Learning to Generate Images With Perceptual Similarity Metrics

Jake Snell
University of Toronto

Karl Ridgeway
University of Colorado, Boulder

Renjie Liao
University of Toronto

Brett D. Roads
University of Colorado, Boulder

Michael C. Mozer
University of Colorado, Boulder

Richard S. Zemel
University of Toronto

Abstract

Deep networks are increasingly being applied to problems involving image synthesis, e.g., generating images from textual descriptions and reconstructing an input image from a compact representation. Supervised training of image-synthesis networks typically uses a pixel-wise loss (PL) to indicate the mismatch between a generated image and its corresponding target image. We propose instead to use a loss function that is better calibrated to human perceptual judgments of image quality: the multiscale structural-similarity score (MS-SSIM) [31]. Because MS-SSIM is differentiable, it is easily incorporated into gradient-descent learning. We compare the consequences of using MS-SSIM versus PL loss on training deterministic and stochastic autoencoders. For three different architectures, we collected human judgments of the quality of image reconstructions. Observers reliably prefer images synthesized by MS-SSIM-optimized models over those synthesized by PL-optimized models, for two distinct PL measures (L_1 and L_2 distances). We also explore the effect of training objective on image encoding and analyze conditions under which perceptually-optimized representations yield better performance on image classification. Finally, we demonstrate the superiority of perceptually-optimized networks for super-resolution imaging. Just as computer vision has advanced through the use of convolutional architectures that mimic the structure of the mammalian visual system, we argue that significant additional advances can be made in modeling images through the use of training objectives that are well aligned to characteristics of human perception.

1. Introduction

There has been a recent explosion of interest in developing methods for image representation learning, focused in particular on training neural networks to synthesize images. The reason for this surge is threefold. First, the problem of image generation spans a wide range of difficulty, from synthetic images to handwritten digits to naturally cluttered and high-dimensional scenes, the latter of which provides a fertile development and testing ground for generative models. Second, learning good generative models of images involves learning new representations. Such representations are believed to be useful for a variety of machine learning tasks, such as classification or clustering, and can also support transfer between tasks. They are also applicable to other vision problems, including analysis by synthesis, learning of 3D representations, and future prediction in video. Third, image generation is fun and captures popular imagination, as efforts such as Google's Inceptionism machine demonstrate.

Figure 1. Three examples showing reconstructions of an original image (center) by a standard reconstruction approach (left) and our technique (right). The compression factor is high to highlight the differences.

While unsupervised image representation learning has become a popular task, there is surprisingly little work on studying loss functions that are appropriate for image generation. A basic method for learning generative image models is the *autoencoder* architecture. Autoencoders are made up of two functions, an encoder and a decoder. The encoder compresses an image into a feature vector, typically of low dimension, and the decoder takes that vector as input and reconstructs the original image as output. The standard loss function is the squared Euclidean (L_2) distance between the original and reconstructed images, also referred to as the *mean squared error* or *MSE*. A city-block (L_1) distance is sometimes used as well, referred to as the *mean absolute error* or *MAE*. As we will show, both loss functions yield blurry results–synthesized images that appear to have been low-pass filtered. Probabilistic formulations of autoencoders have also been proposed that maximize the likelihood of the observed images being generated, but estimating likelihoods in high-dimensional image space is notoriously difficult [26].

In this paper, we explore loss functions that, unlike MSE, MAE, and likelihoods, are grounded in human perceptual judgments. We show that these perceptual losses lead to representations are superior to other methods, both with respect to reconstructing given images (Figure 1), and generating novel ones. This superiority is demonstrated both in quantitative studies and human judgements. Beyond achieving perceptually superior synthesized images, we also show that that our perceptually-optimized representations are better suited for image classification. Finally, we demonstrate that perceptual losses yield a convincing win when applied to a state-of-the-art architecture for single image super-resolution.

2. Background and Related Work

2.1. Neural networks for image synthesis

The standard neural network for image synthesis is the autoencoder, of which there are two primary types. In the classic *deterministic* autoencoder, the input is mapped directly through hidden layers to output a reconstruction of the original image. The autoencoder is trained to reproduce an image that is similar to the input, where similarity is evaluated using a pixel-wise loss between the image and its reconstruction. In a *probabilistic* autoencoder, the encoder is used to approximate a posterior distribution and the decoder is used to stochastically reconstruct the data from latent variables; the model output is viewed as a distribution over images, and the model is trained to maximize the likelihood of the original image under this distribution. The chief advantage of probabilistic autoencoders is that they permit stochastic generation of novel images. The key issue with probabilistic autoencoders concerns the intractability of inference in the latent variables, e.g., Helmholtz Machines [6]. Variational autoencoders (VAEs) [16] utilize simple variational distributions to address this issue.

A second approach to building generative models for image synthesis uses variants of Boltzmann Machines [25, 14] and Deep Belief Networks [13]. While these models are very powerful, each iteration of training requires a computationally costly step of MCMC to approximate derivatives of an intractable partition function (normalization constant), making it difficult to scale them to large datasets.

