Pixel Recursive Super Resolution

Ryan Dahl *   Mohammad Norouzi   Jonathon Shlens

Google Brain

{rld,mnorouzi,shlens}@google.com

Abstract

Super resolution is the problem of artificially enlarging a low resolution photograph to recover a plausible high resolution version. In the regime of high magnification factors, the problem is dramatically underspecified and many plausible, high resolution images may match a given low resolution image. In particular, traditional super resolution techniques fail in this regime due to the multimodality of the problem and strong prior information that must be imposed on image synthesis to produce plausible high resolution images. In this work we propose a new probabilistic deep network architecture, a pixel recursive super resolution model, that is an extension of PixelCNNs to address this problem. We demonstrate that this model produces a diversity of plausible high resolution images at large magnification factors. Furthermore, in human evaluation studies we demonstrate how previous methods fail to fool human observers. However, high resolution images sampled from this probabilistic deep network do fool a naive human observer a significant fraction of the time.

1. Introduction

The problem of *super resolution* entails artificially enlarging a low resolution photograph to recover a corresponding plausible image with higher resolution [31]. When a small magnification is desired (*e.g.,* 2×), super resolution techniques achieve satisfactory results [42, 8, 16, 40, 22] by building statistical prior models of images [36, 2, 52] that capture low-level characteristics of natural images.

This paper studies super resolution with particularly small inputs and large magnification ratios, where the amount of information available to accurately construct a high resolution image is very limited (Figure 1, left column). Thus, the problem is underspecified and many plausible, high resolution images may match a given low resolution input image. Building improved models for state-of-the-art in super resolution in the high magnification regime
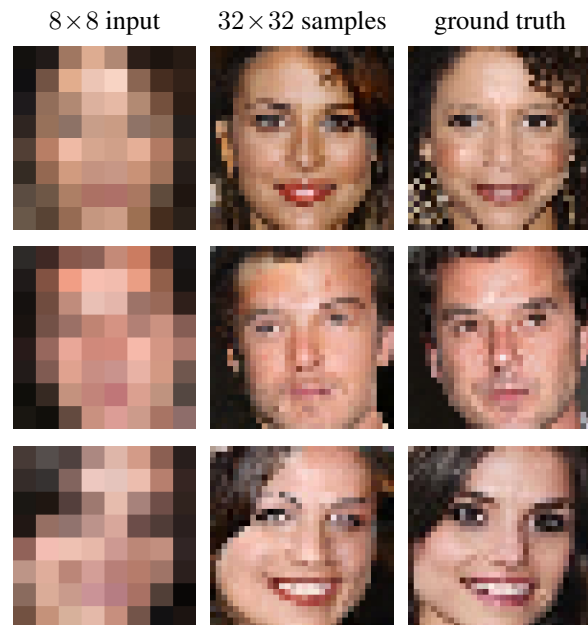

Figure 1: Illustration of our probabilistic pixel recursive super resolution model trained end-to-end on a dataset of celebrity faces. The left column shows 8×8 low resolution inputs from the test set. The middle and last columns show 32 × 32 images as predicted by our model *vs.* the ground truth. Our model incorporates strong face priors to synthesize realistic hair and skin details.

is significant for improving the state-of-art in super resolution, and more generally for building better conditional generative models of images [45, 33, 30, 44].

As the magnification ratio increases, a super resolution model need not only account for textures, edges, and other low-level statistics [16, 40, 22], but must increasingly account for complex variations of objects, viewpoints, illumination, and occlusions. At increasing levels of magnification, the details do not exist in the source image anymore, and the predictive challenge shifts from recovering details (*e.g.,* deconvolution [23]) to synthesizing plausible novel

details *de novo* [33, 45].

Consider a low resolution image of a face in Figure 1, left column. In such 8×8 pixel images the fine spatial details of the hair and the skin are missing and cannot be faithfully restored with interpolation techniques [15]. However, by incorporating prior knowledge of faces and their typical variations, a sketch artist might be able to imagine and draw believable details using specialized software packages [25].

In this paper, we show how a fully *probabilistic* model that is trained *end-to-end* using a log-likelihood objective can play the role of such an artist by synthesizing 32×32 face images depicted in Figure 1, middle column. We find that drawing multiple samples from this model produces high resolution images that exhibit multi-modality, resembling the diversity of images that plausibly correspond to a low resolution image. In human evaluation studies we demonstrate that naive human observers can easily distinguish real images from the outputs of sophisticated super resolution models using deep networks and mean squared error (MSE) objectives [21]. However, samples drawn from our probabilistic model are able fool a human observer up to 27.9% of the time – compared to a chance rate of 50%.

In summary, the main contributions of the paper include:

- Characterization of the *underspecified* super resolution problem in terms of multi-modal prediction.

- Proposal of a new probabilistic model tailored to the super resolution problem, which produces diverse, plausible non-blurry high resolution samples.

- Proposal of a new loss term for conditional probabilistic models with powerful autoregressive decoders to avoid the conditioning signal to be ignored.

- Human evaluation demonstrating that traditional metrics in super resolution (*e.g.,* pSNR and SSIM) fail to capture sample quality in the regime of underspecified super resolution.

We proceed by describing related work, followed by explaining how the multi-modal problem is not addressed using traditional objectives. Then, we propose a new probabilistic model building on top of ResNet [14] and Pixel-CNN [44]. The paper highlights the diversity of high resolution samples generated by the model and demonstrates the quality of the samples through human evaluation studies.

2. Related work

Super resolution has a long history in computer vision [31]. Methods relying on interpolation [15] are easy to implement and widely used, however these methods suffer from a lack of expressivity since linear models cannot express complex dependencies between the inputs and outputs. In practice, such methods often fail to adequately predict high frequency details leading to blurry high resolution outputs.

Enhancing linear methods with rich image priors such as sparsity [2] or Gaussian mixtures [52] have substantially improved the quality of the methods; likewise, leveraging low-level image statistics such as edge gradients improves predictions [48, 42, 8, 16, 40, 22]. Much work has been done on algorithms that search a database of patches and combine them to create plausible high frequency details in zoomed images [9, 17]. Recent patch-based work has focused on improving basic interpolation methods by building a dictionary of pre-learned filters on images and selecting the appropriate patches by an efficient hashing mechanism [35]. Such dictionary methods have improved the inference speed while being comparable to state-of-the-art.

