INTERPOLATIVE AUTOENCODERS FOR UNSUPERVISED FEW-SHOT IMAGE GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

We aim to build image generation models that generalize to new classes from few examples. To this end, we first investigate the generalization properties of classic image generators, and discover that autoencoders generalize extremely well to new domains, even when trained on highly constrained data. We leverage this insight to produce a robust, unsupervised few-shot image generation algorithm, and introduce a novel training procedure. The resulting interpolative autoencoders synthesize realistic images of novel objects from only a few reference images, and are competitive with VAEs trained directly on the full set of novel classes. Our procedure is simple and lightweight, generalizes broadly, and requires no category labels or other supervision during training.

1 INTRODUCTION

Recent work has produced powerful neural models for image generation. These models can synthesize high-quality (28; 40; 46), diverse (16; 36; 40), and high-resolution (9; 26; 28) images never before seen in the training data. We are approaching the point where we can train effective generative image models for *any* class of object, *given a large training dataset for these classes* (13).

However, this requirement of a large dataset is impractical in many scenarios. For example, an artist or designer might want to use these techniques to help create concept art of futuristic vehicles. Roboticists might want to create a diverse simulated environment that can be used to train agents to be robust in the real world. To enable these applications and others, we need generative modeling techniques capable of synthesizing images from a large, *ever-growing* list of object classes. Few of these classes will have large sets of labeled images easily available, and the vast majority will be *unknown* at the time of training.

We therefore need generative models that can train on one set of image classes, and then generalize to a new class using only a small quantity of new images. This is the problem of *few-shot image generation*. This problem is extremely challenging since it involves tackling two open research problems simultaneously. First, the generative model must be capable of even *representing* novel class images in its latent space (*cross-category generalization*). Second, we must be able to *sample* diverse novel class generations given only a few examples of the novel class (*few-shot synthesis*).

While representations learned by *classification networks* are well known to generalize to new classes (15; 17; 39; 42; 43), we do not know if this is true for generative models. In fact, we find that the latest and greatest generative models *do not generalize*. This is clear in Figure 1, where a Progressive Growing GAN (27) trained on adult faces struggles to even *represent* baby faces. Perhaps because of this generalization challenge, recent attempts at few-shot image generation rely on simplifying assumptions and compromises that are ultimately undesirable. Most existing models tend to require *labeled* datasets of thousands of classes, which can be impractical to collect (14). Others involve substantial computation at test time (11). Yet others end up displaying highly domain-specific behavior, generalizing to a new object class reliably only if it is similar to the training classes (24).

In this paper, we introduce a strong, efficient, *unsupervised* baseline for few-shot image generation that avoids *all* of the above compromises. Our key insight derives from our finding that while the latent space of powerful generative models, such as VAEs and GANs, does not generalize to new classes (see Fig. 1), the representation learned by vanilla autoencoders generalizes extremely well. Unfortunately, autoencoders cannot *synthesize* new novel class images. To remedy this, we introduce a new training method to encourage a more interpretable and informative latent space, which allows for meaningful interpolation. We demonstrate on three different settings (handwritten characters, faces and general objects) that the resulting *Interpolative Autoencoder* achieves *simple, robust, highly general, and completely unsupervised* few-shot image generation.

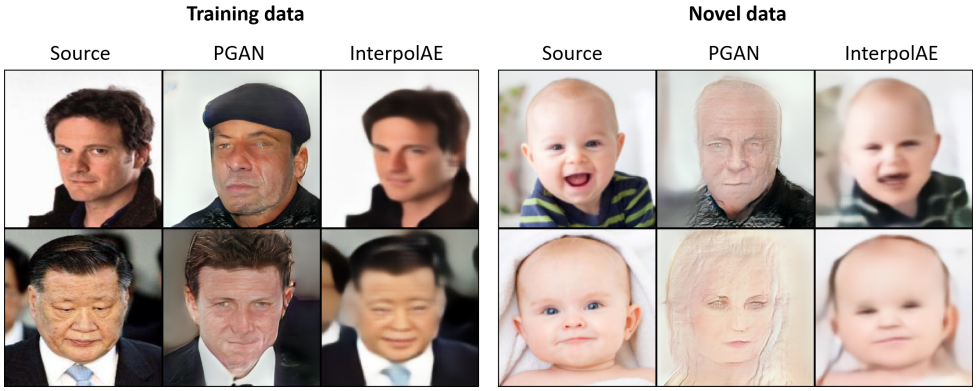

Figure 1: A Progressive Growing GAN - a powerful, high-resolution, state-of-the-art image generator - is trained on Celeb-A and reasonably recovers training images using a learned inversion network followed by latent space optimization (5). However, when we attempt the same reconstruction procedure on images from a novel class (babies), the semantic content is lost: the reconstructed images are clearly not babies (right side, middle column). A simple, six-layer interpolative autoencoder trained on the same dataset, while slightly blurry, successfully preserves the semantic content in both cases (rightmost columns). Details are in appendix, images best viewed digitally.

2 Related Work

2.1 Generative Modeling

Neural generative modeling was initially a mere by-product of unsupervised learning. Autoencoders were originally proposed as a form of learned non-linear data compression, which could then be used for downstream tasks; the generator network was discarded completely (30). This held true even when autoencoders were first applied to image data (22; 37). VAEs do the opposite: by pushing the latent space toward a prior distribution during training, the encoder network can be discarded at test time instead. New images are sampled directly from the prior distribution (29). Subsequent models discard the encoder network entirely. GANs sample directly from a noise distribution and learn to generate images which fool a concurrently-trained real/fake image discriminator (18). GLO (7) and related GLANN (23) architectures treat latent codes as learnable parameters directly, and train separate sampling procedures for synthesizing novel images.

