Patterns

CellPress
OPEN ACCESS

Article

Decontextualized learning for interpretable hierarchical representations of visual patterns

Robert Ian Etheredge,[1,2,3,8,*] Manfred Schartl,[4,5,6,7] and Alex Jordan[1,2,3]

[1]Department of Collective Behavior, Max Planck Institute of Animal Behavior, Konstanz, Germany
[2]Center for the Advanced Study of Collective Behavior, University of Konstanz, Konstanz, Germany
[3]Department of Biology, University of Konstanz, Konstanz, Germany
[4]Centro de Investigaciones Científicas de las Huastecas Aguazarca, A.C., Calnali, Hidalgo, Mexico
[5]Developmental Biochemistry, Biocenter, University of Würzburg, Würzburg, Bavaria, Germany
[6]Hagler Institute for Advanced Study, Texas A&M University, College Station, TX, USA
[7]Xiphophorus Genetic Stock Center, Texas State University San Marcos, San Marcos, TX, USA
[8]Lead contact
*Correspondence: rianetheredge@gmail.com
https://doi.org/10.1016/j.patter.2020.100193

THE BIGGER PICTURE We present a fully featured approach to studying natural images that integrates analytical, virtual, and experimental approaches. Our framework, decontextualized hierarchical representation learning (DHRL), overcomes the limitations of small datasets typical of studies in the natural sciences, enabling the application of unsupervised deep learning models to questions where sample data are much more limited.

DHRL captures more complex features and achieves state-of-the-art interpretability scores and improved latent variable interpretation techniques. The representation provided can be used to perform a range of virtual experiments, transforming the way we study natural color patterns and removing the necessity for less explicit, and sometimes ethically problematic, experimental approaches.

Development/Pre-production: Data science output has been rolled out/validated across multiple domains/problems

SUMMARY

Apart from discriminative modeling, the application of deep convolutional neural networks to basic research utilizing natural imaging data faces unique hurdles. Here, we present decontextualized hierarchical representation learning (DHRL), designed specifically to overcome these limitations. DHRL enables the broader use of small datasets, which are typical in most studies. It also captures spatial relationships between features, provides novel tools for investigating latent variables, and achieves state-of-the-art disentanglement scores on small datasets. DHRL is enabled by a novel preprocessing technique inspired by generative model chaining and an improved ladder network architecture and regularization scheme. More than an analytical tool, DHRL enables novel capabilities for virtual experiments performed directly on a latent representation, which may transform the way we perform investigations of natural image features, directly integrating analytical, empirical, and theoretical approaches.

INTRODUCTION

A key motivation for the expanded use of deep convolutional neural networks (CNNs)[1] lies in their capacity to outperform classical computer vision approaches on discriminative tasks.[2] In the life sciences, researchers are leveraging CNNs in a broad range of domain-specific applications, such as the automated tracking of animal movement,[3–5] the detection and classification of cell lines,[6–8] and the mining of genomics data.[9] The ability to represent complex features in an algorithmically useful way (*expressivity*, Box 1) underlies the success of deep networks. This ability to capture feature complexity, which is unparalleled inv traditional computer vision approaches, would suggest their usefulness across a range of analytical pathways. Nonetheless, the application of CNNs to unsupervised descriptions of natural image features has been much more limited due to the low

Box 1. Key terms and definitions in context

Amortized inference: an efficient approximation of maximum likelihood training, mapping samples to distributions. In variational autoencoders (VAEs), amortized inference is performed by the inference/encoder network.
Decontextualized sample learning: "decontextualized learning" is a term borrowed from psychology concerned with language learning in children, in which new word definitions are learned away from a here-and-now context. We use this analogy here to describe the process of breaking the natural feature contexts within the sample data and using generated "decontextualized" samples as training data to our inference model as part of the proposed training procedure (see Decontextualized sample generation, under Experimental procedures).
Disentanglement: the degree to which independent factors of variation in the sample data are represented by independent variables in the latent code. This is a key meta-prior with importance to building interpretability into the latent code. This contrasts with *entangled* latent variables, where multiple independent factors of variation are represented in a single latent variable.
Explaining away (in generative models): units at lower layers become coupled to those at higher layers, and sampling from one unit must cause a change in how all other units update their state. In VAEs, this results in latent variables that depend on some, or even all, other latent variables.
Expressivity (also, *capacity*): the relative amount of complexity (in terms of the functional relationship between inputs and outputs) that can be captured by an approach. For example, convolutional neural networks have much higher expressivity than linear transformations of sample data such as singular value decomposition. This can also be viewed in terms of a latent code parameterized by a more (or less) complex inference model. Here, we use network depth as a proxy for expressivity.
Hierarchical features: features created from a combination of other lower-level features with increasing spatial scale from a terminal set of atomic features (at the lowest spatial scale). Here, our model architecture achieves this across multiple latent codes wherein features produced by the inference model of each prior latent code are combined with those produced from the next, increasingly expressive, inference models. The combined set of features is then used to create the higher-level latent code (see Variational ladder autoencoder, under Experimental procedures).
Information preference problem: in VAEs, when the generative (decoder) learns to generate outputs with high likelihood without reference to the latent code provided by the inference model. The result is an uninformative latent code, like the noise vector used for generative modeling in generative adversarial networks.
Mean-field assumption (of pixel-wise comparison): unlike feedforward convolutional networks, generative models typically require only the preservation of local features to generate realistic output and do not preserve scale or translation invariance, or feature contexts, without reference to surrounding features.
Meta-priors: general (not task specific) assumptions about how data are organized and used by algorithms, which can be enforced during training. Examples include sparsity, spatial and temporal coherence, and the presence of manifolds in the high-dimensional space (see Box 2 for those used in our approach).
Mutual information: a metric describing the amount of information held in one variable (the latent code) informs us about another variable (sample data).

numbers of training samples available to most investigations and the difficulties in interpreting their outputs. As such, approaches built on traditional computer vision algorithms[10–13] continue to dominate despite their comparatively diminished capacity.

Here we address the hurdles to applying these more expressive models to basic research outside of discriminative tasks, providing a highly extensible new framework for the integrated study of natural image features. In doing this, we identify the key functionalities such a framework; it should: (1) provide a useful representation that disentangles factors of variation along a set of interpretable axes; (2) capture feature contexts and *hierarchical feature relationships* (Box 1); (3) incorporate existing knowledge of feature importance and relationships between samples when available; (4) allow for statistical inference of complex traits; and (5) provide direct connections between approaches (i.e., it should allow for integrating analytical, experimental, and theoretical approaches).

In contrast to discriminative models, unsupervised learning seeks to find unknown patterns in data and offers an alternative approach to compression, clustering, and feature extraction using deep networks. Generative modeling techniques, i.e., generative adversarial networks (GANs)[14] and variational autoencoders (VAEs),[15,16] have been especially effective in representing the complexity of natural images and generating photorealistic examples.

In addition to the increased expressivity provided by using stacks of convolutional layers, VAEs offer an intuitive approach to analysis. An extension of variational inference, VAEs combine an inference model with a generative model. The inference model, or encoder, performs *amortized inference* (Box 1) to approximate the posterior distribution over a low-dimensional set of latent variables ($q_\phi(z|x)$). The generative model, or decoder, is used to generate samples conditioned on the latent code ($p_\theta(x|z)$). Instead of optimizing on a specific discriminative task, the objective function in VAEs can take on a variety of forms, which should maximize the likelihood of the data conditioned on the latent variables (e.g., reconstruction error) and minimize the divergence between latent variables and the prior. VAEs provide a means to evaluate sample likelihoods, estimate feature distributions, and generate novel samples.

Despite these qualities, however, which in themselves make VAEs a strong basis for an approach to investigating natural features, several outstanding issues limit their development and application; these include: (1) the information preference

Box 2. Combining existing approaches to address VAE shortfalls

Despite their promise, several outstanding issues limit the development of variational autoencoders (VAEs) for research applications. These issues include: (1) the information preference problem, (2) the restrictive mean-field assumption of reconstruction error metrics, (3) the explaining away of variables between layers, and (4) the entanglement of factors of variation in the latent representation (see Box 1 for definitions). The relative importance of each of these outstanding issues varies, but recent work has proposed several potential solutions to one or more of these outstanding issues. These solutions include changes to the divergence metric used[19,20,21] and specialized model architectures.[22,23] Outside of research on VAEs, there have been general approaches proposed for measuring large-scale perceptual distances,[24,25] decreasing dependence on local features,[26] and capturing the dependence between components of complex features,[27] which can also be applied in the context of generative modeling.

Here, we combine these key contributions to provide a robust basis for inference and generative modeling. We use a VAE with ladder model architecture (VLAE), which proposes to mitigate the explaining away problem and encourage the disentanglement of factors of variation in the latent encoding based on feature complexity (i.e., spatial scale).[22] Using VLAEs, we create multiple latent codes with increasing expressivity. How we measure divergence of latent distributions from the prior can lead to a trade-off between inference and data fit and lead to an uninformative latent code. Here we choose an information-preserving (see *mutual information*, Box 1) latent regularization technique: maximum mean discrepancy (MMD).[28,29] In contrast to the Kullback-Leibler (KL) divergence (the most commonly used divergence metric across VAEs), MMD does not suffer from variance over estimation (overfitting) or an uninformative latent code.[20] In contrast to KL divergence, MMD makes less-restrictive assumptions about the independence of samples. Previous work has focused on adjustments to KL divergence (e.g., β-VAEs);[19,21] these approaches come with increased overhead in terms of additional hyperparameters and training approaches (e.g., KL annealing).[30] Finally, while VLAEs provide a basis for capturing more complex features and MMD provides for a less-restrictive, information-preserving latent code, the choice of reconstruction loss may still undermine this by emphasizing local feature importance (the restrictive mean-field assumption of pixel-wise losses). We balance the effect of pixel-wise error using an additional, perceptual loss function[24] calculated on generated output (note that removing pixel-wise error entirely can lead to poor reconstruction quality). Commonly used for neural style transfer,[31,32] perceptual loss functions balance the effects of local features with more abstract measures of visual similarity calculated across spatial scales. Using this novel combination of techniques, together with powerful encoder and decoder models (see additional details under Experimental procedures) gives a modern basis for investigating natural images and color patterns.

problem, (2) the restrictive mean-field assumption of reconstruction error metrics, (3) the explaining away of variables between layers, and (4) the entanglement of factors of variation (see Box 2). Although several approaches have been proposed to address specific shortcomings of VAEs (Table 1), they must be integrated into a unified approach. They also require better tools for building interpretability and meaningful extensions to other approaches. Another lingering concern is how to apply these approaches to modestly sized datasets. Whereas many proposed techniques have been developed using large datasets, such as CelebA[17] (>200,000 samples) or dSprites[18] (>700,000 samples), typical sample sizes in the life sciences are many orders of magnitude smaller.