Another approach for super resolution is to abandon inference speed requirements and focus on constructing the high resolution images at increasingly higher magnification factors. Convolutional neural networks (CNNs) represent an approach to the problem that avoids explicit dictionary construction, but rather implicitly extracts multiple layers of abstractions by learning layers of filter kernels. Dong *et al.* [7] employed a three layer CNN with MSE loss. Kim *et al.* [21] improved accuracy by increasing the depth to 20 layers and learning only the residuals between the high resolution image and an interpolated low resolution image. Most recently, SRResNet [26] uses many ResNet blocks to achieve state of the art pSNR and SSIM on standard super resolution benchmarks–we employ a similar design for our conditional network and catchall regression baseline.

Instead of using a per-pixel loss, Johnson *et al.*[18] use Euclidean distance between activations of a pre-trained CNN for model's predictions *vs.* ground truth images. Using this so-called preceptual loss, they train feed-forward networks for super resolution and style transfer. Bruna *et al.* [4] also use perceptual loss to train a super resolution network, but inference is done via gradient propagation to the low-res input (*e.g.,* [12]).

Another promising direction has been to employ an adversarial loss for training a network. A super-resolution network is trained in opposition to a secondary network that attempts to discriminate whether or not a synthesized high resolution image is real or fake. Networks trained with traditional L_p losses (e.g. [21, 7]) suffer from blurry images, where as networks employing an adversarial loss predict compelling, high frequency detail [26, 50]. Sønderby *et al.* [19] employed networks trained with adversarial losses but constrained the network to learn affine transformations that ensures the model only generate images that downscale back to the low resolution inputs. Sønderby *et al.* [19] also explore a masked autoregressive model but without the gated layers and using a mixture of gaussians instead of a multinomial distribution. Denton *et al.* [5] use a multi-scale adversarial network for image synthesis that is amenable for super-resolutions tasks.

Although generative adversarial networks (GANs) [13] provide a promising direction, such networks suffer from several drawbacks: first, training an adversarial network is unstable [33] and many methods are being developed to increase the robustness of training [29]. Second, GANs suffer from a common failure case of mode collapse [29] where by the resulting model produces samples that do not capture the diversity of samples available in the training data. Finally, tracking the performance of adversarial networks is challenging because it is difficult to associate a probabilistic interpretation to their results. These points motivate approaching the problem with a distinct approach to permit covering of the full diversity of the training dataset.

PixelRNN and PixelCNN [44, 45] are probabilistic generative models that impose an order on image pixels in order to represent them as a long sequence. The probability of subsequent pixels is conditioned on previously observed pixels. One variant of PixelCNN [45] obtained state-of-the-art predictive ability in terms of log-likelihood on academic benchmarks such as CIFAR-10 and MNIST. Since Pixel-CNN uses log-likelihood for training, the model is penalized if negligible probability is assigned to any of the training examples. By contrast, adversarial networks only learn enough to fool a non-stationary discriminator. This latter point suggests that a PixelCNN might be able to predict a large diversity of high resolution images that might be associated with a given low resolution image. Further, using log-likelihood as the training objective allows for hyper parameter search to find models within a model family by simply comparing their log probabilities on a validation set.

3. Probabilistic super resolution

We aim to learn a probabilistic super resolution model that discerns the statistical dependencies between a high resolution image and a corresponding low resolution image. Let $\mathbf{x}$ and $\mathbf{y}$ denote a low resolution and a high resolution image, and let $\mathbf{y}^*$ represent a ground-truth high resolution image. In order to learn a parametric model of $p_\theta(\mathbf{y} \mid \mathbf{x})$, we exploit a large dataset of pairs of low resolution inputs and ground-truth high resolution outputs, denoted $\mathcal{D} \equiv \{(\mathbf{x}^{(i)}, \mathbf{y}^{*(i)})\}_{i=1}^N$. One can easily collect such a large dataset by starting from some high resolution images and lowering the resolution as much as needed. To optimize the parameters $\boldsymbol{\theta}$ of the conditional distribution p, we maximize a conditional log-likelihood objective defined as,

$$O(\boldsymbol{\theta} \mid \mathcal{D}) = \sum_{(\mathbf{x}, \mathbf{y}^*) \in \mathcal{D}} \log p(\mathbf{y}^* \mid \mathbf{x}) . \qquad (1)$$

The key problem discussed in this paper is the exact form of $p(\mathbf{y} \mid \mathbf{x})$ that enables efficient learning and inference, while generating realistic non-blurry outputs. We first discuss pixel-independent models that assume that each output pixel is generated with an independent stochastic process given the input. We elaborate why these techniques result in sub-optimal blurry super resolution results. Then, we describe our pixel recursive super resolution model that generates output pixels one at a time to enable modeling the statistical dependencies between the output pixels, resulting in sharp synthesized images given very low resolution inputs.

3.1. Pixel independent super resolution

The simplest form of a probabilistic super resolution model assumes that the output pixels are conditionally independent given the inputs. As such, the conditional distribution of $p(\mathbf{y} \mid \mathbf{x})$ factors into a product of independent pixel predictions. Suppose an RGB output $\mathbf{y}$ has M pixels each with three color channels, *i.e.,* $\mathbf{y} \in \mathbb{R}^{3M}$. Then,

$$\log p(\mathbf{y} \mid \mathbf{x}) = \sum_{i=1}^{3M} \log p(y_i \mid \mathbf{x}) . \qquad (2)$$

Two general forms of pixel prediction models have been explored in the literature: *Gaussian* and *multinomial* distributions to model continuous and discrete pixel values respectively. In the Gaussian case,

$$\log p(y_i \mid \mathbf{x}) = -\frac{1}{2\sigma^2} \|y_i - C_i(\mathbf{x})\|_2^2 - \log \sqrt{2\sigma^2 \pi} , \quad (3)$$

where $C_i(\mathbf{x})$ denotes the ith element of a non-linear transformation of $\mathbf{x}$ via a convolutional neural network. Accordingly, $C_i(\mathbf{x})$ is the estimated mean for the ith output pixel y_i, and σ^2 denotes the variance. Often the variance is not learned, in which case maximizing the conditional log-likelihood of (1) reduces to minimizing the MSE between y_i and $C_i(\mathbf{x})$ across the pixels and channels throughout the dataset. Super resolution models based on MSE regression fall within this family of pixel independent models [7, 21, 26]. Implicitly, the outputs of a neural network parameterize a set of Gaussians with fixed variance. It is easy to verify that the joint distribution $p(y \mid \mathbf{x})$ is unimodal as it forms an isotropic multi-variate Gaussian.