While all of these approaches can generate new images, it is unclear if any of them can quickly generalize to novel object classes. Some results suggest the opposite: a VAE sufficiently powerful to model the training data becomes incapable of producing anything else (8).

2.2 Few-Shot Image Generation

Current attempts at few-shot image generation span a wide range of approaches and models. Neural Statistician, an early attempt, is similar to the early autoencoder in that it is built for few-shot classification, and largely discards the generative capability (14). Generation-oriented iterations of the same idea exist, but likewise depend on a large and varied set of labelled data for training (21). Other approaches based on few-shot classification algorithms include generative matching networks (4) and adversarial meta-learning (11). These models also depend on heavy supervision from the training data, and as a general rule, are fairly complicated, involving multiple networks and training procedures working in tandem - making them potentially difficult to reliably train in practice.

Separate work has converged on the problem of few-shot image generation from the side of generative modeling. (44; 38) and (45) investigate the ability of GANs to handle domain adaptation via fine-tuning - thus requiring substantial computation, and more novel class examples than are likely available in the true few-shot setting. DAGAN (1) and FUNIT (34) use adversarial training to produce entirely feed-forward few-shot image generators. However, both algorithms still depend on labelled training data from a variety of classes, and risk exhibiting the same problematic behaviors as standard GANs: mode collapse and sensitivity to hyperparameters (2; 3).

(24) introduces an algorithm for class-conditioning an unconditioned generative model. By matching batch statistics with real images in a pretrained latent space, the algorithm produces novel images from the same, possibly

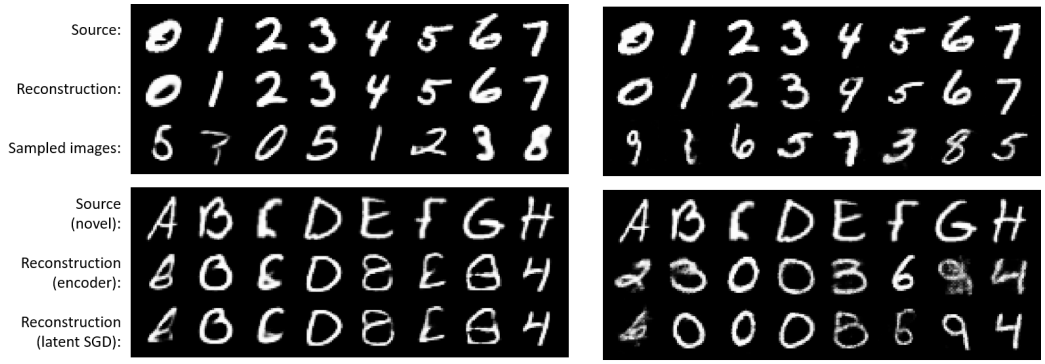

Figure 2: A VAE (left) and a WGAN-GP (right) are trained on MNIST (top row), and successfully recover the training images (second row) and synthesize new ones (third row). However, they struggle to represent images from novel classes (EMNIST letters, third and fourth rows). Even when latent codes are optimized directly, representing an oracle encoder, reconstruction quality is very poor. The generator itself is incapable of representing novel classes. Best viewed digitally.

novel class. However, the model is never evaluated on novel classes - it is unclear if it generalizes. It also requires heavy computation at test time.

In contrast to these prior works, interpolative autoencoders are simple and lightweight, train robustly, and generalize broadly from unlabelled data.

2.3 Interpolation

Ours is not the first attempt to improve latent space interpolation in autoencoders, or to recognize the need. (41) and (6) both use adversarial training to improve interpolation quality, while (35) uses interpolation under certain priors for data augmentation. None of these techniques, however, have been evaluated in the context of novel classes.

3 Problem Setup

Let X be a large collection of images depicting objects from a set of distinct classes C. We assume that image-level class labels exist, and are based on the semantic content of each image. We do not have access to these labels, however, or even know what or how many classes there are. Let X' be a very small collection of images - as few as two - all belonging to a single, novel class c' where $c' \notin C$. Our goal is to train an image generator on X which can rapidly adapt to X', such that subsequent generated images clearly belong to class c'.

This is an extremely difficult problem, as it encompasses two different challenges simultaneously. The first challenge is image synthesis: how to produce a novel image x such that $x \notin X$ while still belonging to some class $c \in C$. The second challenge is generalization: how to ensure that a model trained on X is still capable of producing images from some novel, unknown class $c' \notin C$.

We argue that the problem of generalization is much more difficult than synthesis. If X is highly constrained or domain-specific, then we would expect any network trained on X to model the data in an appropriately constrained and specific manner. This behavior is frequently observed in neural classifiers: it is common knowledge that fine-grained classifiers trained for a narrow, specific domain do not generalize well to different domains. Thus it is not clear that a model trained on X should generalize to novel class c' at all.