Here, we develop a novel preprocessing and pretraining technique, *decontextualized learning* (or decontextualized hierarchical representation learning, DHRL) (Box 1; Decontextualized sample generation, Experimental procedures). Inspired by GAN chaining, decontextualized learning uses the restrictive assumptions of generator networks outlined in Box 2 as an advantage, relaxing the natural covariances in the data by generating decontextualized training samples. When used as part of pretraining, this alleviates the drawbacks of using small datasets by enriching sample variance, encouraging the disentanglement of generative factors, allowing for better estimates of sample likelihood.

We also develop a novel approach for quantifying the attribution of latent variables on generated image features via gradient integration.[35] Whereas several metrics have been proposed for assessing latent codes in terms of disentanglement,[19,36,37] these metrics rely on access to labeled samples, well-defined features, or the specific requirements of image classification competitions (e.g., the Unsupervised and Transfer Learning Challenge[38]). Without labeled samples, traversals of the latent code are often necessary.[15,19,20,22,39] For most practical investigations in the natural sciences, feature definitions and labels may not exist.

Table 1. Desired characteristics of an integrative tool for investigations of natural image data, general representation learning meta-priors, and previously proposed enforcement strategies

Desired characteristic	Representation learning meta-prior[33]	Example approach
Disentangling factors of variation	Limited number of shared factors of variation	Latent regularization[19,21]
Capturing spatial relationships	Hierarchical organization of representation	Hierarchical model architecture[22]
Incorporating existing domain knowledge	Local variation on manifolds	Structured latent codes[34]
Connect analyses and experiments	Local variation on manifolds	Generative models[14–16]
Inference	Probability mass and local variation on manifolds	Variational inference[15]

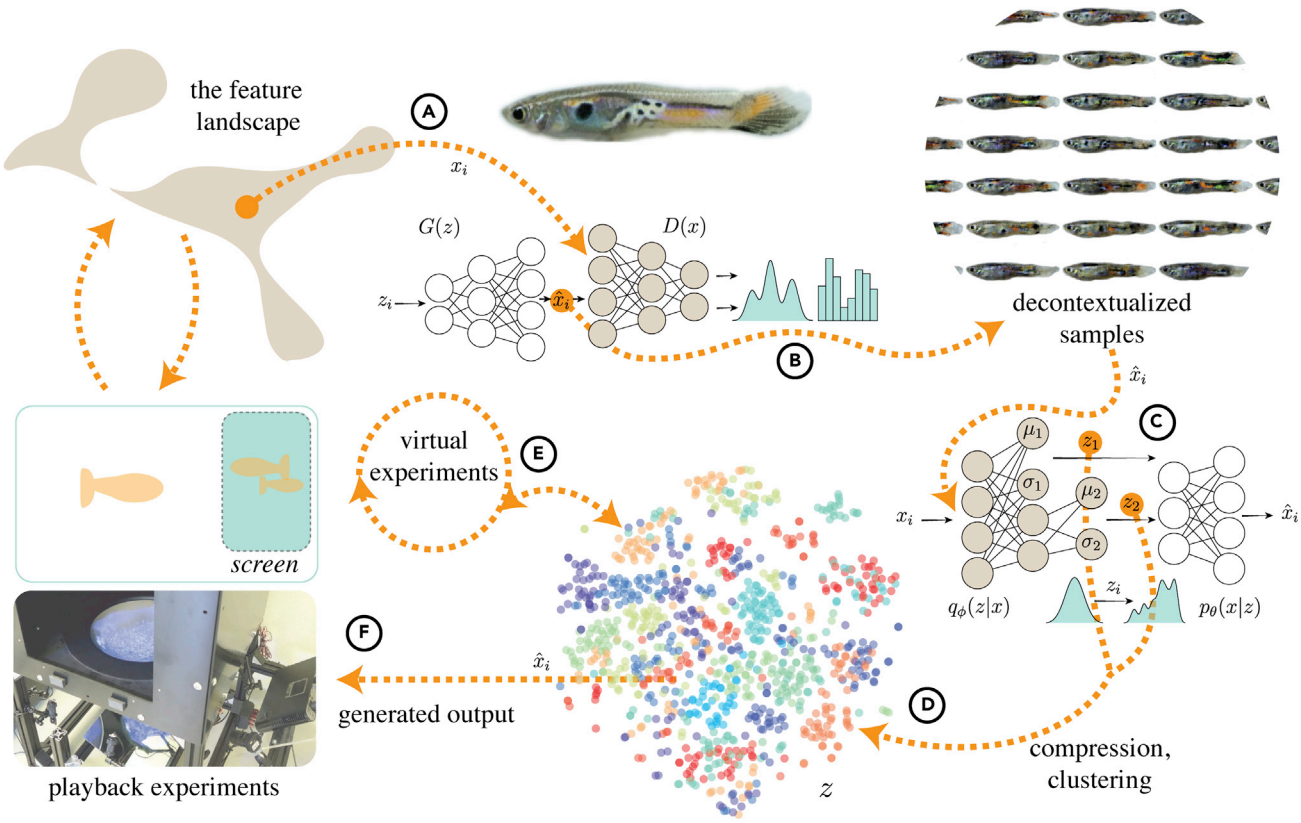

Figure 1. DHRL overview

(A) Many patterns (e.g., male guppy ornaments) consist of combinations of several elements that have hierarchical relationships, spatial dependence, and feature contexts, which may hold distinct biological importance. (B) In our proposed framework, small sample sizes are supplemented using decontextualized samples, which are generated in a preprocessing step using a generative adversarial network, which learns image statistics sufficient to produce novel out of sample examples. This model can be used to produce an unlimited number of novel samples and relaxes to covariance between unrelated samples. Both increased sample sizes and increased variance across categories can be advantageous for disentangling generative features. (C) Decontextualized samples are used to pretrain our specialized model. Based on a variational ladder autoencoder, we use a specific combination of meta-prior enforcement strategies to capture a hierarchy of features (which combines low-level features across spatial scales) and disentangle factors of variation in interpretable ways. The learned distribution over these latent variables can be used for a range of analytical and experimental applications downstream. We can (D) define a color-pattern space or (E) interface with downstream models such as evolutionary algorithms, and even (F) produce photorealistic outputs to be used directly in playback experiments and immersive virtual reality (image credit: loopbio.com). By addressing existing shortcomings of related approaches, our framework provides a robust and integrated framework for investigating natural visual stimuli.

However, more than just qualitative interpretations, the feature attribution approach presented seeks to provide a quantitative, localized metric of latent variables. This formalizes the latent traversal approach without the necessity of labeled data and reduces the influence of our own biases on those assessments (Latent feature attribution and disentanglement, Experimental procedures).

Finally, to show the extensibility and power of this framework we develop and perform synthetic experiments connecting analytical, virtual, and empirical approaches. We demonstrate this in application to the study of animal color patterns, which underlies investigations in sensory ecology, cognitive neuroscience, collective behavior, and evolution. There are many practical and ethical barriers to the study of evolution, which have historically been both disruptive and costly, or in some cases completely intractable. Here we outline how, by using the representations provided by this technique, we can be more explicit about the constraints of an evolutionary model and provide direct connections between analytical, virtual, and experimental approaches to test those assumptions more effectively. In Figure 1, we outline the overall framework and how it may be used in research.

RESULTS

Decontextualized training and DHRL

We first generate decontextualized samples (Figure S1) using an InfoGAN[34] model architecture. This approach uses our prior knowledge about sample relationships to improve the quality of generated samples. In each of our sample datasets (guppies, n = 987, and butterflies, n = 9,531), there are known subcategories relating to subspecies and varieties. Although prior sample knowledge is not required for using DHRL, here we use these categories to inform decontextualized sample preprocessing (Decontextualized sample generation, Experimental procedures). We also use these categories for quantifying the degree of disentanglement

Table 2. Disentanglement and completeness metrics for VAE, β-VAE, and VLAE (our model) across datasets compared with the combined DHRL approach

Model-dataset	$D(z_1)$, $C(z_1)$	$D(z_2)$, $C(z_2)$	$D(z_3)$, $C(z_3)$	$D(z_4)$, $C(z_4)$
VAE-guppies (n = 987)	–	–	–	0.25, 0.24
VAE-butterflies (n = 9,531)	–	–	–	0.51, 0.24
β-VAE-guppies	–	–	–	0.33, 0.31
β-VAE-butterflies	–	–	–	0.70, 0.57
Our model-guppies	0.29, 0.32	0.12, 0.13	0.13, 0.16	0.56, 0.66
Our model-butterflies	0.64, 0.60	0.67, 0.55	0.63, 0.51	0.88, 0.60
Our model + DHRL-guppies	0.14, 0.16	0.18, 0.23	0.31, 0.39	0.90, 0.95

and completeness of the latent code (Table 2) (Latent feature attribution and disentanglement, Experimental procedures), and interpreting sample clusters across approaches.

The value in generating decontextualized samples and the overall method comes in two forms: (1) reducing overfit to a limited amount of sample data by providing an informative pretraining dataset and (2) breaking the correlation between features and exaggerating the variance in sample data. This is needed to produce latent variables that are interpretable and useful for informing our understanding of study systems and connecting across approaches.

Across studies,[19,36,37,21,33] a score of latent variable disentanglement, $D(z)$, and the completeness of the latent code, $C(z)$, are used as general measures of interpretability and usefulness of latent representation to downstream tasks. Our procedure of decontextualized training contributes to higher $D(z)$ between factors compared with existing approaches, including VAE[15,16] and β-VAE.[19,21] Although our model architecture (Variational ladder autoencoder, Experimental procedures) outperforms these existing architectures on its own (Table 2), performance is significantly improved by using DHRL ($D(z)$ = 0.56 without the use of decontextualized samples versus $D(z)$ = 0.90 using the same model with the decontextualized pretraining procedure).

We can provide some insight into these results by highlighting the motivation for the technique. Small sample sizes, which are common for many studies, tend to capture less of the true variance in the data (by the reduced sampling frequency). At the same time, we turn the drawbacks of existing generative models (being overreliant on local features) to our advantage. Specifically, in contrast to our inference model, where we want to reduce the effects of an overreliance on local features, here in a pretraining step, we use them to create decontextualized samples, which have increased variability between samples at higher spatial scales. The decontextualized guppy samples used here show 210% more variance between versus within classes compared with our original samples, which showed only 67.3% more variance between versus within samples measured by a perceptual distance metric across two spatial scales. Similar increases in variance were also seen at both local (190% versus 52%) and non-local scales (230% versus 82%). For pretraining the DHRL model, we use 32,000 decontextualized samples.

Next, we visualize the latent representation produced by our DHRL model in comparison to two alternative approaches, raw pixel differences and perceptual similarity score (which have intuitive interpretations). We project each of these to a two-dimensional embedding for visualization via t-distributed stochastic neighbor embedding (t-SNE)[40,41] (Figure 2). Whereas raw pixel embeddings tend to highlight local similarity in terms of pixel similarity, the perceptual distance metric favors more abstract similarity between samples (although this is more difficult to interpret). In contrast to these approaches, DHRL balances local and higher-level similarity between samples. In Figure 2Ai–iii, we see that local neighborhoods of samples are much more interpretable and stable using our approach (Figure 2Aiii, Aiv). Using perceptual distance alone can often lead to unexpected results. For example, using perceptual similarity scores results in expected clusters to be scattered across the embedded space (red X's, Figure 2Ci). These clusters are better preserved in raw pixel embeddings and DHRL. However, while using raw pixel distributions captures color and contrast similarity (Figure 2Cii), it misses subclusters based on shape and pattern (higher-level features), which are captured by DHRL (Figure 2Ciii).