A third approach to learning generative image models, which we refer to as the *direct-generation* approach, involves training a generator that maps random samples drawn from a uniform distribution through a deep neural network that outputs images, and attempts through training to make the set of images generated by the model indistinguishable from real images. Generative Adversarial Networks (GANs) [11] is a paradigm that involves training a discriminator that attempts to distinguish real from generated images, along with a generator that attempts to trick the discriminator. Recently, this approach has been scaled by training conditional GANs at each level of a Laplacian pyramid of images [8]. With some additional clever training ideas, these adversarial networks have produced very impressive generative results, e.g., [23]. Drawbacks of the GAN include the need to train a second network, a deep and complicated adversary, and the fact that the training of the two networks are inter-dependent and lack a single common objective. An alternative approach, moment-matching networks [19], directly trains the generator to make the statistics of these two distributions match.

Because the goal of image generation is to synthesize images that humans would judge as high quality and natural, current approaches seem inadequate by failing to incorporate measures of human perception. With direct-generation approaches, human judgments could in principle be incorporated by replacing the GAN with human discrimination of real from generated images. However, in practice, the required amount of human effort would make such a scheme impractical. In this paper, we describe an alternative approach using the autoencoder architecture; this approach incorporates image assessments consistent with human perceptual judgments without requiring human data collection.

We focus on autoencoders over direct-generation approaches for a second reason: autoencoders *interpret* images in addition to generating images. That is, an input image can be mapped to a compact representation that encodes the underlying properties of the world responsible for the observed image features. This joint training of the encoder and decoder facilitates task transfer: the encoder can be used as the initial image mapping that can be utilized for many different applications. Although adversarial training can be combined with autoencoding, here we explore autoencoding in isolation, to study the effects of optimizing with perceptually-based metrics.

Because autoencoders reconstruct training images, training the network requires evaluating the quality of the reconstruction with respect to the original. This evaluation is based on a pixel-to-pixel comparison of the images—a so-called *full-reference metric*. Deterministic autoencoders typically use *mean-squared error* (*MSE*), the average square of the pixel intensity differences, or *mean-absolute* error (*MAE*), the average of the absolute difference in pixel intensity. Probabilistic autoencoders typically use a likelihood measure that is a monotonically decreasing function of pixelwise differences. In many instances, these three standard measures—MSE, MAE, and likelihood—fail to capture human judgments of quality. For example, a distorted image created by decreasing the contrast can yield the same standard measure as one created by increasing the contrast, but the two distortions can yield quite different human judgments of visual quality; and distorting an image with salt-and-pepper impulse noise obtains a small perturbation by standard measures but is judged by people as having low visual quality relative to the original image.

2.2. Perception-Based Error Metrics

As digitization of photos and videos became commonplace in the 1990s, the need for digital compression also became apparent. Lossy compression schemes distorted image data, and it was important to quantify the drop in quality resulting from compression in order to optimize the compression scheme. Because compressed digital artifacts are eventually used by humans, researchers attempted to develop full-reference image quality metrics that take into account features to which the human visual system is sensitive and that ignore features to which it is insensitive. Some are built on complex models of the human visual system, such as the Sarnoff JND model [20], the visual differences predictor [5], the moving picture quality metric [28], the perceptual distortion metric [32], and that of [10].

Other metrics take more of an engineering approach, and are based on the extraction and analysis of specific features of an image to which human perception is sensitive. The most popular of these metrics is the structural similarity metric (SSIM) [29], which aims to match the luminance, contrast, and structure information in an image. Other such metrics are the visual information fidelity metric [24], which is an information theory-based measure, and the visual signal-to-noise ratio [3].

Finally, there are transform-based methods, which compare the images after some transformation has been applied. Some of these methods include DCT/wavelets, discrete orthonormal transforms, and singular value decomposition.

2.3. Structural Similarity

In this paper, we train neural nets with the structural-similarity metric (SSIM) [29] and its multiscale extension (MS-SSIM) [31]. We chose the SSIM family of metrics because it is well accepted and frequently utilized in the literature. Further, its pixelwise gradient has a simple analytical form and is inexpensive to compute. In this work, we focus on the original grayscale SSIM and MS-SSIM, although there are interesting variations and improvements such as colorized SSIM [17, 12].

The single-scale SSIM metric [29] compares corresponding pixels and their neighborhoods in two images, denoted x and y, with three comparison functions—luminance (I), contrast (C), and structure (S):

$$I(x,y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \quad C(x,y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2}$$

$$S(x,y) = \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3}$$

The variables μ_x, μ_y, σ_x, and σ_y denote mean pixel intensity and the standard deviations of pixel intensity in a local image patch centered at either x or y. Following [29], we chose a square neighborhood of 5 pixels on either side of x or y, resulting in 11×11 patches. The variable σ_{xy} denotes the sample correlation coefficient between corresponding pixels in the patches centered at x and y. The constants C_1, C_2, and C_3 are small values added for numerical stability. The three comparison functions are combined to form the SSIM score:

$$\text{SSIM}(x,y) = I(x,y)^\alpha C(x,y)^\beta S(x,y)^\gamma$$

This single-scale measure assumes a fixed image sampling density and viewing distance, and may only be appropriate for certain range of image scales. This issue is addressed in [31] with a variant of SSIM that operates at multiple scales simultaneously. The input images x and y are iteratively downsampled by a factor of 2 with a low-pass filter, with scale j denoting the original images downsampled by a factor of 2^{j-1}. The contrast $C(x,y)$ and structure $S(x,y)$ components are applied at all scales. The luminance component is applied only at the coarsest scale, denoted M. Additionally, a weighting is allowed for the contrast and structure components at each scale, leading to the definition:

$$\text{MS-SSIM}(x,y) = I_M(x,y)^{\alpha_M} \prod_{j=1}^{M} C_j(x,y)^{\beta_j} S_j(X,y)^{\gamma_j}$$

In our work, we weight each component and each scale equally ($\alpha = \beta_{1..M} = \gamma_{1..M} = 1$), a common simplification of MS-SSIM. Following [31], we use $M = 5$ downsampling steps.