Therefore, we first examine how well existing generative models generalize. We train a VAE (29) and a WGAN-GP (19) on MNIST (33) handwritten digits (details in appendix). The networks converge nicely and achieve good synthesized image quality (see Fig. 2, third row). We then evaluate the ability of each generative model to recover particular images. We train a separate encoder network for the WGAN-GP which inverts the learned image/latent vector mapping of the generator. Using the built-in encoder of the VAE, and the new inversion network for the WGAN-GP, we find that both generative models can successfully autoencode and reconstruct a given set of training images (Fig. 2, top two rows). This is not surprising: the models are powerful and successfully learn the training data distribution. However, the same approach clearly fails on images coming from novel classes - in this case,

Figure 3: A random selection of autoencoder image reconstructions for unseen classes. From left to right: MNIST generalizes to EMNIST, Omniglot train generalizes to Omniglot test, and CIFAR10 generalizes to CIFAR100. Best viewed digitally.

handwritten letters from EMNIST (12). The outputs do not resemble the inputs; crucial semantic information is lost (Fig. 2, fourth and fifth rows). It could still be the case that the generator networks are capable of producing letters, but require better latent codes. We test this by simulating an oracle encoder network, by refining the latent codes via SGD using a pixel reconstruction loss. The resulting reconstructions are not much better (Fig. 2, bottom row), indicating that it is the generators, not the encoders, which fail to generalize to novel classes. This result confirms prior findings (8), and demonstrates the difficulty of few-shot image generation: despite their power and sophistication, current generative approaches simply *cannot* recover images from novel classes. Fig. 1 demonstrates a similar failure in a large, state-of-the-art GAN model.

Why do sophisticated generative models fail to generalize? We argue this is largely by design. Generative models such as VAEs and GANs are trained to minimize the divergence between a prior distribution and a learned posterior distribution. VAEs assign a prior over the space of latent embeddings $p(z)$, and push the posterior distribution toward it, where the posterior is a learned distribution over the training data $p(E(x))$, and E is the encoder network. GANs, on the other hand, minimize a divergence between prior and posterior distributions in image space, rather than latent space. The adversarial generator $G(z)$ parameterizes a learned distribution on images, which is trained to approach the true distribution of training data. The divergence between these distributions is parameterized and dynamically updated by the adversarial discriminator network. Regardless of the exact mechanism, however, training both models involves approximating a distribution via repeated sampling in latent space, and sending those samples through the generator network. Thus, by the time the generator is trained to convergence, and the posterior approaches the prior (or vice-versa), every region of the latent space feasible under the prior will have been mapped at some point to a training image - or, in the case of GANs, to an image indistinguishable from a training image. This means that a properly trained VAE or GAN cannot construct or reconstruct new object classes. If it could, then it would have been able to sample such images during training - which would mean that it had not been properly trained at all.

4 Interpolative AutoEncoders

Minimizing the divergence between prior and posterior training data distributions ensures good performance in image synthesis, but also ensures poor generalization. We conjecture that the opposite might also be true: plain autoencoders do not enforce any particular latent distribution on the data posterior, and thus it is unclear how to sample new images. However, this could also mean that they generalize well. More formally, given a network architecture E which maps an input image x to latent vector z, and a generator architecture G which maps the latent vector z back to image space, we refer to the function composition $G(E(\cdot))$ as the autoencoder. E and G are trained jointly over X such that the pixel reconstruction error between $x \in X$ and $G(E(x))$ is minimized. The question of generalization becomes, to what degree does a trained autoencoder maintain close proximity between x' and $G(E(x'))$ for x' which lies far from the data manifold of X?

We find that our conjecture holds: autoencoders generalize *extremely* well. Examples are given in Fig. 3, demonstrating near-perfect generalization performance over three pairs of class-disjoint datasets: MNIST digits to EMNIST letters, Omniglot training alphabets to Omniglot testing alphabets (32), and CIFAR10 to CIFAR100 (31). A quantitative evaluation can be found in Table 1. In addition to the three dataset pairings in Fig. 3, we evaluate generalization in the other direction, from EMNIST to MNIST, Omniglot test to Omniglot train, and CIFAR100 to CIFAR10. Finally, we test the ability of MNIST/EMNIST and Omniglot networks to generalize to each others' datasets in the face of domain shift (stroke width in Omniglot is almost universally thinner than for MNIST/EMNIST). Reconstruction quality is high across the board - extremely high, accounting for the fact that the MNIST and CIFAR networks were trained on only ten distinct classes! Autoencoders appear to exhibit hardly any overfitting whatsoever with respect to object classes, no matter how constrained those classes may be. Instead, it appears that even in cases where the domain of the training data is narrow, autoencoders learn an extremely general mapping of pixels to latent embeddings and back.

Table 1: Average autoencoder per-pixel L1 reconstruction error. Values are normalized to the color output range [0, 255]. Rows give the training dataset, columns give the evaluation dataset, and cells measure generalization from row to column. Generally, error is very low. The bottom row contains scores for a VAE, which generalizes poorly.

# training images:	MNIST *60,000*	EMNIST *124,800*	Omni (train) *19,280*	Omni (test) *13,180*
MNIST	2.29	9.89	6.03	6.88
EMNIST	3.79	2.73	4.48	5.30
Omniglot (train)	8.36	14.6	1.92	4.62
Omniglot (test)	9.41	15.2	4.59	2.16
MNIST (VAE)	6.25	23.0	15.9	17.9

# training images:	CIFAR-10 *50,000*	CIFAR-100 *50,000*
CIFAR-10	4.39	5.40
CIFAR-100	5.27	4.63

Figure 4: Autoencoders do not generalize trivially to any input. Color inversion on the training data represents a clear failure mode.

It is worth asking to what degree the mapping extracts semantically meaningful information in the first place. If the intrinsic dimensionality of the image manifold is low enough, then perhaps the autoencoder is not compressing the data at all, in which case it will generalize trivially to *any* image input. Fortunately, we see that this is not the case: when the input data is too far from the training distribution, autoencoders do fail. Fig. 4 displays one such failure mode for the MNIST and Omniglot networks: simple color inversion. The mapping learned by an autoencoder, while general, is also nontrivial.