The latent space produced by DHRL consists of distinct sets of latent variables. Each successive set of latent variables ($z_1, \ldots, z_4$) combines outputs of the previous latent variable set with a more expressive model (i.e., using additional stacked convolutional layers) (Variational ladder autoencoder, Experimental procedures). The increased expressivity at each level captures a hierarchy of features over increasing spatial scales. In Figure S2, we confirm this by qualitatively comparing embeddings and sample-nearest-neighbor pairs across the four sets of latent encodings. In z_1 (left columns), samples are organized along axes corresponding to color and contrast similarity (S2a). Likewise, nearest neighbors have strong color similarity (S2b, left). At higher levels (z_2, z_3) (b, middle), local pattern similarity appears to better describe nearest-neighbor pairs. In z_3 we find the highest similarity within body patterns, and at the highest level, z_4, overall shape and orientation similarities dominate nearest-neighbor pairs (S2b, right).

Because model optimization is performed using decontextualized samples from outside the original datasets, we can get an unbiased measure of sample likelihood of the original samples without training. In our guppy dataset, one of the subgroups carries a rare trait not seen in any other subgroup (a distinctive melanization pattern). These samples cluster together in the latent space produced by DHRL (Figure 2Aiv) and have low sample likelihood estimated given the parameters of the trained model (Figure 3).

As mentioned above, pretraining using decontextualized samples produced the best disentanglement scores ($D(z)$ = 0.90) (Table 2). For completeness we also compare with results from related work, VAE[15,16] and β-VAE[19,21] + Kullback-Leibler (KL) annealing,[30] which have shown previous state-of-the-art levels of disentanglement when used on large datasets (e.g., CelebA[17] and dSprites[18]). Using our butterfly dataset (n = 9,531), we achieved disentanglement scores of $D(z)$ = 0.51 (VAE) and $D(z)$ = 0.70 (β-VAE with KL annealing), and our smaller guppy dataset (n = 987) showed a similar trend ($D(z)$ = 0.25 and $D(z)$ = 0.33, respectively). In both cases, even without the use of decontextualized samples, our model architecture showed improved

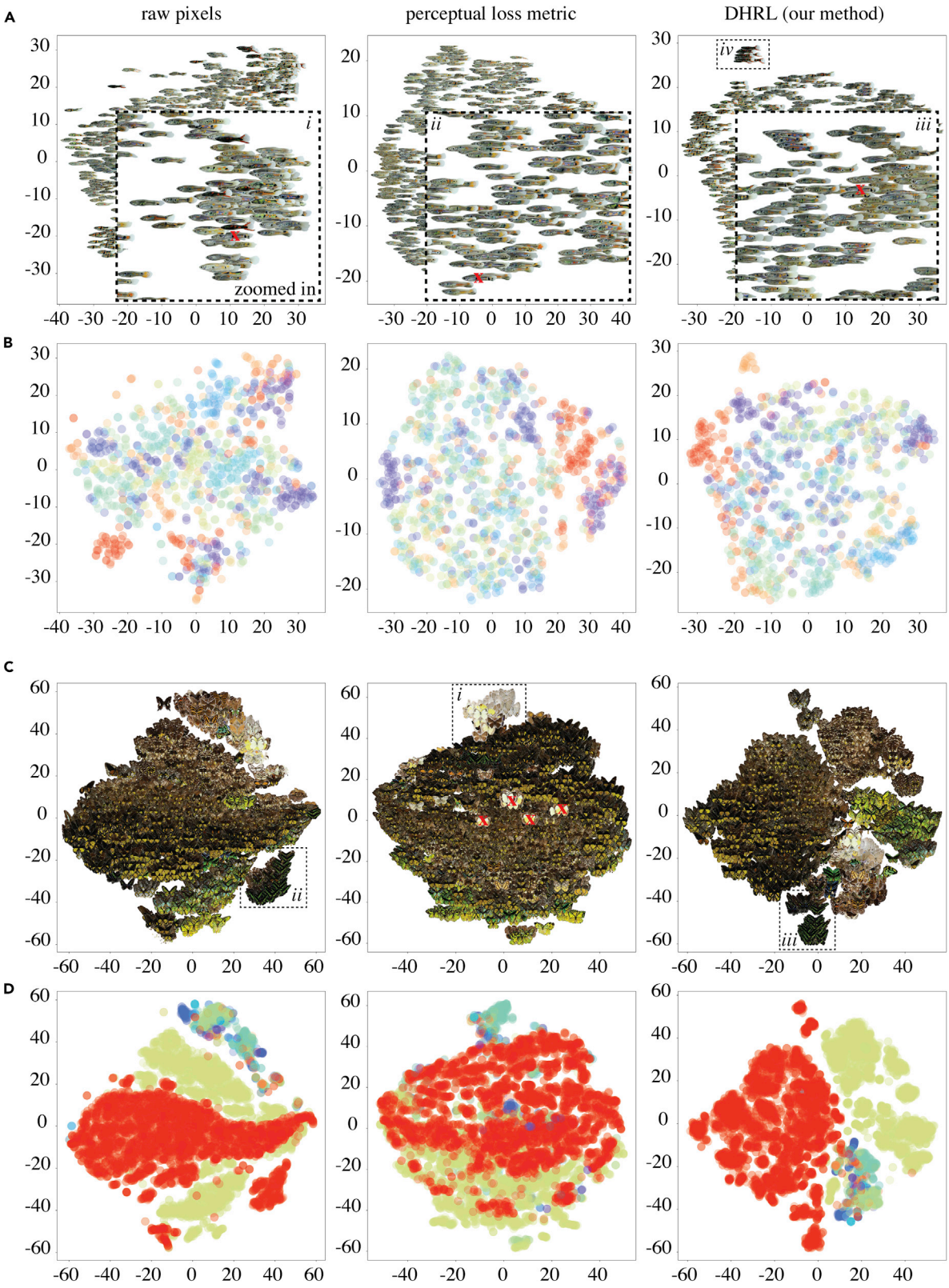

Figure 2. Qualitative comparisons

Two-dimensional t-SNE embedding of raw pixel distributions (left column, A–D), using a perceptual similarity score[24,42] (middle column, A–D), and the latent variables provided by our framework (right column, A–D). (A) Guppy images, (B) guppy class labels, (C) butterfly images, and (D) butterfly class labels. Colors

(legend continued on next page)

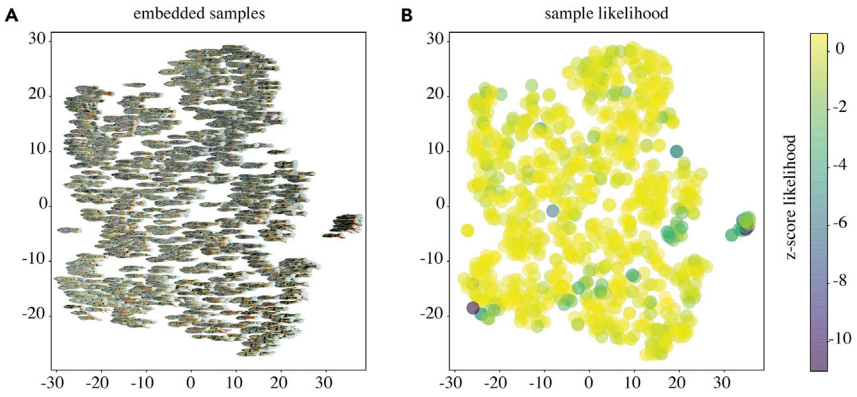

Figure 3. Sample likelihood estimates

(A) Embedded samples.

(B) Normalized (standard score) likelihood estimates for each sample. Rare samples with distinct color patterns also show reduced likelihood (e.g., the highly melanized samples, which cluster to the left of the plot).

disentanglement results, $D(z)$ = 0.56 (guppies) and $D(z)$ = 0.81 (butterflies) compared with VAE and β-VAE + KL annealing (Table 2), on these small datasets. However, again, the best results were seen using the entire DHRL framework ($D(z) = 0.90$).

Latent variable feature attribution

To provide quantitative support for interpreting latent variables, we demonstrate the use of latent feature attributions (Latent feature attribution and disentanglement, Experimental procedures) on two examples. In Figures 4A we visualize one variable (z_{13}) of the trained DHRL model. Qualitatively, we find that the same latent variable controls the relative intensity of green color patches across individuals. Latent feature attributions help quantify that directly on generated output (heatmaps, Figure 4), providing a quantitative output to compare with our qualitative interpretations. Again, looking at a single variable (z_{27}) of the trained butterfly model (Figures 4B) we find that this latent variable controls the size of yellow patches on the lower wings relative to the size of yellow patches on the upper wings, and when patches are not present this variable has no effect (Figure 4B, upper right). Whereas we present only two variables here (for clarity), further investigation of latent variables using this feature attribution technique can be performed using the provided toolset (https://github.com/ietheredge/VisionEngine/blob/master/notebooks/FeatureAttribution.ipynb).

Latent evolution

So far, we have demonstrated the benefits of DHRL and latent feature attribution generally with comparison to related techniques and highlighted how the latent variables can be used to fulfill the first four requirements of a framework for investigating natural image data: disentangle factors in interpretable ways, capture feature relationships across scale, incorporate existing knowledge when available, and allow for statistical inference of complex traits (Introduction). Next, we demonstrate how we may extend this same framework to integrate analysis with virtual and experimental approaches, achieving the fifth aim of our framework (provide direct, meaningful connections between approaches).

Using the latent representation of our trained DHRL model of guppy ornaments as input, we conducted a pilot study on ornament evolution. We defined a fitness function based on general findings from the guppy literature: more orange, higher contrast males are preferred by females.[43] We initialize the population using a random sample of our sample embedding (900 samples); we then simulate 500 generations under selection for more orange, higher contrast males, with an additional constraint on sample likelihood to produce bounded solutions, weighted equally. By projecting the latent representation of each generation, we found large shifts in the distribution of traits in the population (Figure 5A) (Video S1). After 500 generations, we observed exaggerated and more numerous orange and black patches in novel configurations compared with the initial population (Figure 5B). We confirmed this quantitatively, finding a significant increase in the population means of orange (generation 1, 4.3×10^{-6}, 1.4×10^{-5}; generation 500, 3.6×10^{-4}, 3.9×10^{-4}; bootstrapped 95% CI) and within-body contrast (generation 1, 4.2×10^{-3}, 5.4×10^{-3}; generation 500, 6.4×10^{-2}, 6.7×10^{-2}; bootstrapped 95% CI) (Figure 5C). At generation 500, instead of a single peak, two novel solutions are optimized (Figure 5A). Investigating the values of the latent variables over generations reveals two distinct latent factors driven to fixation in the population under these selective forces (Figure S3).