Alternatively, one could discrete the output dimensions into K possible values (*e.g.,* $K = 256$), and use a multinomial distribution as the predictive model for each pixel [51], where $y_i \in \{1, \ldots, K\}$. The pixel prediction model based on a multinomial softmax operator is represented as,

$$p(y_i = k \mid \mathbf{x}) = \frac{\exp\{C_{ik}(\mathbf{x})\}}{\sum_{v=1}^K \exp\{C_{iv}(\mathbf{x})\}} , \qquad (4)$$

where a network with a set of softmax weights, $\{\mathbf{w}_{jk}\}_{j=1, k=1}^{3, K}$, for each value per color channel is used to induce $C_{ik}(\mathbf{x})$. Even though $p(y_i \mid \mathbf{x})$ in (4) can express multimodal distributions, the conditional dependency between the pixels cannot be captured, *i.e.,* the model cannot choose between drawing an edge at one position *vs.* another since that requires coordination between the samples.

How the dataset was created

Samples from trained model

L_2 regression
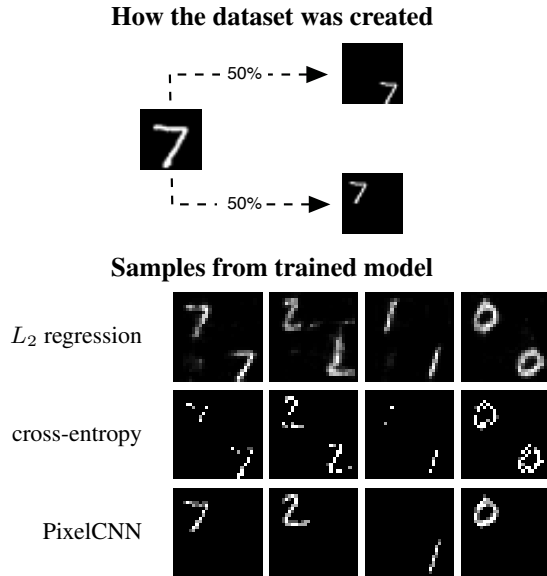
cross-entropy

PixelCNN

Figure 2: Simulated dataset demonstrates challenge of multimodal prediction. *Top*: Synthesized dataset in which samples are randomly translated to top-left or bottom-right corners. *Bottom*: Example predictions for various algorithms trained on this dataset. The pixel independent L_2 regression and cross-entropy models fail to predict a single mode but instead predict a blend of two spatial locations even though such samples do not exist in the training set. Conversely, the PixelCNN stochastically predicts the location of the digit at either corner with mutual exclusion.

3.2. Synthetic multimodal task

To demonstrate how pixel independent models fail at conditional image modeling, we create a synthetic dataset that explicitly requires multimodal prediction. For many dense image predictions tasks, e.g. super resolution [31], colorization [51, 6], and depth estimation [38], models that are able to predict a single mode are heavily preferred over models that blend modes together. For example, in the task of colorization selecting a strong red or green for an apple is better than selecting a brown-toned color that reflects the smeared average of all of the apple colors observed in the training set.

We construct a simple multimodal *MNIST corners* dataset to demonstrate the challenge of this problem. *MNIST corners* is constructed by randomly placing an MNIST digit in either the top-left or bottom-right corner (Figure 2, top). Several networks are trained to predict individual samples from this dataset to demonstrate the unique challenge of this simple example.

The challenge behind this toy example is for a network to exclusively predict an individual digit in a corner of an image. Training a moderate-sized 10-layer convolutional neu-

ral network ($\sim 100K$ parameters) with an L_2 objective (i.e. MSE regression) results in blurry image samples in which the two modes are blended together (Figure 2, L_2 *regression*). That is, *never* in the dataset does an example image contain a digit in both corners, yet this model incorrectly predicts a blend of such samples. Replacing the loss with a discrete, per-pixel cross-entropy produces sharper images but likewise fails to stochastically predict a digit in a corner of the image (Figure 2, *cross-entropy*). However, by using the chain rule to factor the multi-modal joint distribution into a product of sequential, conditional distributions, PixelCNN is able to learn dependencies between pixel values and generate good examples.

Other models like GANs or markov random fields could solve this task. GANs would likely produce both modes here correctly, but studies have quantified a significant degree of mode dropping in GANs when the number of modes is numerous [29], and we expect that the latter situation is the realm in which the under-specified super resolution problem lies. Pairwise MRFs with connections between pixels in the two corners of the output could learn mutual exclusion between the two sets of pixels for this dataset, but we did not explore MRFs experimentally due to intractability of high order potentials necessary for super resolution.

4. Pixel recursive super resolution

The lack of conditional independence between predicted pixels is a significant failure mode for the previous probabilistic objectives in the synthetic example (Equations 3 and 4). One approach to this problem is to define the conditional distribution of the output pixels jointly as a multivariate Gaussian mixture [53] or an undirected graphical model [10]. Both of these conditional distributions require constructing a statistical dependency between output pixels for which inference may be computationally expensive.

A second approach is to factorize the joint distribution using the chain rule by imposing an order on image pixels,

$$\log p(\mathbf{y} \mid \mathbf{x}) = \sum_{i=1}^{M} \log p(\mathbf{y}_i \mid \mathbf{x}, \mathbf{y}_{<i}), \qquad (5)$$

where the generation of each output dimension is conditioned on the input and previous output pixels [24, 43]. We denote the conditioning[1] up to pixel i by $\mathbf{y}_{<i}$ where $\{\mathbf{y}_1, \ldots, \mathbf{y}_{i-1}\}$. The benefits of this approach are that the exact form of the conditional dependencies is flexible and the inference is straightforward.