4.1 Interpolative Training

The fact that autoencoders generalize indicates that they are capable of acting as few-shot image generators for novel classes. The generator network G can construct a wide range of images. The challenge is in determining a reliable way to *sample* appropriate latent z vectors, given only a small set of real example images. A simple way to do this is to interpolate between data points in latent space: every sampled point should be a weighted sum of two or more real points. This allows us to produce novel combinations of our given reference images without changing the overall semantic content.

Unfortunately, it is a known fact that autoencoders do not interpolate well (6). Semantic information is sometimes only weakly encoded in the learned mapping from x to z and back. Thus image content is not always preserved when moving between the latent space embeddings for two images of the same object. Indeed, we observe that our autoencoders frequently introduce visual artifacts on interpolated handwritten characters (see Fig. 6 and 7, middle row). Interpolation on natural images in CIFAR largely reduces to pixel-level fading (see Fig. 11, second row).

We introduce a novel training procedure for autoencoders to remove these visual artifacts and improve the preservation of semantic content during latent space interpolation. In addition to reconstructing images, we train the interpolative autoencoder to also recover known, semantically interpolated images from the corresponding interpolated latent code. In particular, suppose that we have a triplet of images $A = f(\theta_1)$, $B = f(\theta_2)$ and $C = f(\alpha\theta_1 + (1-\alpha)\theta_2)$, where f is some unknown image generating process, and θ is some semantic variable. Using this triplet, we train the interpolative autoencoder to reconstruct C by decoding the interpolated latent code of A and B. Formally, we minimize a loss L_{interp} such that

$$L_{interp} = ||C - G(\alpha E(A) + (1-\alpha)E(B))||_p \tag{1}$$

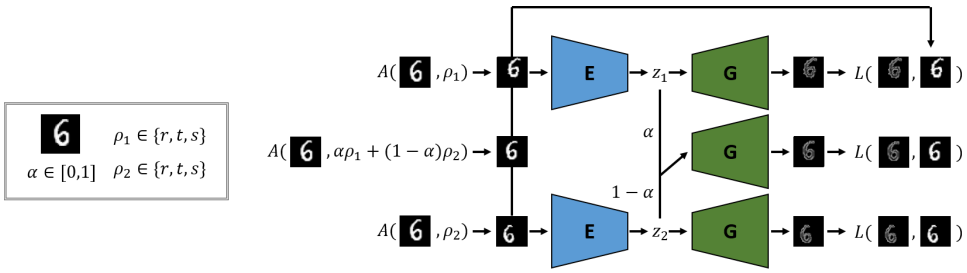

Figure 5: The training process for interpolative autoencoders. For every training image, two affine transformations ρ are sampled from predefined ranges of rotation, translation, and scale, as well as a mixing coefficient α. In addition to reconstructing the transformed images (top and bottom branches), the network is also trained to recover the mixed transformation image, by applying the same mixture in latent space. This forces the interpolation path in latent space to encode only spatially - and thus semantically - meaningful information.

for some choice of norm p, where E and G are the encoder and generator network functions and α represents the ground-truth interpolation mixing weight.

In practice, finding appropriate image triplets A, B, C is difficult, as images in the real world generally cannot be described as straightforward combinations of two other images. Instead, we generate A, B, C *synthetically* using affine spatial transformations. For a given training image x, we randomly sample two sets of rotation, translation, and scale parameters. Applying each of these transformations independently to x yields A and B. C is constructed by random uniform sampling of $\alpha \in [0, 1]$, which is used to compute a weighted average of our paired rotation, translation, and scale parameters. This weighted average parameterizes a third affine transformation, which applied to x yields C.

The interpolative autoencoder is trained to faithfully reconstruct A and B using an image reconstruction loss. At the same time, the network receives the mixing parameter α for each image triplet, and attempts to recover C from the α-weighted latent space interpolation. This corresponds to Equation 1, and the network is trained to minimize L_{interp} with a choice of norm corresponding to the reconstruction loss for A and B. The full procedure is displayed in Fig. 5.

In practice, we weight the reconstruction and interpolation loss terms so that they contribute equally to the overall loss. Since we measure the reconstruction loss on both A and B, the final interpolative autoencoder loss is

$$L = \frac{L_{reconstruct}}{2} + L_{interp} \tag{2}$$

5 Experiments

Through both qualitative and quantitative results, we demonstrate the effectiveness of interpolative autoencoders on the unsupervised few-shot image generation task. All encoder and decoder networks are simple 4- to 6-layer architectures. We employ such shallow networks to illustrate that the interpolative autoencoder approach, and not network power, is responsible for good performance. Training and evaluation details for all experiments can be found in the appendix.

5.1 Handwritten Characters

Image quality: We evaluate the performance of interpolative autoencoders on two handwritten character dataset pairs. We train one set of models on MNIST digits and evaluate on EMNIST letters, while the other generalizes from the Omniglot training alphabets to the testing alphabets. The autoencoder architecture in both cases consists of a 4-layer encoder and generator based on InfoGAN (10). Input images have a 28×28 resolution, and the latent dimensionality of the autoencoder is 64, representing over an order of magnitude in dimensionality reduction.

Figures 6 and 7 display representative interpolation results for these two contexts. It is clear that interpolative training produces better quality interpolations and removes visual artifacts present in the vanilla autoencoder images. These qualitative results are verified quantitatively in Table 2. Interpolative autoencoders, as measured by FID score (20), outperform both their vanilla counterparts and VAEs trained directly on the evaluation data, de-

Figure 6: A random selection of interpolations between image pairs from novel classes (EMNIST). Pixel interpolation is included as a simple baseline. Red squares indicate places where vanilla autoencoders break or overextend semantically important strokes. Interpolative autoencoders successfully remove these artifacts. Best viewed digitally.