In terms of efficiency, using a single Titan Xp GPU with 12 GB memory, we could simulate a population size of 1,000 individuals in an average of 19.5 s per generation. This provides the possibility of directly testing the results of virtual experiments using video playback and immersive virtual reality (see Video S2, where we visualize a continuous trajectory of sample evolution from generation 1 to generation 500).

DISCUSSION

Supervised discriminative learning algorithms are already becoming an integral tool for researchers across disciplines, achieving state-of-the-art performance. In contrast, unsupervised

indicate unique subgroups for each sample (guppy variety and butterfly species). For raw pixel embeddings (left column), we see clusters based on overall color and contrast similarity, but these similarities are tough to interpret in some cases (e.g., box Ai); this is also true for perceptual loss metrics (Aii). In (Ai–iii), red X's indicate the same sample across the three approaches. Overall, our approach shows much more visible consistency in local neighborhoods (Aiii, Aiv). Using a perceptual loss, clusters are often unintuitive or omit similar samples. For example, in (C), middle, the samples marked by red X's should intuitively cluster with the samples in box (Ci), which is observed both in raw pixel embeddings (left) and using our approach (right). On the other hand, relying on raw pixels alone misses larger scale differences. Samples in box (Cii) that have similar color and contrast form a single cluster, whereas, by using our approach, we find additional clusters of the same samples based on wing shape (Ciii).

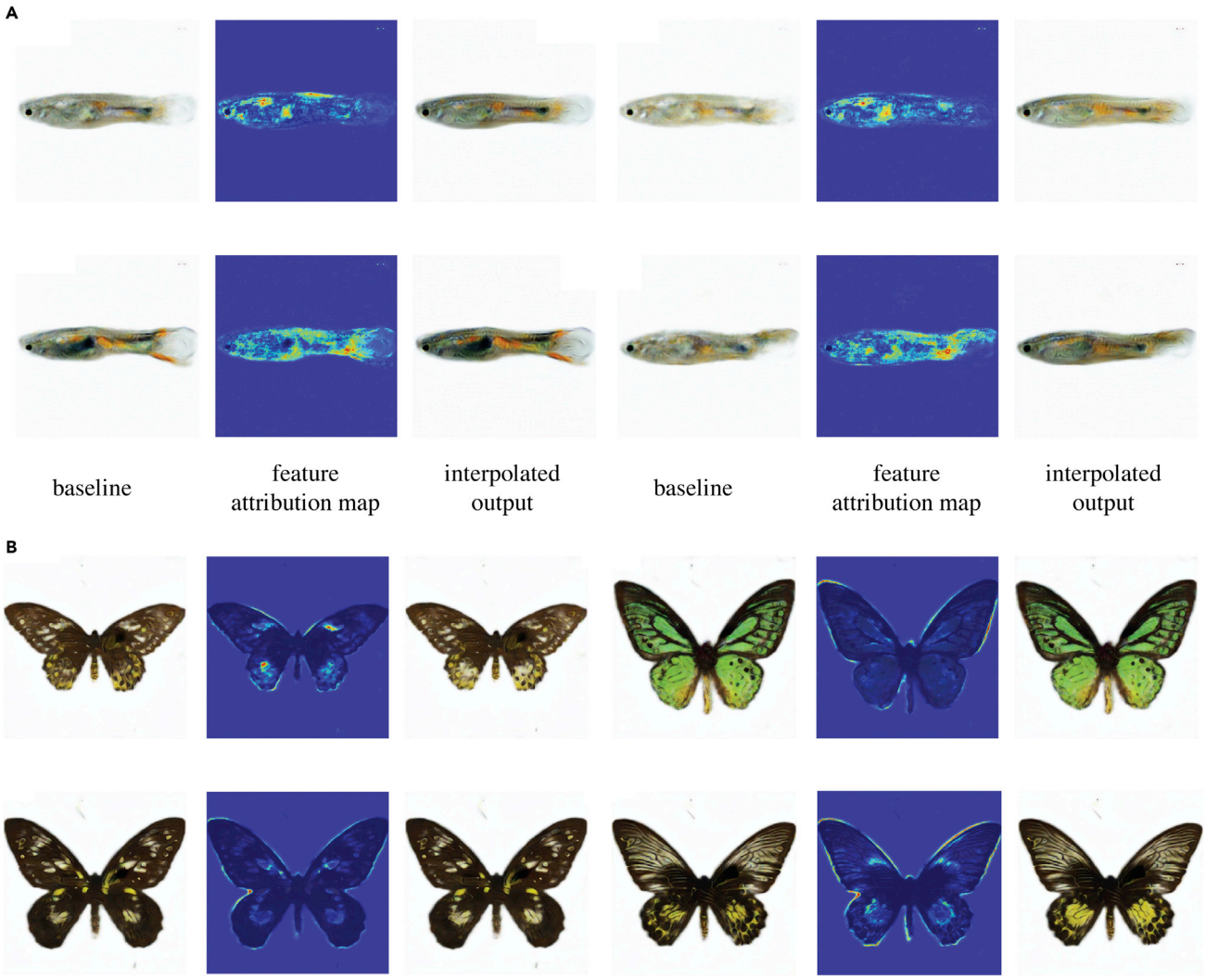

Figure 4. Latent variable feature attribution

Examples of latent variable feature attribution of a latent variable of the trained variational model (chosen at random as a demonstration) across four random samples. Left and right images are generated outputs at the minimum and maximum values and provide a means to qualitatively assess latent variables. Heatmaps provide complementary, quantitative results to compare with those assessments. For example, in (A) we investigate the latent variable $\boldsymbol{z}_{13}$, which controls the intensity of green color patches in generated samples (guppies). In (B), generated butterfly examples, latent variable $\boldsymbol{z}_{37}$ controls the relative size of light-yellow patches in generated samples. Heatmap values have been normalized using a standard score. Images to the left are generated with the latent feature set to its lowest value in the sample and those to the right with the highest value in the sample. Further investigation of additional latent variables using this feature attribution technique can be performed using the provided toolset.

generative modeling approaches are still a relatively young area of research, but may prove to be even more transformative by providing more direct connections to hypothesis testing.

In the life sciences, datasets are typically much smaller compared with those typically used for machine learning research (e.g., compared with ImageNet[44]). Our DHRL model achieves state-of-the-art disentanglement scores and outperforms existing techniques (Table 2) using small datasets, providing a valuable new technique to enable the use of realistic sample sizes. The prevalence of small datasets is perhaps one of the underlying reasons for the continued dominance of approaches built on traditional computer algorithms,[10–13] which are far less data hungry compared with deep networks. Although many tools have been developed built on classical computer vision approaches,[45–54] fundamental gaps remain in building quantitative descriptions of complex features. Existing approaches fail to capture the full complexity of many color patterns because the algorithms themselves are insufficiently expressive. By providing a means for scaling small datasets in informative ways, we relax this hurdle to using more expressive techniques (i.e., CNNs and deep generative models).

The expressivity of deep networks is a primary motivation for researchers seeking to adopt them. DHRL balances local and large-scale feature similarity and captures spatial relationships of different scales across increasingly expressive sets of latent variables (Figure S2), relationships that other approaches do

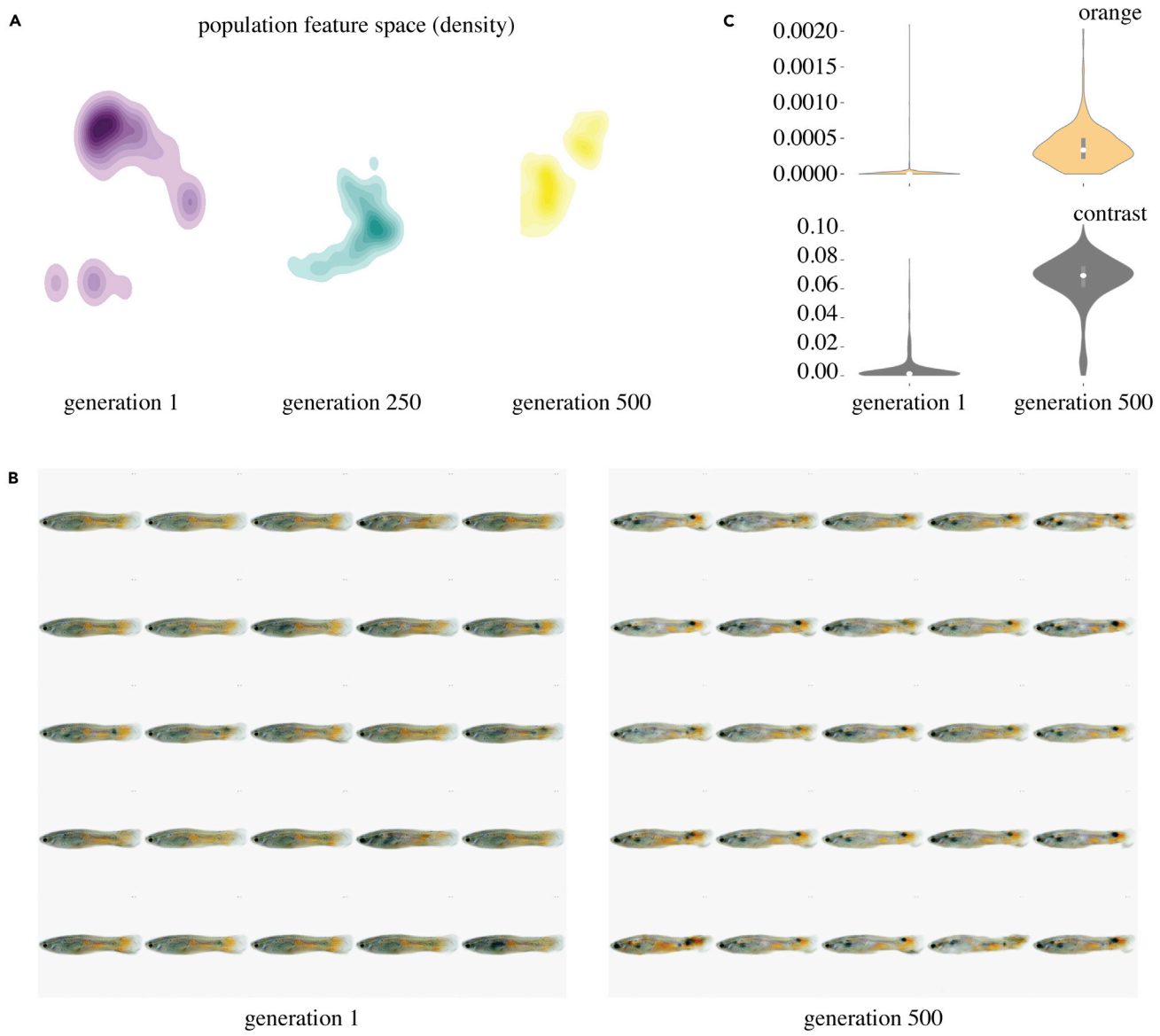

Figure 5. Virtualizing evolution experiments

(A) Kernel density plot of samples over generations 1, 250, and 500 selecting orange ornaments and contrast. After 500 generations the population has shifted from the initial sample distribution, finding two peaks that maximize the fitness function.