PixelCNN is a stochastic model that provides an explicit model for $\log p(\mathbf{y}_i \mid \mathbf{x}, \mathbf{y}_{<i})$ as a gated, hierarchical

---

[1]Note that in color images one must impose an order on both spatial locations as well as color channels. In a color image the conditioning is based on the the input and previously outputted pixels at previous spatial locations as well as pixels at the same spatial location.

chain of cleverly masked convolutions [44, 45, 37]. The goal of PixelCNN is to capture multi-modality and capture pixel correlations in an image. Indeed, training a PixelCNN on the *MNIST corners* dataset successfully captures the bimodality of the problem and produces sample in which digits reside exclusively in a single corner (Figure 2, *PixelCNN*). Importantly, the model never predicts both digits simultaneously.

Applying the PixelCNN to a super-resolution problem is a straightforward application that requires modifying the architecture to supply a conditioning on a low resolution version of the image. In early experiments we found the auto-regressive distribution of the model largely ignore the conditioning of the low resolution image. This phenomenon referred to as "optimization challenges" has been readily documented in the context of sequential autoencoder models [3] (see also [39, 41] for more discussion).

To address this issue we modify the architecture of PixelCNN to more explicitly depend on the conditioning of a low resolution image. In particular, we propose a late fusion model [20] that factors the problem into auto-regressive and conditioning components (Figure 3). The auto-regressive portion of the model, termed a *prior network* captures the serial dependencies of the pixels while the conditioning component, termed a *conditioning network* captures the global structure of the low resolution image. Specifically, we formulate the prior network to be a PixelCNN and the conditioning network to be a deep convolutional network employed previously for super resolution [26].

Given an input $\mathbf{x} \in \mathbb{R}^L$, let $A_i(\mathbf{x}) : \mathbb{R}^L \to \mathbb{R}^K$ denote a conditioning network predicting a vector of logit values corresponding to the K possible values that the ith output pixel can take. Similarly, let $B_i(\mathbf{y}_{<i}) : \mathbb{R}^{i-1} \to \mathbb{R}^K$ denote a prior network predicting a vector of logit values for the ith output pixel. Our probabilistic model predicts a distribution over the ith output pixel by simply adding the two sets of logits and applying a softmax operator on them,

$$ p(y_i \mid \mathbf{x}, \mathbf{y}_{<i}) = \mathrm{softmax}(A_i(\mathbf{x}) + B_i(\mathbf{y}_{<i})) . \quad (6) $$

To optimize the parameters of A and B jointly, we perform stochastic gradient ascent to maximize the conditional log likelihood in (1). That is, we optimize a cross-entropy loss between the model's predictions in (6) and discrete ground truth labels $y_i^* \in \{1, \ldots, K\}$,

$$ O_1 = \sum_{(\mathbf{x},\mathbf{y}^*)\in\mathcal{D}} \sum_{i=1}^{M} \Big(\mathbb{1}[\mathbf{y}_i^*]^\mathsf{T}(A_i(\mathbf{x}) + B_i(\mathbf{y}_{<i}^*)) $$
$$ - \mathrm{lse}(A_i(\mathbf{x}) + B_i(\mathbf{y}_{<i}^*)) \Big) , \quad (7) $$

where $\mathrm{lse}(\cdot)$ is the log-sum-exp operator corresponding to the log of the denominator of a softmax, and $\mathbb{1}[k]$ denotes a
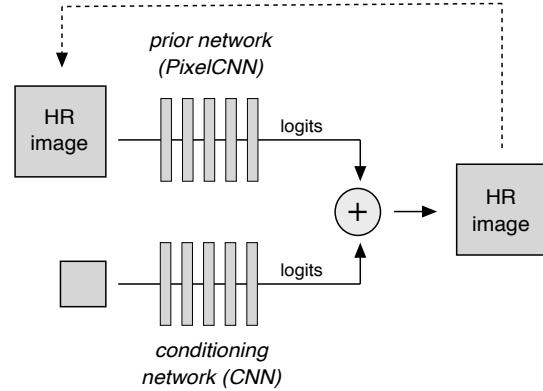

Figure 3: The proposed network comprises a *conditioning network* and a *prior network*. The *conditioning network* is a CNN that receives a low resolution image as input and outputs logits predicting the conditional log-probability of each high resolution (HR) image pixel. The *prior network*, a PixelCNN [45], makes predictions based on previous stochastic predictions (indicated by dashed line). The model's probability distribution is computed as a softmax operator on top of the sum of the two sets of logits from the two networks.

K-dimensional one-hot indicator vector with its kth dimension set to 1.

Our preliminary experiments indicate that models trained with (7) tend to ignore the conditioning network as the statistical correlation between a pixel and previous high resolution pixels is stronger than its correlation with low resolution inputs. To mitigate this issue, we include an additional loss in our objective to enforce the conditioning network to be optimized. This additional loss measures the cross-entropy between the conditioning network's predictions via $\mathrm{softmax}(A_i(\mathbf{x}))$ and ground truth labels. The total loss that is optimized in our experiments is a sum of two cross-entropy losses formulated as,

$$ O_2 = \sum_{(\mathbf{x},\mathbf{y}^*)\in\mathcal{D}} \sum_{i=1}^{M} \Big(\mathbb{1}[\mathbf{y}_i^*]^\mathsf{T}(2\,A_i(\mathbf{x}) + B_i(\mathbf{y}_{<i}^*)) $$
$$ - \mathrm{lse}(A_i(\mathbf{x}) + B_i(\mathbf{y}_{<i}^*)) - \mathrm{lse}(A_i(\mathbf{x})) \Big) . \quad (8) $$

Once the network is trained, sampling from the model is straightforward. Using (6), starting at $i = 1$, first we sample a high resolution pixel. Then, we proceed pixel by pixel, feeding in the previously sampled pixel values back into the network, and draw new high resolution pixels. The three channels of each pixel are generated sequentially in turn.

We additionally consider *greedy decoding*, where one always selects the pixel value with the largest probability and

sampling from a tempered softmax, where the concentration of a distribution p is adjusted by using a temperature parameter $\tau > 0$,

$$p_\tau = \frac{p^{(1/\tau)}}{\|p^{(1/\tau)}\|_1} \, .$$

To control the concentration of our sampling distribution $p(y_i \mid \mathbf{x}, \mathbf{y}_{<i})$, it suffices to divide the logits from A and B by a parameter τ. Note that as τ goes towards 0, the distribution converges to the mode.