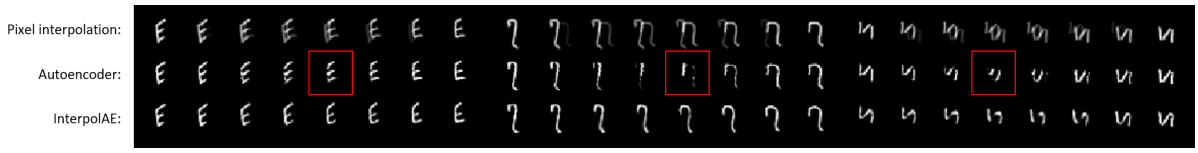

Figure 7: A random selection of interpolations between image pairs from novel classes (Omniglot). Pixel interpolation is included as a simple baseline. Red squares indicate places where vanilla autoencoders remove semantically important strokes. Interpolative autoencoders successfully address, or at least mitigate, these artifacts. Best viewed digitally.

spite significant domain shift. This indicates that in terms of image quality, interpolative autoencoders are effective few-shot image generators.

Generalizability: We compare our approach to two prior few-shot generation methods: Neural Statistician (14) and Data Augmentation GANs (DAGAN) (1). Both of these approaches require class labels on the training data, while ours does not. We train both models on MNIST and then attempt to synthesize new images from EMNIST. Fig. 8 makes clear that interpolative autoencoders generalize more broadly than prior approaches and are much less restricted by a narrow training data domain. Both neural statistician and DAGAN overfit to the training classes, and generate additional number images instead of the desired letters.

Data hallucination: To demonstrate the practical utility of interpolative autoencoders, we examine their performance as a form of data augmentation. A series of classifiers are trained on the "ByMerge" split of EMNIST, with the digit classes removed (i.e. letter classes only, lower- and upper-case separated). During training, half of each batch is dedicated to augmented data, with a mixing coefficient of 0.5 in all cases.

We compare four classifiers, and find that accuracy directly improves with the sophistication of the data augmentation technique (see Table 3). Pixel-level interpolation, otherwise known as MixUp (47), provides a small boost to accuracy. Interpolating in the latent space of an autoencoder trained on MNIST yields a larger boost, and an interpolative autoencoder trained on MNIST improves further still.

5.2 CELEB-A

We extend our results to the more difficult domain of higher-resolution natural images. We use the Celeb-A dataset of faces, scaled to a resolution of 128×128. Our models are trained strictly on male faces, and attempt to generalize to female faces. Encoder and decoder networks are six layers deep and fully convolutional, with a latent dimensionality of 512. Since the input images have 3 color channels, the resulting dimensionality reduction is almost two orders of magnitude.

Table 2: FID scores across models and datasets. The VAE model is trained directly on the evaluation data as a strong baseline. Interpolative autoencoders are superior in all contexts.

Model	EMNIST	Omniglot
AE	17.66	106.33
VAE	18.73	87.44
InterpolAE	17.32	74.70

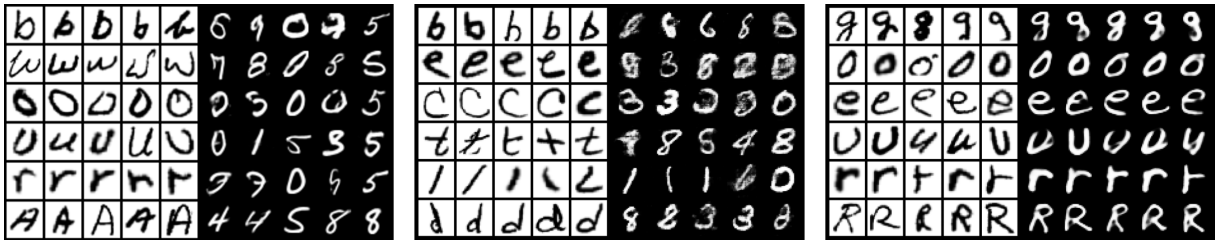

Figure 8: Comparison to prior work. Seed images are in color negative on the left, synthesized images on the right. Neural Statistician (left) and DAGAN (middle) overfit to the training data and fail to maintain class-consistency in the synthesized images. Interpolative autoencoders (right) successfully generate new letters.

Table 3: Autoencoder interpolation as a form of data augmentation on EMNIST. Results are averaged over ten runs, with 95% confidence intervals. Interpolative autoencoders provide the most effective form of data augmentation.

Augmentation	**Accuracy**
None	$90.33 \pm .06$
Mixup	$90.66 \pm .04$
AE	$91.27 \pm .04$
InterpolAE	$\mathbf{91.50 \pm .05}$

The increased output resolution, combined with the relative simplicity of our networks, makes this a much more difficult task than for handwritten characters. In practice, this difficulty manifests as blurriness in the outputs. Blur in generative models is a well-studied phenomenon and techniques exist to eliminate it. We augment our training by employing a perceptual loss (25), aligning not just pixel values to the ground truth but also activations inside a frozen feature extractor. In order to maintain the unsupervised nature of our algorithm, however, we cannot use a pretrained classifier as the feature extractor, as is common practice. Instead, we formulate an "autoperceptual loss" where the encoder of the current model is frozen and used to calculate the perceptual loss. We found that the autoperceptual loss, weighted equally with the reconstruction loss, improves convergence and removes some, though not all, of the blur in the outputs. This partial effectiveness is unsurprising, as our networks are only six layers deep. We predict that a more powerful, more sophisticated network would achieve better performance, but we focus on simple models here since we are primarily concerned with generalization, not image quality *per se*.