(B) Samples of initial parent population, left, with the highest fitness, compared with those with the highest fitness after 500 generations under selection, right. Samples of later generations show higher numbers of brighter orange and dark melanized patches and increased within-body contrast.

(C) Percentages of orange and contrast (the two selective forces acting over generations) increase over generations, confirming qualitatively the results seen in (B). From generation 1 to generation 500 we see a marked increase in both metrics. Constrained by model expectations, the selective forces have produced an increased number of spots near the tail, more pronounced caudal fins, and increased orange patches often see in natural populations.

not account for (including alternative approaches built on deep CNNs).[10–13,15,16,19,21] Although the importance of these spatial relationships will vary across investigations, they may be particularly important in studies of natural features. In terms of both feature context[55–60] and the perceptions of shape, motion, and attention,[61–65] spatial relationships between pattern elements have been shown to hold key biological importance. Moreover, in the brain, perception is hierarchically organized,[66] and representations made at higher levels of the visual cortex influence the perception of low-level features.[67,68]

In Box 3, we discuss some specific outstanding questions surrounding the study of signal evolution that can benefit from the use of DHRL, both analytically and experimentally. Whereas here, we focus on experimental evolution as a particularly exciting and transformative application to studies using natural image data, manipulations of the latent representation can take on many forms (e.g., learning experiments). Using a broad range of domain-specific manipulations, we can design complex real-time assays that leverage playback experiments to directly test outcomes.

Box 3. Applying DHRL to the study of evolution

This platform may be used to address many outstanding questions regarding the functional significance of color pattern traits; here, we provide some examples to inspire future work. (1) What are the constraints on the evolvability of a given trait? By identifying the topographical relationship between different traits within the color pattern space, we can test predictions about the selective forces acting on them related to their geometric relationships; e.g., the axes of variation in traits meant to communicate viability should show increased orthogonality compared with co-occurring traits that have evolved under a Fisherian process.[69–74] (2) Categorical perception is an important perceptual mechanism for understanding the evolution of color signals.[75] But in systems where color patterns are used for mimicry[76–78] or novelty, investigating the boundaries between complex traits is fundamental. By performing traversals across the distribution of the latent variables, interpolating between samples can allow for tests of continuous[79] versus categorical perception[80] of complex traits. (3) Many color pattern traits have evolved under selective pressure from multiple receivers; e.g., both females and predators shape the diversity of male guppy ornaments.[81] Establishing these types of evolutionary trade-offs is difficult and often requires large, highly disruptive manipulations such as translocation experiments.[82] Using evolutionary models similar to the ones presented here, researchers can simulate multiple fitness landscapes and evolutionary trajectories simultaneously to perform a broad range of virtual experiments. Importantly, while each of these examples places analytical, experimental, or virtual results at the center, by using the platform presented here, they maintain direct connections across approaches. Furthermore, they can incorporate existing techniques[48–52,83,84] as image preprocessing routines, during playback, or as constraints on virtual experiments.

More than compressing complex traits into a low-dimensional space for analysis, because this approach is generative it can transform the way researchers design investigations. By performing virtual experiments on the same representation used for analysis, researchers may test analytical results with virtual experiments, and empirically, by using virtual reality playback experiments or observational studies (Video S2) based purely on analytical results (without any human biases). This can better inform experimental manipulations and have a lasting impact on creating high-throughput approaches for hypothesis generation and offline prototyping of experimental manipulations. This can be especially valuable in study systems that currently rely on highly disruptive manipulations for studying traits (e.g., using introduction, translocation, and manipulation experiments),[82,85–87] which often raise legal, ethical, and conservation issues.[82] Using approaches like the one presented here allows researchers to be both much more explicit about the parameters of a given evolutionary model (or any experimental paradigm) and more prudent with the use of animal subjects.

EXPERIMENTAL PROCEDURES

Resource availability

Lead contact

Further information and requests for resources and reagents should be directed to and will be fulfilled by the lead contact, R. Ian Etheredge (rianetheredge@gmail.com).

Materials availability

Guppy images were collected from a maintained stock at the University of Würzburg 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[88] using a Canon D600 digital camera. Images were downsampled and center cropped to a final size of 256 × 256 pixels. The dataset consisted of 977 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.org/10.5519/qd.gvq3p7xq, doi.org/10.5519/qd.pw8srv43). This dataset consisted of 9,531 RGB images.

For each dataset, we segmented samples from the background using a customized object segmentation network adapted from Caelles et al.[89] For each dataset we annotated eight samples to train the segmentation network. All samples were cropped and resized to 256 × 256 and placed on a transparent background (RGBA). For calculating the perceptual loss during training, images were translated to three-channel images with a white background using alpha blending. Updated links to original data repositories can be accessed at github.com/ietheredge/VisionEngine/README.md.

Data and code availability

All key methods (Figure 6) and models were implemented using Tensorflow 2.2 and can be accessed via the github repository, including installation and evaluation scripts to reproduce our results. Instructions for creating new data loaders for training new datasets using this method can be found in the repository readme file. The original data have been deposited for both guppies and butterflies.

Network specifications and methods

Decontextualized sample generation

The basis of our approach to increase disentanglement for small datasets relies on the use of decontextualized samples for pretraining. Here, we use a modified InfoGAN,[34] which can incorporate prior knowledge about the sample data via the number of discrete latent codes (e.g., providing 10 categorical latent codes for generating handwritten digits). Whereas prior knowledge about samples is not strictly required, even without prior knowledge, InfoGAN provides a more stable training procedure for generating samples.[34] Because it is available, we incorporate prior knowledge about our samples of male guppy ornamentation images by providing a 32-class discrete latent code. These 32-classes represent the 32 individual tanks, unique subsets of the overall sample, with shared traits related to guppy ornamentation patterns (Figure S1). Although we do not quantify them here, early results suggested that by increasing the number of discrete latent codes we were able to produce more consistent images. For future work, we suggest the number of discrete codes be chosen with respect to meaningful structure in the data (e.g., number of species, subgroups, etc.) or left at one (1) if none are known.

GAN training and VAE training are performed in separate steps so that models are not jointly optimized. The generated samples from the trained GAN model are used as training data to a variational model (Figure 1) with a hierarchical model architecture,[22] which consists of 10 latent variables across four codes ($z_1, \ldots, z_4$) with increasing expressivity (Variational ladder autoencoder, below).

InfoGAN

We use an unsupervised approach to disentangle discrete and continuous latent factors adapted from Chen et al. (InfoGAN),[34] which modifies the minimax game typically used for training GANs, such that:

$$\min_{G,Q}\max_{D} V_I(D,G,Q) = V(D,G) - \lambda L_I(G,Q), \qquad \text{(Equation 1)}$$

where $V(D,G)$ is the original GAN objective introduced by Goodfellow et al.[14] and $L_I(G,Q)$ approximates the lower bound of the mutual information

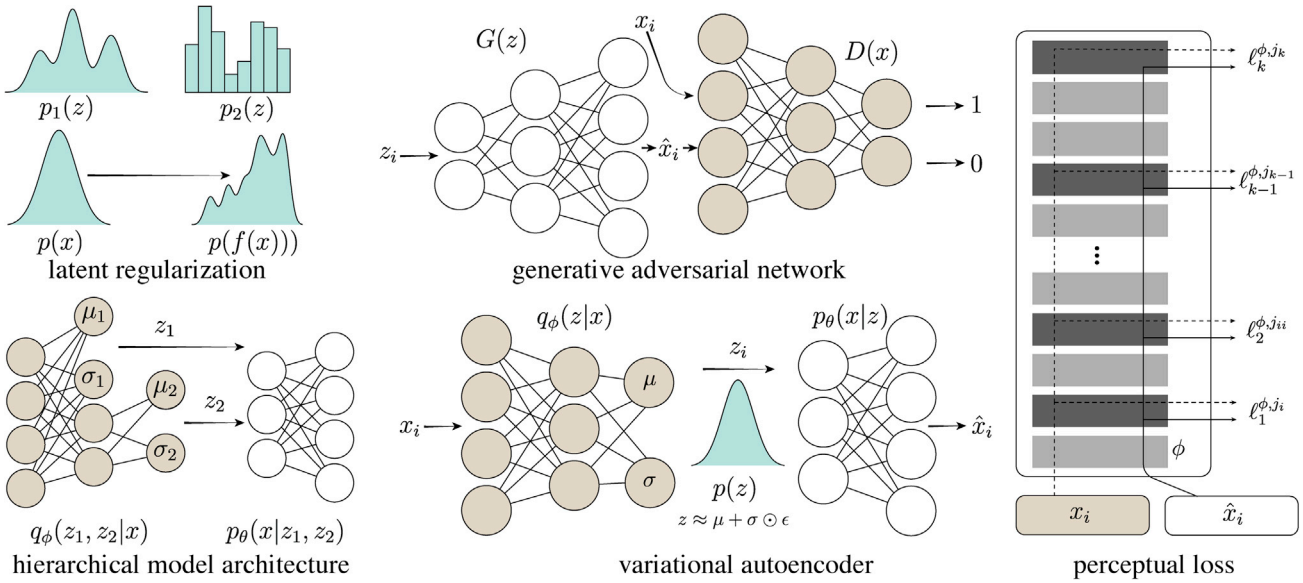

Figure 6. Key methods

Top left: latent variable priors can be parameterized by either continuous or discrete variables. In the model used to generate decontextualized samples, we use both categorical and continuous latent codes (top). In our inference model (DHRL) we use an information-preserving regularization approach, which is less restrictive and allows for more complex posterior estimates (bottom). Top middle: example structure of a generative adversarial network. Here, a noise vector, $\boldsymbol{z_i}$ is input to the generator network $\boldsymbol{G}(\boldsymbol{z})$, which produces a reconstructed output $\hat{\boldsymbol{x}}_i$. A real sample, $\boldsymbol{x_i}$, and generated sample, $\hat{\boldsymbol{x}}_i$, are subsequently passed through a separate discriminator network $\boldsymbol{D}(\boldsymbol{x})$, which determines if the sample is real (1) or generated (0). For decontextualized sample generation, the latent encoding of generated samples is optimized by an additional network, $\boldsymbol{Q}$, which shares all convolutional layers with $\boldsymbol{D}$. Bottom left: the generic architecture of a variational ladder autoencoder. Multiple latent spaces $(\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_k)$ are learned, with each successive latent variable space $(\boldsymbol{z_j})$ layer having increasing expressivity and abstraction to hierarchically organized complex features across spatial scales. Bottom middle: structure of a variational autoencoder. $\boldsymbol{x_i}$ and $\hat{\boldsymbol{x}}_i$ are an example input and its reconstructed output; the probabilistic encoder or inference model, $\boldsymbol{q}_\phi(\boldsymbol{z}|\boldsymbol{x})$, performs posterior inference, learning shared model parameters, ϕ, across samples, approximating the true posterior distribution. The probabilistic decode, $\boldsymbol{p}_\theta(\boldsymbol{Z}|\boldsymbol{X}), \boldsymbol{p}_\theta(\boldsymbol{X}|\boldsymbol{Z})$, learns a joint distribution of the encoded space, $\boldsymbol{Z}$, and the data space $\boldsymbol{x}$. The low-dimensional bottleneck, $\boldsymbol{Z}$, is a distribution of latent variables capable of reconstructing sample inputs, parameterized by a vector of means $\boldsymbol{\mu}$ and standard deviations $\boldsymbol{\sigma}$. The noise term ε allows for the parameters of this multivariate distribution to be optimized using back-propagation, known as the reparameterization trick. Right: example perceptual loss models use a pretrained network, ϕ, e.g., VGG-16.[92] Two samples are input to the model and the activations across one or more layers are used as outputs for each sample. The distance between these outputs provides a measure of visual similarity that does not rely on pixel-wise differences, emphasizing higher-level similarity. Perceptual loss functions can be used as a stand-alone transfer-learning approach to find perceptual differences between samples or as part of any network as an additional or alternative reconstruction loss.