Our objective is to minimize the loss related to the sum of structural-similarity scores across all image pixels,

$$\mathcal{L}(X,Y) = -\sum_i \text{MS-SSIM}(X_i, Y_i),$$

where X and Y are the original and reconstructed images, and i is an index over image pixels. This equation has a simple analytical derivative [30] and therefore it is trivial to perform gradient descent in the MS-SSIM-related loss.

We now turn to two sets of simulation experiments that compare autoencoders trained with a pixelwise loss (MSE and MAE) to those trained with a perceptually optimized loss (SSIM or MS-SSIM). The first set of experiments is based on deterministic autoencoders, and the second is based on a probabilistic autoencoder, the VAE [16].

3. Deterministic Autoencoders

We demonstrate the benefits of training deterministic autoencoders to optimize SSIM or MS-SSIM across images of various sizes, for various bottlenecks in the autoencoder, and for various network architectures. We begin with a study using small, highly compressed images and a fully connected architecture. We then present results on larger images with a convolutional autoencoder architecture.

3.1. Architectures and Data Sets

In the first simulation, we trained networks on small (32×32) images using a fully-connected architecture with a six-layer encoder mapping the 1024-dimensional input to a bottleneck layer of 256 units. The decoder component of the architecture mirrors the encoder. We trained two networks that are identical except for their loss function—one to optimize MSE, and one to optimize SSIM. Because the images are so small, the single-scale SSIM is appropriate; downsampling the images any further blurs the content to the point where humans have trouble distinguishing objects in the image. We train the fully-connected autoencoders using a subset of approximately two million images of the 80 million Tiny-Images data set [27], consisting of the first 30 images for every English proper noun. The 32×32 images in this dataset consist of RGB color channels. We mapped the three color channels to a single grayscale channel using the ITU-R 601-2 luma transform. All testing and evaluation of our models used the CIFAR-10 data set, which consists of 60,000 color images, each drawn from one of ten categories. We chose a diverse data set for training in order ensure that the autoencoders were learning general statistical characteristics of images, and not peculiarities of the CIFAR-10 data set. The CIFAR-10 color images were converted to a single grayscale channel, as was done for the training data set. Additional details regarding the architecture and training procedure can be found in the supplementary materials.

Next we trained networks on larger images (96×96 pixels) with a *convolutional* autoencoder architecture [22]: convolutional layers encode the input and deconvolutional layers decode the feature representation in the bottleneck layer. The convolutional network architecture consists of 3 convolutional layers, each with a filter size of 5 and a stride of 2. The deconvolutional layers again mirror the convolutional layers. For these larger images, which may have structure at multiple spatial scales, we used the MS-SSIM rather than SSIM as our perceptual similarity metric. We compared MS-SSIM to two pixelwise measures: MSE and MAE. Because MSE focuses on outliers and we have no reason to believe that the human eye has a similar focus, we felt it important to include MAE. If we observe MS-SSIM outperforming both MSE and MAE, we will have stronger evidence for the conclusion that perceptually-optimized measures outperform pixelwise losses in general. For training and testing, we use the STL-10 dataset [4], which consists of larger RGB color images, of the same classes as CIFAR-10. The images were converted to grayscale using the method we used for CIFAR-10. For our experiments, we train our models on the 100,000 images in STL-10 referred to as the "unlabeled" set, and of the remaining data, we formed a *validation* set of 10,400 images and a *test* set of 2,800 images.

3.2. Judgments of Reconstruction Quality

Do human observers prefer reconstructions produced by perceptually-optimized networks or by the pixelwise-loss optimized networks? We collected judgments of perceptual quality on Amazon Mechanical Turk.

3.2.1 Fully-Connected Autoencoders

Participants were shown image triplets with the original (reference) image in the center and the SSIM- and MSE-optimized reconstructions on either side with the locations counterbalanced. Participants were instructed to select which of the two reconstructions they preferred.

In a first study, twenty participants provided preference judgments on the same set of 100 randomly selected images from the CIFAR-10 data set. For each image triple, we recorded the proportion of participants who choose the SSIM reconstruction of the image over the MSE reconstruction. Figure 2a shows the distribution of inter-participant preference for SSIM reconstructions across all 100 images. If participants were choosing randomly, we would expect to see roughly 50% preference for most images. However, a plurality of images have over 90% inter-participant agreement on SSIM, and almost no images have MSE reconstructions that are preferred over SSIM reconstructions by a majority of participants.

Figure 3a shows the eight image triplets for which the largest proportion of participants preferred the SSIM reconstruction. The original image is shown in the center of the triplet and the MSE- and SSIM-optimized reconstructions appear on the left and right, respectively. (In the actual experiment, the two reconstructions were flipped on half of the trials.) The SSIM reconstructions all show important object details that are lost in the MSE reconstructions and were unanimously preferred by participants. Figure 3b shows the eight image triples for which the smallest proportion of participants preferred the SSIM reconstruction.