4.1. Implementation details

We summarize the network architecture for the pixel recursive super resolution model. The conditioning architecture is similar in design to SRResNet [26]. The conditioning network is a feed-forward convolutional neural network that takes a low resolution image through a series of $18-30$ ResNet blocks [14] and transposed convolution layers [32]. The last layer uses a 1×1 convolution to increase the number of channels to predict a multinomial distribution over 256 possible color channel values for each sub-pixel. The prior network architecture consists of 20 gated PixelCNN blocks with 32 channels at each layer [45]. The final layer of the super-resolution network is a softmax operation over the sum of the activations from the conditioning and prior networks. The model is built by using TensorFlow [1] and trained across 8 GPUs with synchronous SGD updates. For training details and a complete list of architecture parameters, please see Supp. material.

5. Experiments

We assess the effectiveness of the proposed pixel recursive super resolution method on two datasets containing centrally cropped faces (CelebA [27]) and bedroom images (LSUN Bedrooms [49]). In both datasets we resize the images to 8×8 and 32×32 pixels with bicubic interpolation to provide the input $\mathbf{x}$ and output $\mathbf{y}$ for training and evaluation.

We compare our technique against three baselines including (1) **Nearest N.**; a nearest neighbor search baseline inspired by previous work on example-based super resolution [9], (2) **ResNet $\mathbf{L_2}$**; a deep neural network using Resnet blocks trained with MSE objective, and (3) **GAN**; a GAN based super resolution model implemented by [11] similar to [50]. We exclude the results of the GANbaseline on bedrooms dataset as they are not competitive, and the model was developed specifically for faces.

The Nearest N. baseline computes $\mathbf{y}$ for a sample $\mathbf{x}$ by searching the training set $\mathcal{D} = \{(\mathbf{x}^{(i)}, \mathbf{y}^{*(i)})\}_{i=1}^N$ for the nearest example indexed by $i^* = \operatorname{argmin}_i \|\mathbf{x}^{(i)} - \mathbf{x}\|_2^2$, and returns the high resolution counterpart $\mathbf{y}^{*(i^*)}$. The Nearest N. baseline is a representative result of exemplar based super resolution approaches, and helps us test whether the model performs a naive lookup from the training dataset.
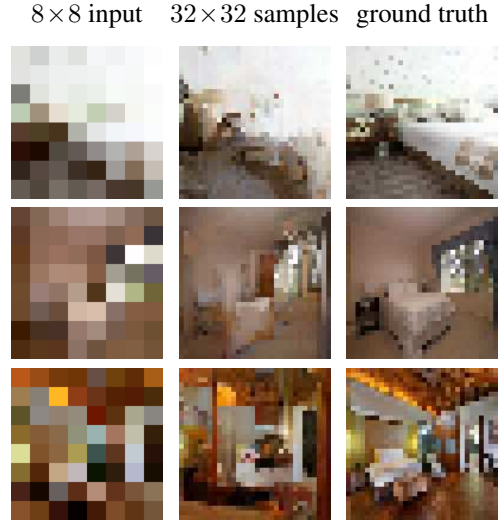
8×8 input   32×32 samples   ground truth

Figure 4: Illustration of our probabilistic pixel recursive super resolution model trained end-to-end on LSUN Bedrooms dataset.

The ResNet L_2 baseline employs a design similar to SR-ResNet [26] that reports state-of-the-art in terms of image similarity metrics[2]. Most significantly, we alter the network to compute the residuals with respect to a bicubic interpolation of the input [21]. The L_2 regression provides a comparison to a state-of-the-art convolutional network that performs a unimodal pixel independent prediction.

The GAN super resolution baseline [11] exploits a conditional GAN architecture, and combines an adversarial loss with a *consistency* loss, which encourages the low-resolution version of predicted $\mathbf{y}$ to be close to $\mathbf{x}$ as measures by L_1. There is a weighting between the two losses specified by [11] as 0.9 for the consistency and 0.1 for the adversarial loss, and we keep them the same in our face experiments.

5.1. Super resolution samples

High resolution samples generated by the pixel recursive super resolution capture the rich structure of the dataset and appear perceptually plausible (Figure 1 and 4; Supp. material). Sampling from the super resolution model multiple times results in different high resolution images for a given low resolution image (Figure 5 and Supp. material). Qualitatively, the samples from the model identify many plausible high resolution images with distinct qualitative features that correspond to a given lower resolution image. Note that the differences between samples for the faces dataset are far

---

[2] Note that the regression architecture is nearly identical to the conditioning network in Section 4.1. The slight change is to force the network to predict bounded values in RGB space. To enforce this behavior, the top layer is outputs three channels instead of one and employ a $\tanh(\cdot)$ instead of a ReLU$(\cdot)$ nonlinearity.
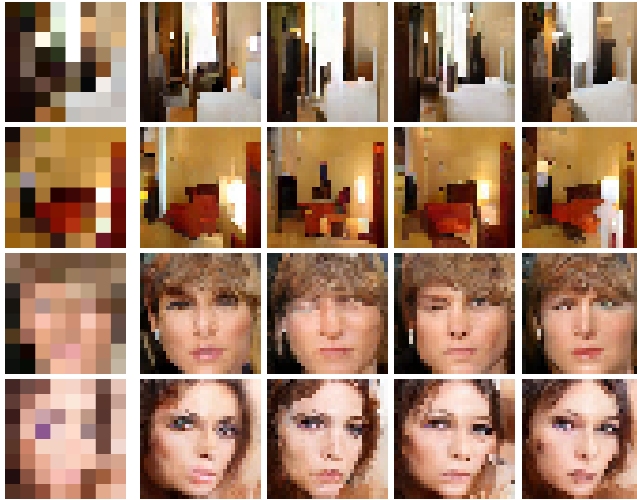
Figure 5: Diversity of samples from our model. Left column: Low resolution input. Right columns: Multiple samples at $\tau = 0.8$ conditioned upon low resolution input.

less drastic than seen in our synthetic dataset, where failure to cleanly predict modes indicated complete failure.