$I(c; G(z, c))$ using Monte Carlo sampling such that $L_I(G,Q) \leq I(c; G(z,c))$.[34] Like the generator G and discriminator D, Q is parameterized as a neural network and shares all convolutional layers with D.

Both discrete $Q(c_d|x)$ and continuous latent codes $Q(c_c|x)$ are provided, with continuous latent codes treated as a factored Gaussian distribution. Importantly, InfoGAN does not require supervision and no labels are provided.[36]

We substitute the original generator and discriminator models from Chen et al.[34] with the architecture described in Redmon et al.[90] and increase the flexibility of the latent code, providing additional continuous and discrete latent codes. For guppy experiments, we provide 2 continuous and 32 discrete codes as part of the model, and we used a 100-unit random noise vector as input to the generator.

Variational ladder autoencoder

In contrast to hierarchical architectures,[91,23] we learn a hierarchy of features by using multiple latent codes with increasing levels of abstraction introduced by Zhao et al.,[22] i.e., $q_\phi(z_1, \ldots, z_L|x)$. The expressivity of z_i is determined by its depth. The encoder $q_\phi(z_1, \ldots, z_L|x)$ consists of four blocks such that:

$$H_l = G_l(H_{l-1}), \quad \text{(Equation 2)}$$

$$z_l \sim \mathcal{N}(\mu_l(H_l), \mathbf{I}), \quad \text{(Equation 3)}$$

where H_l, G_l, and μ_l are neural networks. For our encoder model, G_l is a stack of convolutional, batch normalization, and leaky rectified linear unit activation (Conv-BN-LeakyReLU), and we stack four Conv-BN-LeakyReLU blocks for each G_l with increasing numbers of channels for each subsequent convolutional layer, i.e., N-channels/2, N-channels, N-channels, N-channels * 2, where N-channels is 16, 64, 256, and 1024 for G_1, G_2, G_3, and G_4, respectively. We apply spectral normalization to all convolutional layers (see below). Because we want to preserve feature localization, we use average pooling followed by a squeeze-excite (SE) block to apply a context-aware weighting to each channel (see below).

Similarly, the decoder, $p_\theta(x|z_1, \ldots, z_L)$, is composed of blocks such that:

$$\tilde{z}_l = U_l([\tilde{z}_{l+1}; V_l(z_l)]), \quad \text{(Equation 4)}$$

where $[.;.]$ denotes channel-wise concatenation. Parallel to G_l, blocks in the encoder U_l are composed of Conv-BN-ReLU blocks (note the use of ReLU and not LeakyReLU in the decoder) with decreasing number of channels in each convolutional layer, i.e., N-channels * 2, N-channels, N-channels, N-channels/2, where N-channels is 1024, 256, 64, and 16. No spectral normalization wrappers or SE layers are applied in the decoder.

We chose four sets of latent variables after Zhao et al.,[22] which showed this provided reasonable spatial separation for face images (CelebA[17]). However, more or fewer sets can also be used. Although we do not test those effects here, we would expect increased (or decreased) resolution between scales based on the number of codes. In practice, many latent codes are not practical due to the increased model complexity.

Reconstruction loss

We minimize the negative log likelihood of the sample data by minimizing the mean squared error between input and output, jointly optimizing the reconstruction loss for each sample x:

$$\mathcal{L}_{pixel-wise} = E_{p_{data}}(x)E_{q_\phi}(z|x)[\log p_\theta(x|z)]$$

$$= \frac{1}{n}\sum_{i=1}^{n}(x_i - p_\theta(q_\phi(x_i)))^2 \quad \text{(Equation 5)}$$

To relax the restrictive mean-field assumption, which is implicit in minimizing the pixel-wise error, we jointly optimize the similarity between inputs and outputs using intermediate layers of a pretrained network, VGG16,[92] as feature maps.[31,24] Here we calculate the Gram matrices of feature maps, which match the feature distributions of real and generated outputs for each layer as:

$$\mathcal{L}_{perceptual} = \sum_{l=1}^{L}\frac{\frac{1}{n}\sum_{i=1}^{n}\left(G^l_{ab}(x_i) - G^l_{cd}(p_\theta(q_\phi(x_i)))\right)^2}{L}, \quad \text{(Equation 6)}$$

where

$$G^l_{ab} = \frac{\sum_{cd}F^l_{cda}(x)F^l_{cdb}(x)}{CD}, \quad \text{(Equation 7)}$$

for feature maps F_a and F_b in layer l across locations c and d. This measures the correlation between image filters and is equivalent to minimizing the distance between the distribution of features across feature maps, independent of feature position.[25]

The combined reconstruction loss is a weighted sum of the perceptual loss and pixel-wise error:

$$\mathcal{L}_{reconstruction} = \alpha\mathcal{L}_{perceptual} + \beta\mathcal{L}_{pixel-wise}, \quad \text{(Equation 8)}$$

where α and β are hyperparameters controlling the influence of each loss term. Here we set $\alpha = 1 \times 10^{-6}$ and $\beta = 1 \times 10^{-5}$ to balance the contribution of reconstruction terms with variational loss (see below).

Maximum mean discrepancy

We use the maximum mean discrepancy (MMD) approach[28,29] to maximize the similarity between the statistical moments of $p(z)$ and $q_\phi(x)$ using the kernel embedding trick:

$$MMD(p(z)|q_\phi(z)) = \mathbb{E}_{p(z),p(z')}[k(z,z')] + \mathbb{E}_{q_\phi(z),q_\phi(z')}[k(z,z')] - 2\mathbb{E}_{p(z),q_\phi(z')}[k(z,z')], \quad \text{(Equation 9)}$$

using a Gaussian kernel, $k(z,z')$, such that:

$$k(z,z') = e^{-\frac{\|z-z'\|^2}{2\sigma^2}}, \quad \text{(Equation 10)}$$

to measure the similarity between $p_\theta(z)$ and $q_\phi(z)$ in Euclidean space. We measured similarity using multiple kernels with varying degrees of smoothness, controlled by the value of σ^2, i.e., multi-kernel MMD (MK-MMD),[29] with varying bandwidths: $\sigma^2 = 1 \times 10^{-6}$, 1×10^{-5}, 1×10^{-4}, 1×10^{-3}, 1×10^{-2}, 1×10^{-1}, 1, 5, 10, 15, 20, 25, 30, 35, 100, 1×10^3, 1×10^4, 1×10^5, and 1×10^6.

Weighing the influence of MMD kernel differences on the combined objective function is controlled by the hyperparameter λ applied across each latent code, giving the combined objective:

$$\mathcal{L}_{total} = \left(\sum_{l}^{L}\lambda MK - MMD(q_\phi(z_l)|p(z_l))\right) + \mathcal{L}_{reconstruct}, \quad \text{(Equation 11)}$$

where L is the number of hierarchical latent codes and z_l is the n-dimensional latent code and the prior, $p(z_l) = \mathcal{N}(0, I)$ and $\mathcal{L}_{reconstruction}$ as defined in Equation 8. Here, we set $\lambda = 1$.

The three model hyperparameters α, β, and λ are chosen to balance their relative influence on the objective during training (in reference to λ); here we set $\alpha = 1 \times 10^{-6}$ and $\beta = 1 \times 10^5$. We recommend these be at similar values when using images of the same size ($256 \times 256 \times 3$, $\alpha = 1 \times 10^{-6}$) and number of layers used to calculate the perceptual loss (three output layers, $\beta = 1 \times 10^5$). If changing these values, users should balance their relative contribution to training loss (observed during training).

Improving encoder and decoder features

In addition to further relaxing the contribution of pixel-wise error, we use established ways to increase feature context, stabilize training, and increase sample likelihood (unpublished data).

Squeeze-and-excitation networks[27] were proposed to improve feature interdependence by adaptively weighting each channel within a feature map based on the filter relevance by applying a channel-wise recalibration. Here we apply SE layers prior to each variational layer on outputs from the Conv-BN-LeakyReLU blocks for each G_l such that each embedding z_l may better capture features with cross-channel dependencies. Each SE layer consists of a global average pooling layer, which averages channel-wise features followed by two fully connected layers with ReLU activations, the first with number of input channels/16 and the second with the same size as the number of input channels. Finally, a sigmoid, "excite," layer assigns channel-wise probabilities, which are then multiplied channel-wise with the original inputs.

Spectral normalization has been proposed as a method to prevent exploding gradients when using rectified linear units to stabilize GAN training via a global regularization on the weight matrix of each layer as opposed to gradient clipping to provide bounded first derivatives (the Lipschitz constraint[93]). We perform spectral normalization on each activation layer of both the encoder and the decoder.

Adding a denoising criterion has been shown to yield better sample likelihood by learning to map both training data and corrupted inputs to the true posterior, providing more robust training for out-of-sample data:[26,94]

$$\tilde{x} \sim \mathcal{M}_{\mathcal{D}}(\tilde{\boldsymbol{x}}|x), \quad \text{(Equation 12)}$$

where we implement the mapping between real samples to noisy sample $\mathcal{M}_{\mathcal{D}}$ via a noise layer, which samples a corrupted input $\tilde{x}$ from input x before passing $\tilde{x}$ to the encoder $q_\phi(z|\tilde{x})$. We apply random binomial noise (salt and pepper) to 10% of pixels.