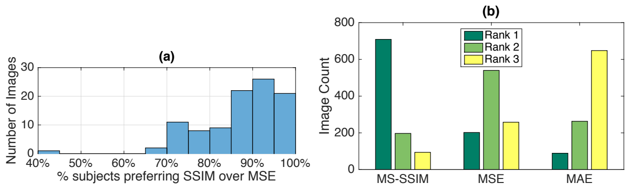


Figure 2. Human judgments of reconstructed images. (a) Fully connected network: Proportion of participants preferring SSIM to MSE for each of 100 image triplets. (b) Deterministic conv. network: Distribution of image quality ranking for MS-SSIM, MSE, and MAE for 1000 images from the STL-10 hold-out set.

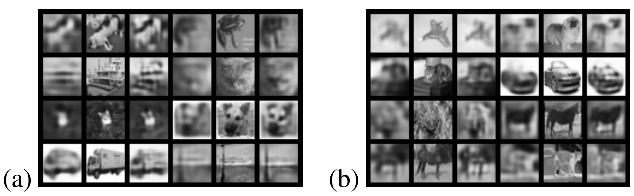


Figure 3. Image triples consisting of—from left to right—the MSE reconstruction, the original image, and the SSIM reconstruction. Image triples are ordered, from top to bottom and left to right, by the percentage of participants preferring SSIM. (a) Eight images for which participants strongly preferred SSIM over MSE. (b) Eight images for which the smallest proportion of participants preferred SSIM.

In the first seven of these images, still a majority (70%) of participants preferred the SSIM reconstruction to the MSE reconstruction; only in the image in the lower right corner did a majority prefer the MSE reconstruction (60%). The SSIM-optimized reconstructions still seem to show as much detail as the MSE-optimized reconstructions, and the inconsistency in the ratings may indicate that the two reconstructions are of about equal quality.

In a second study on Mechanical Turk, twenty new participants each provided preference judgments on a randomly drawn set of 100 images and their reconstructions. The images were different for each participant; consequently, a total of 2000 images were judged. Participants preferred the SSIM- over MSE-optimized reconstructions by nearly a 7:1 ratio: the SSIM reconstruction was chosen for 86.25% of the images. Individual participants chose SSIM reconstruction between 63% and 99% of trials.

3.2.2 Convolutional Autoencoders

We also performed a third Mechanical Turk study, this time on the convolutional autoencoders, to determine whether human observers prefer images generated by the MS-SSIM-optimized networks to MSE- and MAE-optimized networks. Images were chosen randomly from the STL-10 validation set. Participants were presented with a sequence of screens showing the original (reference) image on the left

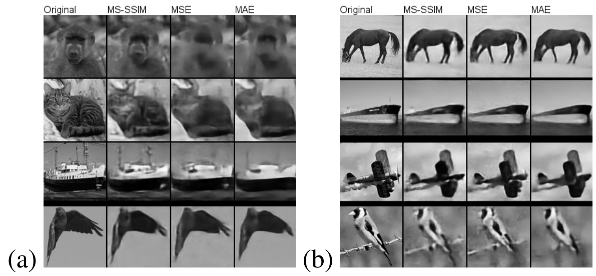


Figure 4. (a) Four randomly selected, held-out STL-10 images and their reconstructions for the 128-hidden-unit networks. For these images, the MS-SSIM reconstruction was ranked as best by humans. (b) Four randomly selected test images where the MS-SSIM reconstruction was ranked second or third.

and a set of of three reconstructions on the right. Participants were instructed to drag and drop the images vertically into the correct order, so that the best reconstruction is on top and the worst on the bottom. The initial vertical ordering of reconstructions was randomized. We asked 20 participants to each rank 50 images, for a total of 1000 rankings. Figure 2b shows the distribution over rankings for each of the three training objectives. If participants chose randomly, one would expect to see the same number of high rankings for each model. However, MS-SSIM is ranked highest for a majority of images (709 out of 1000).

Figure 4a shows examples of images whose MS-SSIM reconstruction was ranked as best by human judges. Figure 4b shows examples of images whose MSE or MAE reconstruction was ranked as the best. The strong preference for MS-SSIM appears to be due to its superiority in capturing fine detail such as the monkey and cat faces and background detail such as the construction cranes. MS-SSIM seems to have less of an advantage on simpler, more homogeneous, less textured images. Note that even when MSE or MAE beats MS-SSIM, the MS-SSIM reconstructions have no obvious defects relative to the other reconstructions.

4. Probabilistic Autoencoders

In order to further explore the role of perceptual losses in learning models for image generation, we adapt the variational autoencoder (VAE) model of [16] to be trained with an arbitrary differentiable image similarity metric. The VAE closely resembles a standard autoencoder, utilizing a combination of an encoding network that produces a code for an image x, and then a decoding network that maps the code to an image $\hat{x}$. The key difference is that the code z is considered a latent variable, endowed with a prior $p(z)$. The encoder $q_{\boldsymbol{\phi}}(z|x)$, parameterized by $\boldsymbol{\phi}$, approximates the intractable posterior of z given the image x. The decoder $p_{\boldsymbol{\theta}}(x|z)$, parameterized by $\boldsymbol{\theta}$, produces a distribution over images given z. The VAE minimizes a variational upper bound on the negative log-likelihood of the data:

$$\mathcal{L}^{VAE} = \mathbb{E}_{q_{\phi}}[-\log p_{\theta}(x|z)] + D_{KL}(q_{\phi}(z|x)||p(z))$$

We modify this learning objective to better suit an arbitrary differential loss $\Delta(x, \hat{x})$ by replacing the probabilistic decoder with a deterministic prediction as a function of the code: $\hat{x} \equiv f_{\theta}(z)$. The objective then becomes a weighted sum of the expected loss of $\hat{x}$ under the encoder's distribution over z and the KL regularization term:

$$\mathcal{L}^{EL} = C \cdot \mathbb{E}_{q_{\phi}}[\Delta(x, \hat{x})] + D_{KL}(q_{\phi}(z|x)||p(z)) \quad (1)$$

where the constant C governs the trade-off between the image-specific loss and the regularizer. We call this modification *Expected-Loss VAE* (EL-VAE).