The quality of samples is sensitive to the temperature (Figure 6, right columns). Greedy decoding ($\tau = 0$) results in poor quality samples that are overly smooth and contain horizontal and vertical line artifacts. Samples from the default temperature ($\tau = 1.0$) are perceptually more plausible, although they tend to contain undesired high frequency content. Tuning the temperature (τ) between 0.9 and 0.8 proves beneficial for improving the quality of the samples.

It's worth noting that sampling is a relatively expensive task. A naive PixelCNN implementation performs full image convolutions for every generated pixel (quadratic complexity), and generating a 32x32 image takes about 40 seconds. Recent work [34] has shown that caching intermediate activation states can improve sample time dramatically.

5.2. Quantitative evaluation of image similarity

Many methods exist for quantifying image similarity that attempt to measure human perception judgements of similarity [46, 47, 28]. We quantified the prediction accuracy of our model compared to ground truth using pSNR and MS-SSIM (Table 1). We found that our own visual assessment of the predicted image quality did not correspond to these image similarities metrics. For instance, bicubic interpolation achieved relatively high metrics even though the samples appeared quite poor. This result matches recent observations that suggest that pSNR and SSIM provide poor judgements of super resolution quality when new details are synthesized [26, 18]. In addition, Figure 6 highlights how the perceptual quality of model samples do not necessarily correspond to negative log likelihood (NLL). Smaller NLL

means the model has assigned that image a larger probability mass. The greedy, bicubic, and regression faces are preferred by the model despite exhibiting worse perceptual quality.

We next measured how well the high resolution samples corresponded to the low resolution input by measuring the consistency. The consistency is quantified as L_2 distance between the low-resolution input image and a bicubic downsampled version of the high resolution estimate. Lower consistencies indicate superior correspondence with the low-resolution image. Note that this is an explicit objective the GAN [11]. The pixel recursive model achieved consistencies on par with the L_2 regression model and bicubic interpolation indicating that even though the model was producing diverse samples, the samples were largely constrained by the low-resolution image. Most importantly, the pixel recursive model achieved superior consistencies then the GAN [11] even though the model does not explicitly optimize for this criterion.[3]

The consistency measure additionally provided an important control experiment to determine if the pixel recursive model were just naively copying the nearest training sample. If the pixel recursive model were just copying the nearest training sample, then the consistency of the Nearest N. model would be equivalent to the pixel recursive model. We instead find that the pixel recursive model has superior consistency values indicating that the model is not just naively copying the closest training examples.

5.3. Perceptual evaluation with humans

Given that automated quantitative measures did not match our perceptual judgements, we conducted a human study to assess the effectiveness of the super resolution algorithm. In particular, we performed a forced choice experiment on crowd-sourced workers in order to determine how plausible a given high resolution sample is from each model. Following [51], each worker was presented a true image and a corresponding prediction from a model, and asked "*Which image, would you guess, is from a camera?*". We performed this study across 283 workers on Amazon Mechanical Turk and statistics were accrued across 40 unique workers for each super resolution algorithm.[4]

---

[3]Note that one may improve the consistency of the GAN by increasing its weight in the objective. Increasing the weight for the consistency term will likely lead to decreased perceptual quality in the images but improved consistency. Regardless, the images generated by the pixel recursive model are superior in both consistency and perceptual quality as judged humans for a range of temperatures.

[4]Specifically, each worker was given one second to make a forced choice decision. Workers began a session with 10 practice questions during which they received feedback. The practice pairs were not counted in the results. After the practice pairs, each worker was shown 45 additional pairs. A subset of the pairs were simple, *golden* questions designed to constantly check if the worker was paying attention. Data from workers that answered golden questions incorrectly were thrown out.

Input	G. Truth	Nearest N.	GAN [11]	Bicubic	ResNet L_2	Greedy	$\tau = 1.0$	$\tau = 0.9$	$\tau = 0.8$

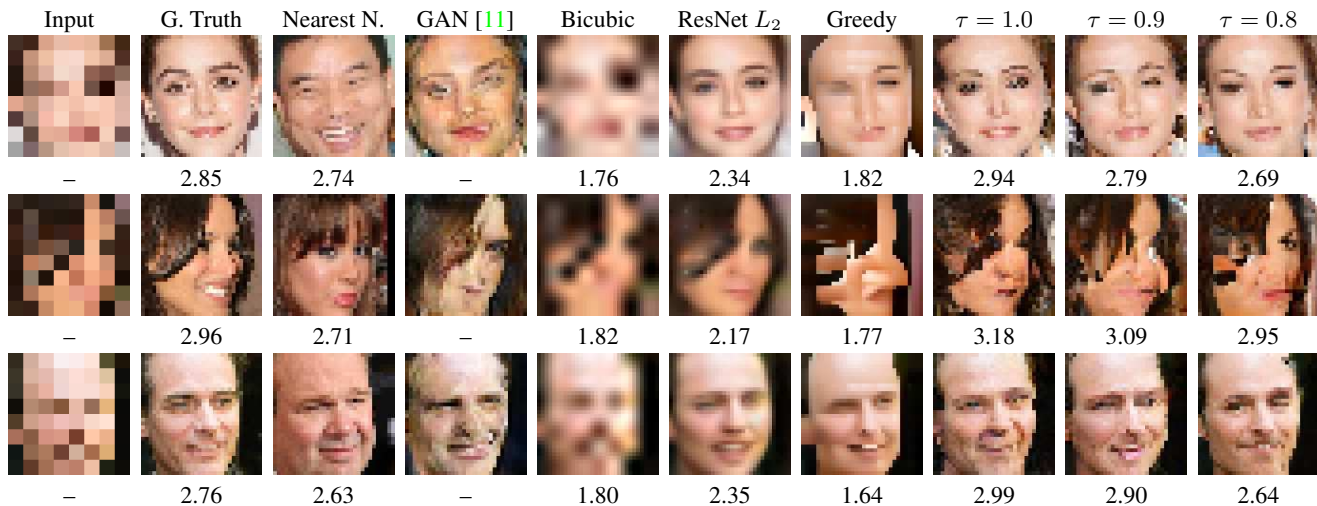| – | 2.85 | 2.74 | – | 1.76 | 2.34 | 1.82 | 2.94 | 2.79 | 2.69 |
| – | 2.96 | 2.71 | – | 1.82 | 2.17 | 1.77 | 3.18 | 3.09 | 2.95 |
| – | 2.76 | 2.63 | – | 1.80 | 2.35 | 1.64 | 2.99 | 2.90 | 2.64 |

Figure 6: Comparison of super resolution models. Columns from left to right include input, Ground truth, Nearest N. (nearest neighbor super resolution), GAN, bicubic upsampling, ResNet L_2(neural network optimized with MSE), greedy decoding is pixel recursive model, followed by sampling with various temperatures (τ) controlling the concentration of the predictive distribution. Negative log-probabilities are reported below the images. Note that the best log-probability is associated with bicubic upsampling and greedy decoding even though the images are poor quality.