The results, displayed in Fig. 9, are similar to our findings for handwritten characters. Autoencoders generalize well, but produce visual artifacts during interpolation. In this case, the artifacts take the form of unrealistic, transparent regions around the head. Compared to the vanilla autoencoder, the interpolative autoencoder with autoperceptual loss removes these artifacts and restores semantic meaning to the interpolation path, though it sacrifices some image detail and quality to do so.

Diversity: a practical concern with using interpolation as a latent sampling mechanism is the accompanying restriction on image diversity. In the extreme case, where only two novel seed images are available, the model is forced to move back and forth between the two images and nowhere else, heavily restricting the semantic range of available outputs. We argue that this restriction seems worse than it is: the number of unique image pairs grows quadratically with the number of seed images, and Fig. 10 demonstrates the rich diversity of sampled images which can result from only six available seeds.

We conclude that interpolative autoencoders are effective not just on highly constrained handwritten characters, but also on higher-resolution color images.

5.3 CIFAR

Finally, we evaluate our model in an extremely challenging setting: natural images with no domain restriction. Unconstrained natural images represent a significant challenge for few-shot image generation, as intra-class variation is so much higher (images have very little consistent global structure). This also makes the natural image domain particularly difficult for interpolation-based models, as it is not always possible to interpolate smoothly and intuitively between two given images. If one image is a flower in a vase, and another is a flower in a field, how do you semantically interpolate between them?

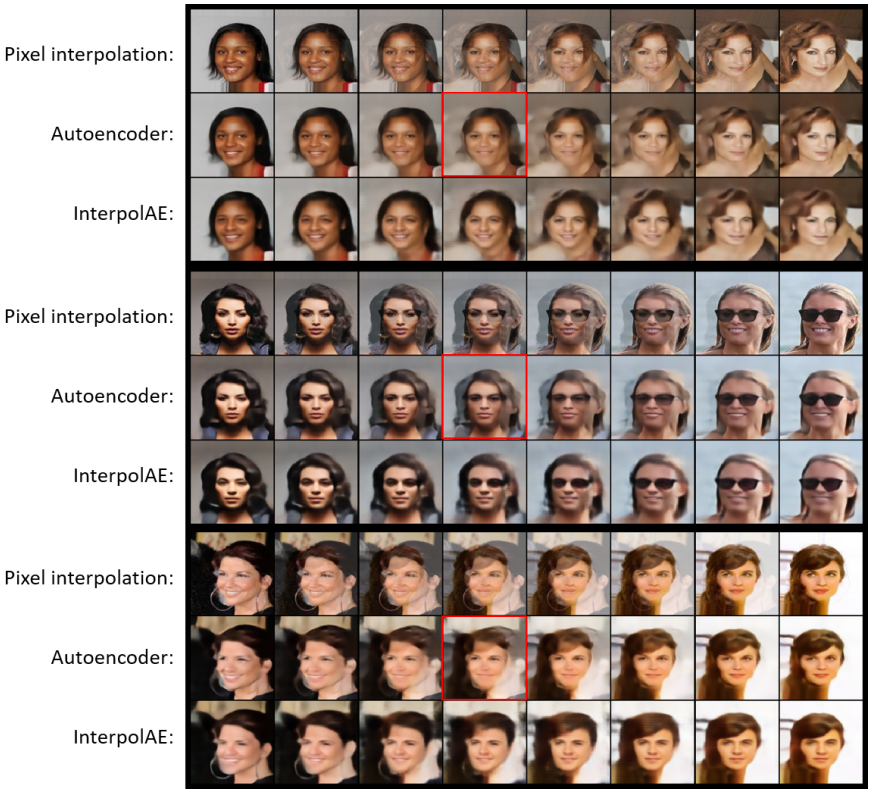

Figure 9: Examples of interpolations between image pairs from a novel class (Celeb-A). Pixel interpolation is included as a simple baseline. Red squares indicate places where vanilla autoencoders introduce transparency artifacts. Interpolative autoencoders successfully mitigate these artifacts. These examples are hand-picked, as transparency artifacts for the vanilla autoencoder do not appear as frequently as in the handwritten character datasets. Best viewed digitally.

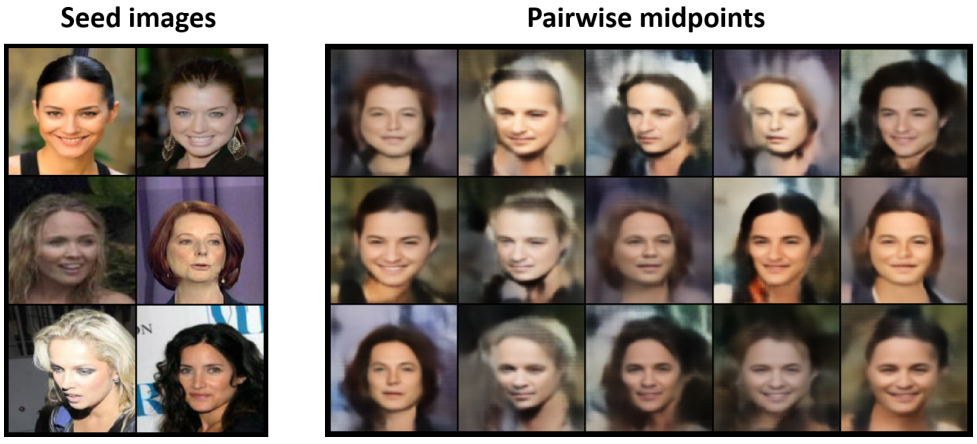

Figure 10: Six seed images (left) are sufficient to produce a diversity of novel images (right). The fifteen displayed here represent the midpoints of the fifteen unique pairwise interpolation paths. We re-emphasize that this network was trained only on male faces. Best viewed digitally.