4.1. EL-VAE Training Methodology

We trained convolutional EL-VAE networks with 128-dimensional z on 96×96 pixel images from the unlabeled portion of STL-10 with a similar architecture to the deterministic convolutional autoencoders. One key choice when training EL-VAEs is the value of C in Equation 1, which governs the trade-off between the KL loss and reconstruction error. As C increases, the model will put greater emphasis on reconstructions. At the same time, the KL-divergence of the prior from the approximate posterior will increase, leading to poorer samples. Selecting a value of C is further complicated due to the different scaling depending on the choice of the image-specific loss Δ.

In order to mitigate the differences in scaling, we normalized each loss (MSE, MAE, and MS-SSIM) by dividing by its expected value as estimated by computing the loss on 10,000 pairs randomly drawn with replacement from the training set. To select the best value of C, we utilized a recent approach to model selection in generative models [2]. This work proposes a statistical test of relative similarity to determine which model generates samples that are significantly closer to the reference dataset of interest. The test statistic is the difference in squared maximum mean discrepancies (MMDs) between the reference dataset and a dataset generated by each model. We trained convolutional EL-VAEs with $C \in \{1, 10, 1000, 10000\}$ for each loss on a 5,000 example subset of the STL-10 unlabeled dataset. We then utilized the test statistic of [2] to determine for each loss the value of C that produced samples with smallest squared MMD compared to the STL-10 train set. For each loss $C = 1000$ was selected by this test and thus we used this value when training EL-VAEs on the full unlabeled STL-10 dataset.

4.2. EL-VAE Results

We performed a final Mechanical Turk study to determine human observer preferences for image reconstructions generated by MS-SSIM-, MSE-, and MAE-optimized EL-VAE architecture. We generated reconstructions of 1000

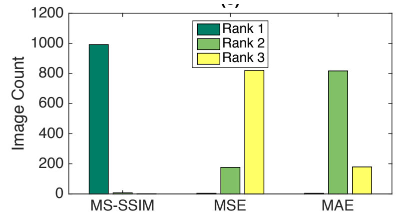
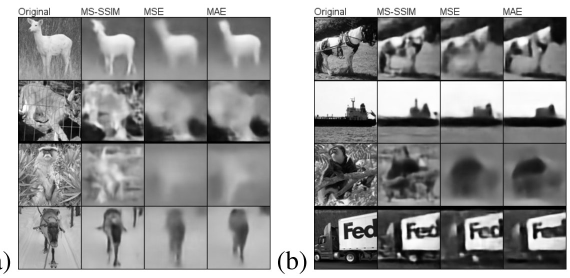


Figure 5. Image quality ranking for MS-SSIM, MSE-, and MAE-optimized EL-VAEs.



(a) (b)

Figure 6. (a) Four randomly selected, held-out STL-10 images and their reconstructions. For these images, the MS-SSIM reconstruction was ranked as best by humans. Reconstructions are from the 128-hidden-unit VAEs. From left to right are the original image, followed by the MS-SSIM, MSE, and MAE reconstructions. (b) Four randomly selected test images where the MS-SSIM reconstruction was ranked second or third.

randomly chosen images from the STL-10 test set by taking the latent code to be the mode of the approximate posterior for each EL-VAE network. The procedure was otherwise the same as detailed above for the deterministic case. MS-SSIM was ranked the highest in 992 out of 1000 cases. Figure 5 shows the distribution of image rankings for each loss. Test reconstructions for the EL-VAE networks are shown in Figure 6. Figure 6a shows reconstructions for which MS-SSIM was ranked as best, and Figure 6b shows reconstructions for which it was not ranked best. As observed in the deterministic case, MS-SSIM is better at capturing fine details than either MSE or MAE.

In order to qualitatively assess the performance of each EL-VAEs as a generative model, Figure 7 shows random samples from each model. Each image was generated by drawing a code z from the prior and then passing it as input to the decoder. The samples generated by the MS-SSIM-optimized net contain a great degree of detail and structure.

5. Classification with Learned Representations

In the previous sections, we showed that using a perceptually-aligned training objective improves the quality of image synthesis, as judged by human observers, for three different neural net architectures. In this section, we investigate whether the MS-SSIM objective leads to the discovery of internal representations in the neural net that are more closely tied to the factors of variation in images. For these experiments we use the Extended Yale B Faces dataset [18]. This dataset contains 2,414 grayscale images of 38 indi-