Latent feature attribution and disentanglement

Understanding the importance of features for model predictions is an active area of research. Integrated gradients, introduced by Sundararajan et al.,[35] assigns feature importance, determining causal relationships between predictions and image features by summing the gradients along paths between x' and x:

$$IG_i(x) ::= (x_i - x'_i) \times \int_{\alpha=0}^{1}\frac{\partial\mathcal{P}(x' + \alpha \times (x - x'))}{\partial x_i}d\alpha. \quad \text{(Equation 13)}$$

We adapt this procedure to investigate the contribution of each latent variable parameter z_i, where we use a baseline z, an encoding of a single sample x, and iterated z_j while holding all other z_l constant and summing the gradients of the decoder $p_\theta(x|z)$ such that:

$$IG_i^{approx}(p_\theta(x|z^i)) ::= (p_\theta(x|z^i)_i - p_\theta(x|z^{i'})_i) \times \sum_{k=1}^{m}\frac{\partial\mathcal{P}\left(p_\theta(x|z^{i'}) : z^{i'}_j = z^{i'}_j + \frac{k}{m} \times \left(z^i_j - z^{i'}_j\right)\right)}{\partial p_\theta(x|z^i)_i} \times \frac{1}{m}, \quad \text{(Equation 14)}$$

where j is the axis of latent code being interpolated, i is the individual feature (pixel), $p_\theta(x|z)$ is the reconstructed output, $p_\theta(x|z')$ is the baseline reconstructed output, k is the perturbation constant, and m is the number of steps in the approximation of the integral. We use the Riemann sum approximation of the integral over the interpolated path $\mathcal{P}$, which involves computing the gradient in a loop over the inputs for $k = 1, \ldots, m$. Here, we use $m = 300$ and $k = 2max(|z|)$ for each z^i starting from a baseline $p_\theta(x|z^{i'}) : z_j = -\max(|z|)$.

We use the technique developed by Eastwood and Williams[37] for assessing disentanglement, measuring the relative entropy of latent factors for predicting class labels. We measure disentanglement of D_i of each latent code as measured by $D_i = (1 - H_K(P_i))$, where H_K is the entropy and P_i is the relative importance of the generative factor. We also include a metric of completeness, C_i, approximating the degree to which the generative factor is captured by a

single latent variable, where $C_j = (1 - H_D(P_j))$, where P_j is the unweighted contribution of generative factors.[37] Here, in the absence of labeled features, we use species (butterflies), breeding line variants (guppies), and predicted class of the generative model for each model as approximate class labels (one class). This approximation naturally overestimates D_i and underestimates C_j, as there is overlap between classes in terms of visual features. While Eastwood and Williams[37] propose a third term to evaluate representations, I, to measure the relative informativeness, we found that this value was highly correlated to the choice of the hyperparameter λ used for latent regularization.

Simulating evolution on the latent space

For demonstrating an example virtual experiment, we use a genetic algorithm, with a parent population of 1,000 random samples, evolved over 500 generations. Parent samples are random initialized across the latent variables of each latent code. Fitness was calculated as an equally weighted sum of the total percentage of pixels within two ranges (orange RGB (0.9, 0.55, 0) > RGB (1.0, 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.[43,95] During each generation the predicted fitness for each sample in the population was measured by the fitness of the nearest neighboring value in the reference table (for processing speed). To simulate weak selective pressure on the fitness function, we drew 500 random parent subsamples weighted by their proportional fitness. An additional 200 samples were drawn, without the proportional fitness weighting. Together, 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 randomly mutated, one by drawing from a random normal distribution and the other by replacing an existing value with zero (like destabilizing and stabilizing mutations). The next generation thus consisted of 1,000 samples, 700 parent samples + 300 offspring. This process was repeated for 500 generations.

SUPPLEMENTAL INFORMATION

Supplemental information can be found online at https://doi.org/10.1016/j.patter.2020.100193.

ACKNOWLEDGMENTS

We would like to thank members of the Department of Collective Behavior, Max Planck Institute of Animal Behavior, and Center for the Advanced Study of Collective Behavior, University of Konstanz, for comments on earlier versions of the manuscript; the Max Planck Computing and Data Facility for use of computational resources; and four anonymous reviewers for their constructive feedback, which greatly improved the manuscript. This research was funded by the Deutsche Forschungsgemeinschaft (German Research Foundation) under Germany's Excellence Strategy-EXC 2117-422037984 and by the Max Planck Institute of Animal Behavior.

AUTHOR CONTRIBUTIONS

Conceptualization, R.I.E; Methodology, R.I.E; Software, R.I.E; Validation, R.I.E; Formal Analysis, R.I.E; Investigation, R.I.E.; Resources, M.S. and A.J.; Data Curation, R.I.E.; Writing – Original Draft, R.I.E; Writing – Review & Editing, R.I.E, M.S., and A.J.; Visualization, R.I.E; Supervision; A.J.; Funding Acquisition, A.J.

DECLARATION OF INTERESTS

The authors declare no competing interests.

Received: August 26, 2020
Revised: November 8, 2020
Accepted: December 17, 2020
Published: January 21, 2021

REFERENCES

1. LeCun, Y., Kavukcuoglu, K., and Farabet, C. (2010). Convolutional networks and applications in vision. In Proceedings of 2010 IEEE International Symposium on Circuits and Systems (IEEE), pp. 253–256.
2. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., and Zisserman, A. (2010). The Pascal Visual Object Classes (VOC) challenge. Int. J. Comput. Vis. *88*, 303–338.
3. Bozek, K., Hebert, L., Mikheyev, A.S., and Stephens, G.J. (2018). Towards dense object tracking in a 2D honeybee hive. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (IEEE), pp. 4185–4193.
4. Mathis, A., Mamidanna, P., Cury, K.M., Abe, T., Murthy, V.N., Mathis, M.W., and Bethge, M. (2018). DeepLabCut: markerless pose estimation of user-defined body parts with deep learning. Nat. Neurosci. *21*, 1281–1289.
5. Pereira, T.D., Aldarondo, D.E., Willmore, L., Kislin, M., Wang, S.S.-H., Murthy, M., and Shaevitz, J.W. (2019). Fast animal pose estimation using deep neural networks. Nat. Methods *16*, 117–125.
6. McQuin, C., Goodman, A., Chernyshev, V., Kamentsky, L., Cimini, B.A., Karhohs, K.W., Doan, M., Ding, L., Rafelski, S.M., Thirstrup, D., et al. (2018). CellProfiler 3.0: Next-generation image processing for biology. PLoS Biol. *16*, e2005970.
7. Falk, T., Mai, D., Bensch, R., Çiçek, Ö., Abdulkadir, A., Marrakchi, Y., Böhm, A., Deubner, J., Jäckel, Z., Seiwald, K., et al. (2019). U-Net: deep learning for cell counting, detection, and morphometry. Nat. Methods *16*, 67–70.
8. Riba, J., Schoendube, J., Zimmermann, S., Koltay, P., and Zengerle, R. (2020). Single-cell dispensing and 'real-time' cell classification using convolutional neural networks for higher efficiency in single-cell cloning. Sci. Rep. *10*, 1–9.
9. Poplin, R., Chang, P.-C., Alexander, D., Schwartz, S., Colthurst, T., Ku, A., Newburger, D., Dijamco, J., Nguyen, N., Afshar, P.T., et al. (2018). A universal SNP and small-indel variant caller using deep neural networks. Nat. Biotechnol. *36*, 983–987.
10. Hough, P.V. (1962). Method and Means for Recognizing Complex Patterns (Google Patents).
11. Harris, C.G., Stephens, M., and others. (1988). Alvey Vision Conference (Citeseer)), pp. 10–5244.
12. Lowe, D.G. (1999). Object recognition from local scale-invariant features. Proceedings of the Seventh IEEE International Conference on Computer Vision, *2* (IEEE), pp. 1150–1157.
13. Lowe, D.G. (2004). Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. *60*, 91–110.
14. Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A.C., and Bengio, Y. (2014)..Generative adversarial networks arXiv:1406.2661
15. Kingma, D.P., and 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, Y. Bengio and Y. LeCun, eds. (arXiv).
16. Rezende, D.J., Mohamed, S., and Wierstra, D. (2014). In Proceedings of the 31st International Conference on Machine Learning, E.P. Xing and T. Jebara, eds. (PMLR), pp. 1278–1286.
17. Liu, Z., Luo, P., Wang, X., and Tang, X. (2015). Deep learning face attributes in the wild. In Proceedings of International Conference on Computer Vision (ICCV), pp. 3730–3738.
18. Matthey, L., Higgins, I., Hassabis, D., and Lerchner, A. (2017). dSprites: Disentanglement Testing Sprites Dataset. https://github.com/deepmind/dsprites-dataset/.
19. Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., and Lerchner, A. (2017). In 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings.