We train our models on CIFAR10 and evaluate on novel CIFAR100 classes. Our network architectures are 5 layers deep and fully convolutional, with a latent dimensionality of 512. With a 32×32 image resolution and 3

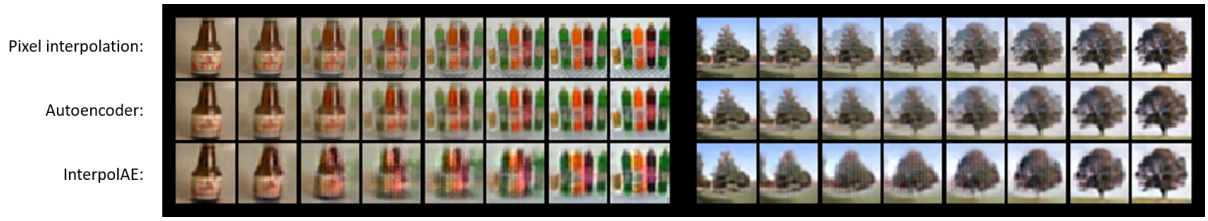

Figure 11: Examples of interpolation between image pairs from novel classes (CIFAR100). Vanilla autoencoders are indistinguishable from pixel interpolation. Interpolative autoencoders produce more semantically meaningful transformations. **Left**: the interpolative autoencoder gradually spreads out an initially tight cluster of bottles. **Right**: the contour of the foliage morphs smoothly. These examples are hand-picked, as natural image interpolation is not always meaningful or interesting. Best viewed digitally.

color channels this amounts to a six times reduction in dimensionality. We again use an autoperceptual loss to aid convergence in the interpolative model and remove blur.

Fig. 11 displays example outputs from vanilla and interpolative autoencoders on novel CIFAR100 data. Again, the vanilla autoencoder produces transparency artifacts - in this case, the output is nearly indistinguishable from the pixel fade. Interpolative training is able to remove most of the transparency and restore semantic content along the interpolation path. We conclude that the combined techniques of interpolative autoencoders and autoperceptual loss are moderately effective at unsupervised few-shot image generation for natural images. It is likely that a more powerful neural architecture could achieve better image quality - our lightweight autoencoder networks are sufficient for generalization, but cannot maintain high interpolation quality at the same time. However, we also point out that few-shot generation on natural images is inherently a very difficult problem, and many challenges remain.

6 Conclusion

We introduce a powerful, lightweight, and class-label-free method for few-shot image generation. Using the fact that autoencoders generalize very broadly from limited data, we incorporate a novel training procedure to produce interpolative autoencoders, which synthesize realistic images of novel classes from as few as two examples. Interpolative autoencoders are robust and generalize far more broadly than prior work, with outputs that are both good-quality and practically useful for downstream tasks. We are confident that with more powerful networks and sampling mechanisms, interpolative autoencoders can be further improved and serve as a valuable base model for unsupervised few-shot image generation.

REFERENCES

[1] Antoniou, A., Storkey, A., Edwards, H.: Data augmentation generative adversarial networks. arXiv preprint arXiv:1711.04340 (2017)

[2] Arora, S., Ge, R., Liang, Y., Ma, T., Zhang, Y.: Generalization and equilibrium in generative adversarial nets (gans). In: Proceedings of the 34th International Conference on Machine Learning-Volume 70. pp. 224–232. JMLR. org (2017)

[3] Arora, S., Risteski, A., Zhang, Y.: Do gans learn the distribution? some theory and empirics (2018)

[4] Bartunov, S., Vetrov, D.: Few-shot generative modelling with generative matching networks. In: International Conference on Artificial Intelligence and Statistics. pp. 670–678 (2018)

[5] Bau, D., Zhu, J.Y., Wulff, J., Peebles, W., Strobelt, H., Zhou, B., Torralba, A.: Seeing what a gan cannot generate. In: Proceedings of the International Conference Computer Vision (ICCV) (2019)

[6] Berthelot, D., Raffel, C., Roy, A., Goodfellow, I.: Understanding and improving interpolation in autoencoders via an adversarial regularizer. arXiv preprint arXiv:1807.07543 (2018)

[7] Bojanowski, P., Joulin, A., Lopez-Paz, D., Szlam, A.: Optimizing the latent space of generative networks. arXiv preprint arXiv:1707.05776 (2017)

[8] Bozkurt, A., Esmaeili, B., Brooks, D., Dy, J., Meent, J.W.: Can vaes generate novel examples? arXiv preprint arXiv:1812.09624 (2018)

[9] Brock, A., Donahue, J., Simonyan, K.: Large scale gan training for high fidelity natural image synthesis. arXiv preprint arXiv:1809.11096 (2018)

[10] Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., Abbeel, P.: Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In: Advances in neural information processing systems. pp. 2172–2180 (2016)

[11] Clouâtre, L., Demers, M.: Figr: Few-shot image generation with reptile. arXiv preprint arXiv:1901.02199 (2019)

[12] Cohen, G., Afshar, S., Tapson, J., van Schaik, A.: Emnist: an extension of mnist to handwritten letters. arXiv preprint arXiv:1702.05373 (2017)