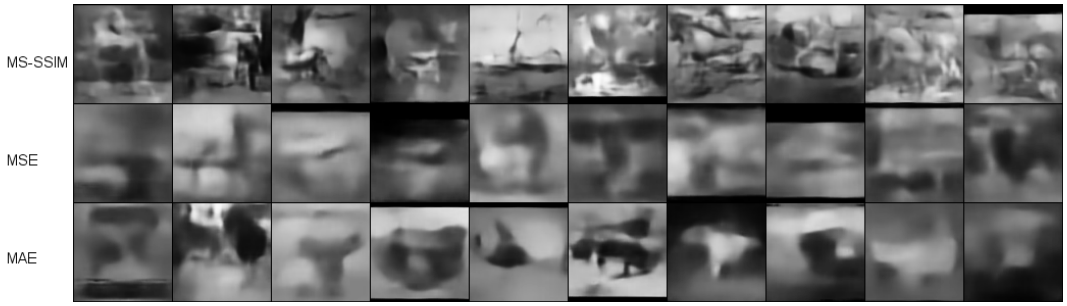


Figure 7. Samples generated from EL-VAEs optimized with MS-SSIM (top row), MSE (middle row) and MAE (bottom row). Sample quality appears to mirror reconstruction quality: the MS-SSIM optimized EL-VAE generates fine details that the other models do not.

viduals and is labeled with the azimuth ($-130°$ to $+130°$) and elevation ($-40°$ to $+90°$) of the light source in relation to the face. We resized the images to 48×48 and split the data randomly into a training, validation, and test set in a 60%-20%-20% ratio. We learned deterministic convolutional autoencoders using MSE, MAE, and MS-SSIM as loss functions and then used the bottleneck representations as features for SVMs trained to predict identity, azimuth, and elevation. We opted to investigate this prediction task as opposed to a more straightforward task (such as STL-10 classification accuracy) because we expect MS-SSIM to obtain superior encodings of low- and mid-level visual features such as edges and contours. Indeed, as predicted, initial studies showed only modest benefits of MS-SSIM for STL-10 classification accuracy, where coarse classification (e.g., plane versus ship) does not require fine image detail.

The deterministic convolutional autoencoders we trained on Yale B had a similar architecture to those described in Section 3.1. Here though we used a 32-unit bottleneck layer with ReLU activations, and used batch normalization on all layers except the output layer of the decoder. After training each of the autoencoders to convergence on the training set, we extracted bottleneck representations for the training and validation sets. We trained a SVM with a linear kernel to predict identity and SVR with RBF kernels to predict azimuth and elevation. Hyperparameters of the SVMs were selected via three-fold cross-validation on the training plus validation set. The resulting performance on the test set (Table 1) demonstrate that MS-SSIM yields robust representations of relevant image factors and thereby outperforms MSE and MAE.

Loss	Identity	Azimuth	Elevation
MSE	5.60%	277.46	51.46
MAE	5.60%	325.19	50.23
MS-SSIM	**3.53%**	**234.32**	**35.60**

Table 1. Test error for SVMs trained on bottleneck representations of deterministic convolutional autoencoders for Yale B. Classification error is the evaluation metric for identity prediction; MSE is the evaluation metric for azimuth and elevation prediction.

	Bicubic	MSE	MAE	MS-SSIM
SET5 PSNR	28.44	**30.52**	29.57	30.35
SSIM	0.8097	0.8621	0.8350	**0.8681**
SET14 PSNR	26.01	**27.53**	26.82	27.47
SSIM	0.7018	0.7512	0.7310	**0.7610**
BSD200 PSNR	25.92	**26.87**	26.47	26.84
SSIM	0.6952	0.7378	0.7220	**0.7484**

Table 2. Super-resolution imaging results.

6. Image Super-Resolution

We apply our perceptual loss to the task of super-resolution (SR) imaging. As a baseline model, we use a state-of-the-art SR method, the SRCNN [9]. We used the SRCNN architecture determined to perform best in [9]. It consists of 3 convolutional layers and 2 fully connected layers of ReLUs, with 64, 32, and 1 filters in the convolutional layers, from bottom to top, and filter sizes 9, 5, and 5. All the filters coefficients are initialized with draws from a zero-mean Gaussian with standard deviation 0.001.

We construct a training set in a similar manner as [9] by randomly cropping 5 million patches (size 33×33) from a subset of the ImageNet dataset of [7]. We compare three different loss functions for the SRCNN: MSE, MAE and MS-SSIM. Following [9], we evaluate the alternatives utilizing the standard metrics PSNR and SSIM. We tested $4\times$ SR with three standard test datasets—Set5 [1], Set14 [33] and BSD200 [21]. All measures are computed on the Y channel of YCbCr color space, averaged over the test set. As shown in Table 2, MS-SSIM achieves a PSNR comparable to that of MSE, and outperforms other loss functions significantly in the SSIM measure. Fig. 8 provides close-up visual illustrations.

7. Discussion and Future Work

We have investigated the consequences of replacing pixel-wise loss functions, MSE and MAE, with perceptually-grounded loss functions, SSIM and MS-SSIM,