20. Zhao, S., Song, J., and Ermon, S. (2019). InfoVAE: balancing learning and inference in variational autoencoders*33* (AAAI Press), pp. 5885–5892, The 33rd AAAI Conference on Artificial Intelligence.
21. Chen, T.Q., Li, X., Grosse, R.B., and Duvenaud, D.K. (2018). Isolating sources of disentanglement in variational autoencoders. Adv. Neural Inf. Process. Syst. *31*, 2610–2620.
22. Zhao, S., Song, J., and Ermon, S. (2017). Learning hierarchical features from deep generative models. In Proceedings of the 34th International Conference on Machine Learning, *Volume 70* (JMLR. org), pp. 4091–4099.
23. Sønderby, C.K., Raiko, T., Maaløe, L., Sønderby, S.K., and Winther, O. (2016). Ladder variational autoencoders. Adv. Neural Inf. Process. Syst. *29*, 3738–3746.
24. Johnson, J., Alahi, A., and Fei-Fei, L. (2016). Perceptual losses for real-time style transfer and super-resolution. European Conference on Computer Vision.
25. Li, Y., Wang, N., Liu, J., and Hou, X. (2017). In Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017, C. Sierra, ed. (ijcai.org), pp. 2230–2236.
26. Im, D.I.J., Ahn, S., Memisevic, R., and Bengio, Y. (2017). Denoising criterion for variational auto-encoding framework. In Thirty-First AAAI Conference on Artificial Intelligence, pp. 2059–2065.
27. Hu, J., Shen, L., and Sun, G. (2018). Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7132–7141.
28. Gretton, A., Borgwardt, K., Rasch, M., Schölkopf, B., and Smola, A.J. (2007). A kernel method for the two-sample-problem. Adv. Neural Inf. Process. Syst. *19*, 513–520.
29. Gretton, A., Sriperumbudur, B.K., Sejdinovic, D., Strathmann, H., Balakrishnan, S., Pontil, M., and Fukumizu, K. (2012). Optimal kernel choice for large-scale two-sample tests. Adv. Neural Inf. Process. Syst. *25*, 1205–1213.
30. Fu, H., Li, C., Liu, X., Gao, J., Celikyilmaz, A., and Carin, L. (2019). Cyclical annealing schedule: a simple approach to mitigating KL vanishing, (Long and Short Papers). In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, *Volume 1*, J. Burstein, C. Doran, and T. Solorio, eds. *(Association for Computational Linguistics), pp. 240–250.*
31. Gatys, L., Ecker, A., and Bethge, M. (2016). A Neural Algorithm of Artistic Style. J. Vis. *16*, 326, https://doi.org/10.1167/16.12.326.
32. Gatys, L.A., Ecker, A.S., and Bethge, M. (2015). Texture synthesis using convolutional neural networks. In Proceedings of the 28th International Conference on Neural Information Processing Systems, *Volume 1* (MIT Press), pp. 262–270.
33. Bengio, Y., Courville, A., and Vincent, P. (2013). Representation learning: a review and new perspectives. IEEE Trans. Pattern Anal. Mach. Intell. *35*, 1798–1828.
34. Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., and Abbeel, P. (2016). InfoGAN: interpretable representation learning by information maximizing generative adversarial nets. Adv. Neural Inf. Process. Syst. *29*, 2172–2180.
35. Sundararajan, M., Taly, A., and Yan, Q. (2017). Axiomatic attribution for deep networks. In Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017, D. Precup and Y.W. Teh, eds. (PMLR), pp. 3319–3328.
36. Kim, H., and Mnih, A. (2018). In Proceedings of the 35th International Conference on Machine Learning, J. Dy and A. Krause, eds. (PMLR), pp. 2649–2658.
37. Eastwood, C., and Williams, C.K.I. (2018). A framework for the quantitative evaluation of disentangled representations. In International Conference on Learning Representations.
38. Dauphin, G.M.Y., Glorot, X., Rifai, S., Bengio, Y., Goodfellow, I., Lavoie, E., Muller, X., Desjardins, G., Warde-Farley, D., Vincent, P., et al. (2012). Unsupervised and transfer learning challenge: a deep learning approach. In Proceedings of ICML Workshop on Unsupervised and Transfer Learning, *27*, I. Guyon, G. Dror, V. Lemaire, G. Taylor, and D. Silver, eds., pp. 97–110.
39. Gulrajani, I., Kumar, K., Ahmed, F., Taiga, A.A., Visin, F., Vázquez, D., and Courville, A.C. (2017). Iclr.
40. van der Maaten, L., and Hinton, G. (2008). Visualizing data using t-SNE. J. Mach. Learn. Res. *9*, 2579–2605.
41. Kobak, D., and Berens, P. (2019). The art of using t-SNE for single-cell transcriptomics. Nat. Commun. *10*, 5416.
42. Ezray, B.D., Wham, D.C., Hill, C.E., and Hines, H.M. (2019). Unsupervised machine learning reveals mimicry complexes in bumblebees occur along a perceptual continuum. Proc. Biol. Sci. *286*, 20191501.
43. Houde, A.E. (1987). Mate choice based upon naturally occurring color-pattern variation in a guppy population. Evolution *41*, 1–10.
44. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. (2015). ImageNet large scale visual recognition challenge. Int. J. Comput. Vis. *115*, 211–252.
45. van den Berg, C.P., Troscianko, J., Endler, J.A., Marshall, N.J., and Cheney, K.L. (2020). Quantitative colour pattern analysis (QCPA): a comprehensive framework for the analysis of colour patterns in nature. Methods Ecol. Evol. *11*, 316–332.
46. Gawryszewski, F.M. (2018). Color vision models: Some simulations, a general *n*-dimensional model, and the colourvision R package. Ecol. Evol. *8*, 8159–8170.
47. Tedore, C., and Johnsen, S. (2016). Using RGB displays to portray color realistic imagery to animal eyes. Curr. Zool. *63*, 27–34.
48. Endler, J.A. (1991). Variation in the appearance of guppy color patterns to guppies and their predators under different visual conditions. Vision Res. *31*, 587–608.
49. Endler, J.A. (2012). A framework for analysing colour pattern geometry: adjacent colours. Biol. J. Linn. Soc. *107*, 233–253.
50. Troscianko, J., and Stevens, M. (2015). Image calibration and analysis toolbox - a free software suite for objectively measuring reflectance, colour and pattern. Methods Ecol. Evol. *6*, 1320–1331.
51. Caves, E.M., and Johnsen, S. (2018). AcuityView: an R package for portraying the effects of visual acuity on scenes observed by an animal. Methods Ecol. Evol. *9*, 793–797.
52. Stoddard, M.C., and Osorio, D. (2019). Animal coloration patterns: linking spatial vision to quantitative analysis. Am. Nat. *193*, 164–186.
53. Maia, R., Gruson, H., Endler, J.A., and White, T.E. (2019). New tools for the spectral and spatial analysis of colour in R. Methods Ecol. Evol. *10*, 1097–1107.
54. Stoddard, M.C., Kilner, R.M., and Town, C. (2014). Pattern recognition algorithm reveals how birds evolve individual egg pattern signatures. Nat. Commun. *5*, 4117.
55. Fechner, G.T. (1840). Ann. Phys. *126*, 427–470, https://doi.org/10.1002/andp.18401260703.
56. Fuller, R., and Santos, J.A. (2002). Human Factors for Highway Engineers (Pergamon Amsterdam).
57. Cole, G.L., and Endler, J.A. (2016). Male courtship decisions are influenced by light environment and female receptivity. Proc. Biol. Sci. *283*, 20160861.
58. Nieder, A., Freedman, D.J., and Miller, E.K. (2002). Representation of the quantity of visual items in the primate prefrontal cortex. Science *297*, 1708–1711.
59. Fujita, S., Kitayama, T., Mizoguchi, N., Oi, Y., Koshikawa, N., and Kobayashi, M. (2012). Spatiotemporal profiles of transcallosal connections in rat insular cortex revealed by in vivo optical imaging. Neuroscience *206*, 201–211.
60. Yang, S.C.-H., Lengyel, M., and Wolpert, D.M. (2016). Active sensing in the categorization of visual patterns. Elife *5*, e12215.

61. Thayer, G.H. (1918). Concealing-coloration in the Animal Kingdom: An Exposition of the Laws of Disguise through Color and Pattern: Being a Summary of Abbott H. Thayer's Discoveries (Macmillan Company).
62. Cott, H.B. (1940). Adaptive Coloration in Animals (Oxford University Press).
63. Kelley, L.A., and Kelley, J.L. (2014). Animal visual illusion and confusion: the importance of a perceptual perspective. Behav. Ecol. *25*, 450–463.
64. Merilaita, S., Scott-Samuel, N.E., and Cuthill, I.C. (2017). How camouflage works. Philos. Trans. R. Soc. B: Biol. Sci. *372*, 20160341.
65. Gasparini, C., Serena, G., and Pilastro, A. (2013). Do unattractive friends make you look better? Context-dependent male mating preferences in the guppy. Proc. Biol. Sci. *280*, 20123072.
66. Marr, D. (1982). Vision (MIT Press).
67. Hubel, D.H., and Wiesel, T.N. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. J. Physiol. *160*, 106–154.
68. Pafundo, D.E., Nicholas, M.A., Zhang, R., and Kuhlman, S.J. (2016). Top-down-mediated facilitation in the visual cortex is gated by subcortical neuromodulation. J. Neurosci. *36*, 2904–2914.
69. Darwin, C. (1871). The Descent of Man (D. Appleton and Company).
70. Fisher, R.A. (1915). The evolution of sexual preference. Eugen. Rev. *7*, 184.
71. Lande, R. (1981). Models of speciation by sexual selection on polygenic traits. Proc. Natl. Acad. Sci. U S A *78*, 3721–3725.
72. Kirkpatrick, M. (1982). Sexual selection and the evolution of female choice. Evolution *36*, 1–12.
73. Iwasa, Y., and Pomiankowski, A. (1994). The evolution of mate preferences for multiple sexual ornaments. Evolution *48*, 853–867.
74. Prum, R.O. (2010). The Lande-Kirkpatrick mechanism is the null model of evolution by intersexual selection: implications for meaning, honesty, and design in intersexual signals. Evol. Int. J. Org. Evol. *64*, 3085–3100.
75. Caves, E.M., Brandley, N.C., and Johnsen, S. (2018). Visual acuity and the evolution of signals. Trends Ecol. Evol. *33*, 358–372.
76. Bates, H.W. (1863). The Naturalist on the River Amazons (John Murray), [1910, reprinted 1921].
77. Wallace, A.R. (1877). The colors of animals and plants. Am. Nat. *11*, 641–662.
78. Joron, M., and Mallet, J.L. (1998). Diversity in mimicry: paradox or paradigm? Trends Ecol. Evol. *13*, 461–466.
79. Searcy, W.A., and Nowicki, S. (2005). The Evolution of Animal Communication: Reliability and Deception in Signaling Systems (Princeton University Press).
80. Roff, D.A. (2015). The evolution of mate choice: a dialogue between theory and experiment. Ann. N. Y. Acad. Sci. *1360*, 1–15.
81. Endler, J.A. (1987). Predation, light intensity and courtship behaviour in *Poecilia reticulata* (Pisces: Poeciliidae). Anim. Behav. *35*, 1376–1385.
82. Kawecki, T.J., Lenski, R.E., Ebert, D., Hollis, B., Olivieri, I., and Whitlock, M.C. (2012). Experimental evolution. Trends Ecol. Evol. *27*, 547–560.
83. Endler, J.A., and Mielke, P.W., JR. (2005). Comparing entire colour patterns as birds see them. Biol. J. Linn. Soc. *86*, 405–431.
84. Endler, J.A., Cole, G.L., and Kranz, A.M. (2018). Boundary strength analysis: combining colour pattern geometry and coloured patch visual properties for use in predicting behaviour and fitness. Methods Ecol. Evol. *9*, 2334–2348.
85. Reznick, D.N., and Bryga, H. (1987). Life-history evolution in guppies (poecilia reticulata): 1. phenotypic and genetic changes in an introduction experiment. Evolution *41*, 1370–1385.
86. Reznick, D.A., Bryga, H., and Endler, J.A. (1990). Experimentally induced life-history evolution in a natural population. Nature *346*, 357–359, https://doi.org/10.1038/346357a0.
87. Reznick, D.N., and Ghalambor, C.K. (2005). Selection in nature: experimental manipulations of natural populations. Integr. Comp. Biol. *45*, 456–462.
88. Kemp, D.J. (2008). Female mating biases for bright ultraviolet iridescence in the butterfly Eurema hecabe (Pieridae). Behav. Ecol. *19*, 1–8.
89. Caelles, S., Maninis, K.-K., Pont-Tuset, J., Leal-Taixé, L., Cremers, D., and Van Gool, L. (2017). One-shot video object segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (IEEE), pp. 221–230.
90. Redmon, J., Divvala, S.K., Girshick, R.B., and Farhadi, A. (2016). 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016 (IEEE Computer Society)), pp. 779–788.
91. Bachman, P. (2016). An architecture for deep, hierarchical generative models. Adv. Neural Inf. Process. Syst. *29*, 4826–4834.
92. Simonyan, K., and Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. International Conference on Learning Representations.
93. Miyato, T., Kataoka, T., Koyama, M., and Yoshida, Y. (2018). Spectral normalization for generative adversarial networks. In 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings (OpenReview.net), arXiv:1802.05957.
94. Vincent, P., Larochelle, H., Bengio, Y., and Manzagol, P.-A. (2008). Extracting and composing robust features with denoising autoencoders. In Proceedings of the 25th International Conference on Machine Learning, pp. 1096–1103.
95. Endler, J.A., and Houde, A.E. (1995). Geographic variation in female preferences for male traits in poecilia reticulata. Evolution *49*, 456–468.