[13] Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Sengupta, B., Bharath, A.: Generative adversarial networks: An overview. IEEE Signal Processing Magazine **35** (10 2017)

[14] Edwards, H., Storkey, A.: Towards a neural statistician. arXiv preprint arXiv:1606.02185 (2016)

[15] Finn, C., Abbeel, P., Levine, S.: Model-agnostic meta-learning for fast adaptation of deep networks. In: Proceedings of the 34th International Conference on Machine Learning-Volume 70. pp. 1126–1135. JMLR. org (2017)

[16] Ghosh, A., Kulharia, V., Namboodiri, V.P., Torr, P.H., Dokania, P.K.: Multi-agent diverse generative adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 8513–8521 (2018)

[17] Gidaris, S., Komodakis, N.: Dynamic few-shot visual learning without forgetting. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4367–4375 (2018)

[18] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in neural information processing systems. pp. 2672–2680 (2014)

[19] Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.: Improved training of wasserstein gans. In: Advances in neural information processing systems. pp. 5769–5779 (2017)

[20] Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. In: Advances in Neural Information Processing Systems. pp. 6626–6637 (2017)

[21] Hewitt, L.B., Nye, M.I., Gane, A., Jaakkola, T., Tenenbaum, J.B.: The variational homoencoder: Learning to learn high capacity generative models from few examples. arXiv preprint arXiv:1807.08919 (2018)

[22] Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. science **313**(5786), 504–507 (2006)

[23] Hoshen, Y., Li, K., Malik, J.: Non-adversarial image synthesis with generative latent nearest neighbors. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5811–5819 (2019)

[24] Jitkrittum, W., Sangkloy, P., Gondal, M.W., Raj, A., Hays, J., Schölkopf, B.: Kernel mean matching for content addressability of GANs. In: Proceedings of the 36th International Conference on Machine Learning. pp. 3140–3151 (2019)

[25] Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: European conference on computer vision. pp. 694–711. Springer (2016)

[26] Karras, T., Aila, T., Laine, S., Lehtinen, J.: Progressive growing of gans for improved quality, stability, and variation. arXiv preprint arXiv:1710.10196 (2017)

[27] Karras, T., Aila, T., Laine, S., Lehtinen, J.: Progressive growing of gans for improved quality, stability, and variation. In: Proceedings of the sixth International Conference on Learning Representations (2018)

[28] Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4401–4410 (2019)

[29] Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013)

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

[31] Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images. Tech. rep., Citeseer (2009)

[32] Lake, B.M., Salakhutdinov, R., Tenenbaum, J.B.: Human-level concept learning through probabilistic program induction. Science **350**(6266), 1332–1338 (2015)

[33] LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., et al.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (1998)

[34] Liu, M.Y., Huang, X., Mallya, A., Karras, T., Aila, T., Lehtinen, J., Kautz, J.: Few-shot unsupervised image-to-image translation. arXiv preprint arXiv:1905.01723 (2019)

[35] Liu, X., Zou, Y., Kong, L., Diao, Z., Yan, J., Wang, J., Li, S., Jia, P., You, J.: Data augmentation via latent space interpolation for image classification. In: 2018 24th International Conference on Pattern Recognition (ICPR). pp. 728–733. IEEE (2018)

[36] Mao, Q., Lee, H.Y., Tseng, H.Y., Ma, S., Yang, M.H.: Mode seeking generative adversarial networks for diverse image synthesis. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1429–1437 (2019)

[37] Masci, J., Meier, U., Cireşan, D., Schmidhuber, J.: Stacked convolutional auto-encoders for hierarchical feature extraction. In: International Conference on Artificial Neural Networks. pp. 52–59. Springer (2011)

[38] Noguchi, A., Harada, T.: Image generation from small datasets via batch statistics adaptation. arXiv preprint arXiv:1904.01774 (2019)

[39] Ravi, S., Larochelle, H.: Optimization as a model for few-shot learning (2016)

[40] Razavi, A., Oord, A.v.d., Vinyals, O.: Generating diverse high-fidelity images with vq-vae-2. arXiv preprint arXiv:1906.00446 (2019)

[41] Sainburg, T., Thielk, M., Theilman, B., Migliori, B., Gentner, T.: Generative adversarial interpolative autoencoding: adversarial training on latent space interpolations encourage convex latent distributions. arXiv preprint arXiv:1807.06650 (2018)

[42] Snell, J., Swersky, K., Zemel, R.: Prototypical networks for few-shot learning. In: Advances in Neural Information Processing Systems. pp. 4077–4087 (2017)

[43] Vinyals, O., Blundell, C., Lillicrap, T., Wierstra, D., et al.: Matching networks for one shot learning. In: Advances in neural information processing systems. pp. 3630–3638 (2016)

[44] Wang, Y., Wu, C., Herranz, L., van de Weijer, J., Gonzalez-Garcia, A., Raducanu, B.: Transferring gans: generating images from limited data. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 218–234 (2018)

[45] Wu, C., Herranz, L., Liu, X., van de Weijer, J., Raducanu, B., et al.: Memory replay gans: Learning to generate new categories without forgetting. In: Advances In Neural Information Processing Systems. pp. 5962–5972 (2018)

[46] Zhang, H., Goodfellow, I., Metaxas, D., Odena, A.: Self-attention generative adversarial networks. arXiv preprint arXiv:1805.08318 (2018)

[47] Zhang, H., Cisse, M., Dauphin, Y.N., Lopez-Paz, D.: mixup: Beyond empirical risk minimization. arXiv preprint arXiv:1710.09412 (2017)