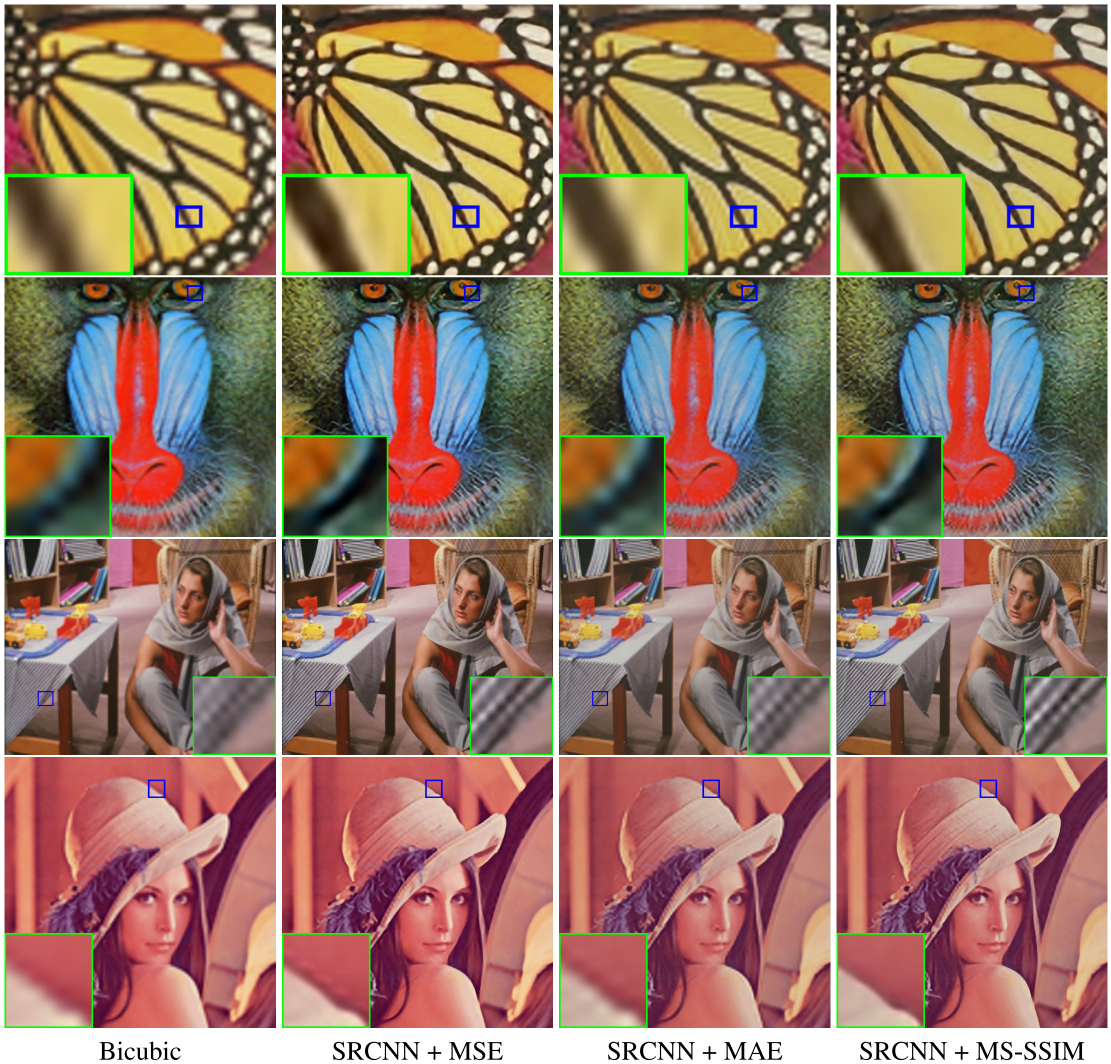


Figure 8. Visual comparisons on super-resolution at a magnification factor of 4. MS-SSIM not only improves resolution but also removes artifacts, *e.g*., the ringing effect in the bottom row, and enhances contrast, *e.g*., the fabric in the third row.

in neural networks that synthesize and transform images. Human observers judge SSIM-optimized images to be of higher quality than PL-optimized images over a range of neural network architectures. We also found that perceptually-optimized representations are better suited for predicting content-related image attributes. Finally, our promising results on single-image super-resolution highlight one of the key strengths of perceptual losses: they can easily be applied to current state-of-the-art architectures by simply substituting in for a pixel loss such as MSE. Taken together, our results support the hypothesis that the MS-SSIM loss encourages networks to encode relevant low- and mid-level structure in images. Consequently, we conjecture that the MS-SSIM trained representations may even be useful for fine-grained classification tasks such as bird species identification, in which small details are important.

A recent manuscript [34] also proposed using SSIM and MS-SSIM as a training objective for image processing neural networks. In this manuscript, the authors evaluate alternative training objectives based not on human judgments, but on a range of image quality metrics. They find that MAE outperforms MSE, SSIM, and MS-SSIM on their collection of metrics, and not surprisingly, that a loss which combines both PL and SSIM measures does best—on the collection of metrics which include PL and SSIM measures. Our work goes further in demonstrating that perceptually-grounded losses attain better scores on the definitive assessment of image quality: that registered by the human visual cortex.

Given our encouraging results, it seems appropriate to investigate other perceptually-grounded loss functions. SSIM is the low-hanging fruit because it is differentiable. Nonetheless, even black-box loss functions can be cached into a *forward model* neural net [15] that maps image pairs into a quality measure. We can then back propagate through the forward model to transform a loss derivative expressed in perceptual quality into a loss derivative expressed in terms of individual output unit activities. This flexible framework will allow us to combine multiple perceptually-grounded loss functions. Further, we can refine any perceptually-grounded loss functions with additional data obtained from human preference judgments, such as those we collected in the present set of experiments.