CelebA	pSNR	SSIM	MS-SSIM	Consistency	% Fooled
Bicubic	28.92	0.84	0.76	0.006	–
Nearest N.	28.18	0.73	0.66	0.024	–
ResNet L_2	**29.16**	**0.90**	**0.90**	**0.004**	4.0 ± 0.2
GAN [11]	28.19	0.72	0.67	0.029	8.5 ± 0.2
$\tau = 1.0$	29.09	0.84	0.86	0.008	**11.0 ± 0.1**
$\tau = 0.9$	29.08	0.84	0.85	0.008	10.4 ± 0.2
$\tau = 0.8$	29.08	0.84	0.86	0.008	10.2 ± 0.1

LSUN	pSNR	SSIM	MS-SSIM	Consistency	% Fooled
Bicubic	**28.94**	0.70	0.70	**0.002**	–
Nearest N.	28.15	0.49	0.45	0.040	–
ResNet L_2	28.87	**0.74**	**0.75**	0.003	2.1 ± 0.1
$\tau = 1.0$	28.92	0.58	0.60	0.016	17.7 ± 0.4
$\tau = 0.9$	28.92	0.59	0.59	0.017	22.4 ± 0.3
$\tau = 0.8$	28.93	0.59	0.58	0.018	**27.9 ± 0.3**

Table 1: Test results on the cropped CelebA (top) and LSUN Bedroom (bottom) datasets magnified from 8×8 to 32×32. We report pSNR, SSIM, and MS-SSIM between samples and the ground truth. Consistency measures the MSE between the low-resolution input and a corresponding downsampled output. % Fooled reports measures how often the algorithms' outputs fool a human in a crowd sourced study; 50% would be perfectly confused.

Table 1 reports the percentage of samples for a given algorithm that a human incorrectly believed to be a real image. Note that a perfect algorithm would fool a human at rate of 50%. The L_2 regression model fooled humans 2-4% of the time and the GAN [11] fooled humans 8.5% of the time. The pixel recursive model fooled humans 11.0% and 27.9% of the time for faces and bedrooms, respectively – significantly above the state-of-the-art regression model. Importantly, we found that the selection of the sampling temperature τ greatly influenced the quality of the samples and in turn the fraction of time that humans were fooled. Nevertheless the pixel recursive model outperformed the strongest baseline model, the GAN, across all temperatures. A ranked list of the best and worst fooling examples is reproduced in the Supp. material along with the fool rates.

6. Conclusion

We present a fully probabilistic method that tackles super resolution with small inputs, demonstrating that even 8×8 images can be enlarged to sharp 32×32 images. Our technique outperforms several strong baselines including the ones optimizing a regression objective or an adversarial loss. We perform human evaluation studies showing that samples from the pixel recursive model look more plausible to humans, and more generally, common metrics like pSNR and SSIM do not correlate with human judgment when the magnification ratio is large.

Acknowledgments

References

[1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org. 6

[2] M. Aharon, M. Elad, and A. Bruckstein. Svdd: An algorithm for designing overcomplete dictionaries for sparse representation. *Trans. Sig. Proc.*, 54(11):4311–4322, Nov. 2006. 1, 2

[3] S. R. Bowman, L. Vilnis, O. Vinyals, A. M. Dai, R. Józefowicz, and S. Bengio. Generating sentences from a continuous space. *CoRR*, abs/1511.06349, 2015. 5

[4] J. Bruna, P. Sprechmann, and Y. LeCun. Super-resolution with deep convolutional sufficient statistics. *CoRR*, abs/1511.05666, 2015. 2

[5] E. L. Denton, S. Chintala, A. Szlam, and R. Fergus. Deep generative image models using a laplacian pyramid of adversarial networks. *NIPS*, 2015. 2

[6] A. Deshpande, J. Lu, M. Yeh, and D. A. Forsyth. Learning diverse image colorization. *CoRR*, abs/1612.01958, 2016. 4

[7] C. Dong, C. C. Loy, K. He, and X. Tang. Image super-resolution using deep convolutional networks. *CoRR*, abs/1501.00092, 2015. 2, 3

[8] R. Fattal. Image upsampling via imposed edge statistics. *ACM Trans. Graph.*, 26(3), July 2007. 1, 2

[9] W. T. Freeman, T. R. Jones, and E. C. Pasztor. Example-based super-resolution. *IEEE Computer graphics and Applications*, 2002. 2, 6

[10] W. T. Freeman and E. C. Pasztor. Markov networks for super-resolution. In *CISS*, 2000. 4

[11] D. Garcia. srez: Adversarial super resolution. https://github.com/david-gpu/srez, 2016. 6, 7, 8

[12] L. A. Gatys, A. S. Ecker, and M. Bethge. A neural algorithm of artistic style. *CoRR*, abs/1508.06576, 2015. 2

[13] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets, 2014. 3

[14] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *CVPR*, 2015. 2, 6

[15] H. Hou and H. Andrews. Cubic splines for image interpolation and digital filtering. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 26(6):508–517, Jan. 2003. 2

[16] J. Huang and D. Mumford. Statistics of natural images and models. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, volume 1. IEEE, 1999. 1, 2

[17] J.-B. Huang, A. Singh, and N. Ahuja. Single image super-resolution from transformed self-exemplars. In *IEEE Conference on Computer Vision and Pattern Recognition)*, 2015. 2

