to identifying complex boundaries. Since networks naturally absorb all the correlations found in the sample data they can become biased by useful (in terms of the objective function) but spurious correlations. For example, say we want to classify cows and camels, we train the network on a large sample dataset of images but strangely the network continually fails when testing images of cows on sandy backgrounds. As it turns out, most of the images in our training set were of cows on grassy pastures and camels on sandy deserts. Heavily influenced by the biases in our training data, the network found a useful shortcut, optimizing on the spurious correlation of background landscape instead of the characteristics of the animals themselves.

An increasing focus is being placed on dealing explicitly with representations themselves, asking what makes a particular representation more or less useful or informative, *generally*, across tasks. At a basic level, one definition of usefulness outside of a specific task could be the accurate recreation of instances of the sample data, minimizing reconstruction error and increasing the maximum likelihood of samples. Relying on our understanding of the world and how data are organized and used across algorithms we can make many additional general-purpose assumptions about how data are organized—*meta-priors*: e.g. the sharing of explanatory factors across a broad range of tasks; the hierarchical organization of explanatory factors; the distributed nature of representations; the likelihood that probability mass concentrates on manifolds in a high-dimensional space; that local variation on manifolds preserve information about class differences; that for any observation only a small subset of possible factors are relevant (see [7]). Fundamentally, all of these meta-priors *disentangle* the generative factors of the sample and provide a way to test these factors experimentally. Disentangled representation, produce latent variables which are sensitive to changes in a specific generative factor and remain relatively invariant to changes in others [7]. For example, if we wanted to model images of people, we know that hair color and height are independent and in a good representation these two factors would be represented in non-overlapping variables. On the other hand, the color of the right eye and the color of the left eye, being highly correlated, could be reasonably represented by a single variable.

References

[1] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Networks. *arXiv e-prints*, page arXiv:1406.2661, Jun 2014.

[2] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

[3] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In Eric P. Xing and Tony Jebara, editors, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 1278–1286, Bejing, China, 22–24 Jun 2014. PMLR.

[4] Tian Qi Chen, Xuechen Li, Roger B Grosse, and David K Duvenaud. Isolating sources of disentanglement in variational autoencoders. In *Advances in Neural Information Processing Systems*, pages 2610–2620, 2018.

[5] Shengjia Zhao, Jiaming Song, and Stefano Ermon. Learning hierarchical features from deep generative models. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 4091–4099. JMLR. org, 2017.

[6] Arthur Gretton, Karsten Borgwardt, Malte J Rasch, Bernhard Scholkopf, and Alexander J Smola. A kernel method for the two-sample problem. *arXiv preprint arXiv:0805.2368*, 2008.

[7] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, Aug 2013.