References

[1] M. Bevilacqua, A. Roumy, C. Guillemot, and M.-L. A. Morel. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. In *BMVC*, 2012. 7

[2] W. Bounliphone, E. Belilovsky, M. B. Blaschko, I. Antonoglou, and A. Gretton. A test of relative similarity for model selection in generative models. *arXiv preprint arXiv:1511.04581*, 2016. 6

[3] D. M. Chandler and S. S. Hemami. Vsnr: A wavelet-based visual signal-to-noise ratio for natural images. *IEEE Transactions on Image Processing*, 16(9):2284–2298, 2007. 3

[4] A. Coates, A. Y. Ng, and H. Lee. An analysis of single-layer networks in unsupervised feature learning. In *International conference on artificial intelligence and statistics*, pages 215–223, 2011. 4

[5] S. J. Daly. Visible differences predictor: an algorithm for the assessment of image fidelity. In *SPIE/IS&T 1992 Symposium on Electronic Imaging: Science and Technology*, pages 2–15. International Society for Optics and Photonics, 1992. 3

[6] P. Dayan, G. E. Hinton, R. M. Neal, and R. S. Zemel. The Helmholtz machine. *Neural Computation*, 7(5):889–904, 1995. 2

[7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 7

[8] E. Denton, S. Chintala, A. Szlam, and R. Fergus. Deep Generative Image Models using a Laplacian Pyramid of Adversarial Networks. *arXiv 1506.05751 [stat.ML]*, pages 1–10, 2015. 2

[9] C. Dong, C. C. Loy, K. He, and X. Tang. Image super-resolution using deep convolutional networks. *IEEE TPAMI*, 38(2):295–307, 2016. 7

[10] T. Frese, C. A. Bouman, and J. P. Allebach. Methodology for designing image similarity metrics based on human visual system models. In *Electronic Imaging'97*, pages 472–483. International Society for Optics and Photonics, 1997. 3

[11] I. J. Goodfellow, J. Pouget-Abadie, and M. Mirza. Generative adversarial networks. *arXiv 1406.266v1 [stat.ML]*, pages 1–9, 2014. 2

[12] M. Hassan and C. Bhagvati. Structural Similarity Measure for Color Images. *International Journal of Computer Applications (0975 8887)*, 43(14):7–12, 2012. 3

[13] G. E. Hinton, S. Osindero, and Y. W. Teh. A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7):1527–54, 2006. 2

[14] G. E. Hinton and T. J. Sejnowski. Learning and relearning in boltzmann machines. In D. E. Rumelhart and J. L. McClelland, editors, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Volume 1: Foundations*, volume 1, pages 283–317. MIT Press, Cambridge, MA, 1986. 2

[15] M. I. Jordan and D. E. Rumelhart. Forward models: Supervised learning with a distal teacher. *Cognitive Science*, 16(3):307–354, 1992. 8

[16] D. P. Kingma and M. Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations – ICLR*, 2014. 2, 4, 5

[17] A. Kolaman and O. Yadid-Pecht. Quaternion structural similarity: a new quality index for color images. *Image Processing, IEEE Transactions on*, 21(4):1526–1536, 2012. 3

[18] K.-C. Lee, J. Ho, and D. J. Kriegman. Acquiring linear subspaces for face recognition under variable lighting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(5):684–698, 2005. 6

[19] Y. Li, K. Swersky, and R. Zemel. Generative Moment Matching Networks. In *Proceedings of The 32nd International Conference on Machine Learning*, pages 1718–1727, 2015. 2

[20] J. Lubin. A human vision system model for objective image fidelity and target detectability measurements. In *Proc. EUSIPCO*, volume 98, pages 1069–1072, 1998. 3

[21] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *ICCV*, 2001. 7

[22] J. Masci, U. Meier, D. Cireşan, and J. Schmidhuber. Stacked convolutional auto-encoders for hierarchical feature extraction. In *Artificial Neural Networks and Machine Learning–ICANN 2011*, pages 52–59. Springer, 2011. 4

[23] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015. 2

[24] H. R. Sheikh and A. C. Bovik. Image information and visual quality. *IEEE Transactions on Image Processing*, 15(2):430–444, 2006. 3

[25] P. Smolensky. Information processing in dynamical systems: Foundations of harmony theory. In D. E. Rumelhart and J. L. McClelland, editors, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Volume 1*, pages 194–281. MIT Press, Cambridge, MA, 1986. 2

[26] L. Theis, A. van den Oord, and M. Bethge. A note on the evaluation of generative models. In *International Conference on Learning Representations*, 2016. 2

[27] A. Torralba, R. Fergus, and W. T. Freeman. 80 million tiny images: A large data set for nonparametric object and scene recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(11):1958–1970, 2008. 4

[28] C. J. Van den Branden Lambrecht and O. Verscheure. Perceptual quality measure using a spatiotemporal model of the human visual system. In *Electronic Imaging: Science & Technology*, pages 450–461. International Society for Optics and Photonics, 1996. 3

[29] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to struc-

tural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. 3

[30] Z. Wang and E. P. Simoncelli. Maximum differentiation (mad) competition: A methodology for comparing computational models of perceptual quantities. *Journal of Vision*, 8(12):8, 2008. 3

[31] Z. Wang, E. P. Simoncelli, and A. C. Bovik. Multi-scale structural similarity for image quality assessment. *IEEE Asilomar Conference on Signals, Systems and Computers*, 2:9–13, 2003. 1, 3

[32] S. Winkler. A perceptual distortion metric for digital color images. In *ICIP (3)*, pages 399–403, 1998. 3

[33] R. Zeyde, M. Elad, and M. Protter. On single image scale-up using sparse-representations. In *International conference on curves and surfaces*, pages 711–730. Springer, 2010. 7

[34] H. Zhao, O. Gallo, I. Frosio, and J. Kautz. Is l2 a good loss function for neural networks for image processing? *arXiv preprint arXiv:1511.08861*, 2015. 8