[18] J. Johnson, A. Alahi, and F. Li. Perceptual losses for real-time style transfer and super-resolution. *CoRR*, abs/1603.08155, 2016. 2, 7

[19] C. Kaae Sønderby, J. Caballero, L. Theis, W. Shi, and F. Huszár. Amortised MAP Inference for Image Super-resolution. *ArXiv e-prints*, Oct. 2016. 2

[20] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-scale video classification with convolutional neural networks. In *CVPR*, 2014. 5

[21] J. Kim, J. K. Lee, and K. M. Lee. Accurate image super-resolution using very deep convolutional networks. *CoRR*, abs/1511.04587, 2015. 2, 3, 6

[22] K. I. Kim and Y. Kwon. Single-image super-resolution using sparse regression and natural image prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(6):1127–1133, 2010. 1, 2

[23] D. Kundur and D. Hatzinakos. Blind image deconvolution. *IEEE signal processing magazine*, 13(3):43–64, 1996. 1

[24] H. Larochelle and I. Murray. The neural autoregressive distribution estimator. In *The Proceedings of the 14th International Conference on Artificial Intelligence and Statistics*, volume 15 of *JMLR: W&CP*, pages 29–37, 2011. 4

[25] K. R. Laughery and R. H. Fowler. Sketch artist and identi-kit procedures for recalling faces. *Journal of Applied Psychology*, 65(3):307, 1980. 2

[26] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi. Photo-realistic single image super-resolution using a generative adversarial network. *arXiv:1609.04802*, 2016. 2, 3, 5, 6, 7

[27] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2015. 6

[28] K. Ma, Q. Wu, Z. Wang, Z. Duanmu, H. Yong, H. Li, and L. Zhang. Group mad competition - a new methodology to compare objective image quality models. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 7

[29] L. Metz, B. Poole, D. Pfau, and J. Sohl-Dickstein. Unrolled generative adversarial networks. *CoRR*, abs/1611.02163, 2016. 3, 4

[30] M. Mirza and S. Osindero. Conditional generative adversarial nets. *CoRR*, abs/1411.1784, 2014. 1

[31] K. Nasrollahi and T. B. Moeslund. Super-resolution: A comprehensive survey. *Mach. Vision Appl.*, 25(6):1423–1468, Aug. 2014. 1, 2, 4

[32] A. Odena, V. Dumoulin, and C. Olah. Deconvolution and checkerboard artifacts. *Distill*, 2016. http://distill.pub/2016/deconv-checkerboard. 6

[33] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *CoRR*, abs/1511.06434, 2015. 1, 2, 3

[34] P. Ramachandran, T. L. Paine, P. Khorrami, M. Babaeizadeh, S. Chang, Y. Zhang, M. A. Hasegawa-Johnson, R. H. Campbell, and T. S. Huang. Fast generation for convolutional autoregressive models. *CoRR*, abs/1704.06001, 2017. 7

[35] Y. Romano, J. Isidoro, and P. Milanfar. RAISR: rapid and accurate image super resolution. *CoRR*, abs/1606.01299, 2016. 2

[36] S. Roth and M. J. Black. Fields of experts: A framework for learning image priors. *CVPR*, 2005. 1

[37] T. Salimans, A. Karpathy, X. Chen, D. P. Kingma, and Y. Bulatov. Pixelcnn++: A pixelcnn implementation with discretized logistic mixture likelihood and other modifications. under review at ICLR 2017. 5

[38] A. Saxena, S. H. Chung, and A. Y. Ng. Learning depth from single monocular images. In *In NIPS 18*. MIT Press, 2005. 4

[39] I. V. Serban, A. Sordoni, R. Lowe, L. Charlin, J. Pineau, A. C. Courville, and Y. Bengio. A hierarchical latent variable encoder-decoder model for generating dialogues. *CoRR*, abs/1605.06069, 2016. 5

[40] Q. Shan, Z. Li, J. Jia, and C.-K. Tang. Fast image/video upsampling. *ACM Transactions on Graphics (TOG)*, 27(5):153, 2008. 1, 2

[41] C. K. Sø nderby, T. Raiko, L. Maalø e, S. r. K. Sø nderby, and O. Winther. Ladder variational autoencoders. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 3738–3746. Curran Associates, Inc., 2016. 5

[42] J. Sun, Z. Xu, and H.-Y. Shum. Image super-resolution using gradient profile prior. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008. 1, 2

[43] B. Uria, I. Murray, and H. Larochelle. Rnade: The real-valued neural autoregressive density-estimator. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 2175–2183. Curran Associates, Inc., 2013. 4

[44] A. van den Oord, N. Kalchbrenner, and K. Kavukcuoglu. Pixel recurrent neural networks. *ICML*, 2016. 1, 2, 3, 5

[45] A. van den Oord, N. Kalchbrenner, O. Vinyals, L. Espeholt, A. Graves, and K. Kavukcuoglu. Conditional image generation with pixelcnn decoders. *NIPS*, 2016. 1, 2, 3, 5, 6

[46] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 7

[47] Z. Wang, E. P. Simoncelli, and A. C. Bovik. Multiscale structural similarity for image quality assessment. In *Signals, Systems and Computers, 2004. Conference Record of the Thirty-Seventh Asilomar Conference on*, volume 2, pages 1398–1402. Ieee, 2004. 7

[48] C. Y. Yang, S. Liu, and M. H. Yang. Structured face hallucination. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1099–1106, June 2013. 2

[49] F. Yu, Y. Zhang, S. Song, A. Seff, and J. Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015. 6

[50] X. Yu and F. Porikli. *Ultra-Resolving Face Images by Discriminative Generative Networks*, pages 318–333. Springer International Publishing, Cham, 2016. 2, 6

[51] R. Zhang, P. Isola, and A. A. Efros. Colorful image colorization. *ECCV*, 2016. 3, 4, 7

[52] D. Zoran and Y. Weiss. From learning models of natural image patches to whole image restoration. In *Proceedings of the 2011 International Conference on Computer Vision*, ICCV '11, pages 479–486, Washington, DC, USA, 2011. IEEE Computer Society. 1, 2

[53] D. Zoran and Y. Weiss. From learning models of natural image patches to whole image restoration. In *CVPR*, 2011. 4