Frequency Domain-based Perceptual Loss for Super Resolution

Shane Sims
University of British
Columbia, Department of
Computer Science
Vancouver, Canada
shane.sims@protonmail.com

ABSTRACT

We introduce Frequency Domain Perceptual Loss (FDPL), a loss function for single image super resolution. Unlike previous loss functions used for this task, which are all calculated in the pixel (spacial) domain, FDPL is computed in the frequency domain. By working in the frequency domain we can encourage a given model to learn a mapping that prioritizes those frequencies most related to human perception. While the goal of FDPL is not to maximize the Peak Signal to Noise Ratio (PSNR), we found that there is a correlation between decreasing FDPL and increasing PSNR. Training a model with FDPL results in a higher average PSRN (30.94), compared to the same model trained with pixel loss (30.59), as measured on the *Set5* image dataset. We also show that our method results in higher qualitative results, which is the goal of a perceptual loss function. However, it is not clear that the improved perceptual quality is due to the slightly higher PSNR or the perceptual nature of FDPL.

INTRODUCTION

Single image Super Resolution (SR) is a classic problem in computer vision, which seeks to recover a high-resolution image from a given low-resolution input. As a low-resolution image is typically generated via down-sampling vertical and/or horizontal rows of pixels, there may be no single solution to the problem. That is, many different high-resolution images can be down-sampled into the same low-resolution counterpart. The task of a SR model then is to learn what should exist between any two rows or columns of a given low-resolution image; a difficult task given the variation possible in even a small set of images.

Improvements in SR performance have been driven mainly by innovations in model architecture. Few previous works focus on adjusting the loss function to correspond with what is most important for a given use case. Janocha and Czarnecki focused on this issue directly (for classification) and showed that the choice of loss function has a direct effect on learning dynamics and what is ultimately learned by the model. [4]. Most recent work on SR has addressed the difficulty of the task by training image transformation Convolutional Neural Networks (CNNs) of varying complexity using a per-pixel loss function. Such losses minimize the distance between a low-resolution image and its ground truth target, which is measured by either ℓ_1 or ℓ_2 norm based functions. A notable exception to the use of pixel-loss is the work of Johnson et al. [5], which uses a pre-trained loss network and compares an intermediate activation layer of that network with the output of their image transformation network using mean squared error (MSE) between the two tensors.

In this work we also depart from using a pixel-loss and like Johnson et al., we construct a perceptual loss function to resolve images in a way that is more conducive to human perception. Where we differ from the previous SR perceptual loss work (and all others before ours) is that we compute loss in the frequency domain following a Fourier transform. Specifically, we use the Discrete Cosine Transform (DCT) [1], which enables us to use a perceptual model inspired by the JPEG image compression codec, the principals of which allow for excellent compression even without perceptible loss of quality. We use the same principles to guide our model to focus on resolving those parts of the image most important for human perception, by biasing the loss towards the frequency components lost during image down sampling and weighting these according to perceptual importance. While this work is in its early stages, our experiments show that a model trained with FDPL achieves higher quantitative performance (measured by PSNR) than the same model trained with MSE pixel loss. We also believe that the FDPL trained model produces better qualitative results, though it is difficult to tell if this is due to the increased PSNR or because of the perceptual emphasis of the loss function.

More specifically, the primary contribution of this work is the introduction and evaluation of Frequency Domain Perceptual Loss, a novel, model-agnostic perceptual loss function for super resolution.

RELATED WORKS

As a classical computer vision problem, SR has a wide body of related literature. Most related the current work are those that

CHI'20, April 25–30, 2020, Honolulu, HI, USA

ISBN 978-1-4503-6708-0/20/04…$15.00

DOI: https://doi.org/10.1145/3313831.XXXXXXX

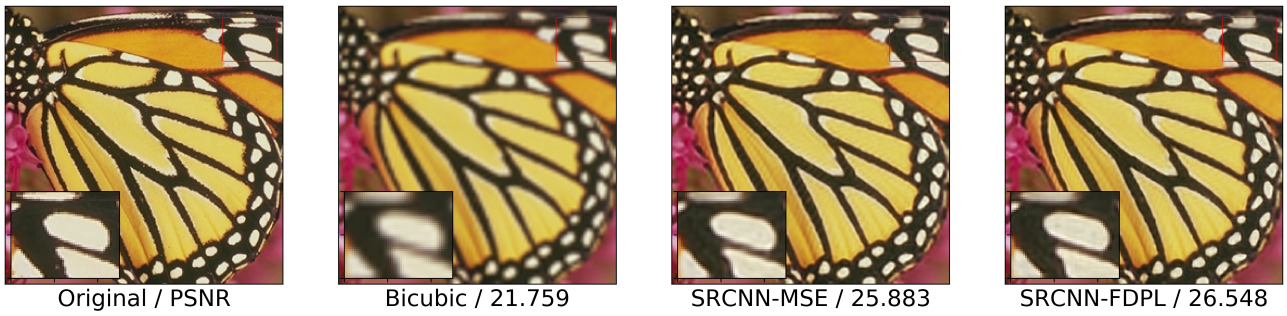

Figure 1. The image resolved by SRCNN trained on proposed FDPL surpasses the image resolved by SRCNN trained with pixel loss, in terms of PSNR (dB, higher is better) and perceptual quality. Upscale factor: 3.

use CNNs for the task, which mostly use a pixel loss during training. Pixel loss is instantiated in two ways in the literature: as either the mean Euclidean distance or mean absolute value distance between the pixels of the output and target images. Regardless of the particulars of the loss function used, all are computed in the pixel domain. Our work is different from all previous works in this respect, as we calculate loss in the frequency domain in order to use the appealing properties of having access to frequency coefficients.

Pixel Loss for Super Resolution

Most related to this work is the seminal work of Dong and Loy [3], which introduced Super Resolution CNN (SRCNN). This was the first published use of CNNs for SR and is still used as a bench mark in the most recent papers for the task. SRCNN uses a simple three layer architecture, each of which is grounded in a sub-task completed by previous non-CNN based approaches. SRCNN was trained by minimizing a per pixel-loss; the mean square error between the transformed image and the ground truth (or target) image (see Methodology section for details).

The current state of the art in SR is Fong's and Fowlkes' Predictive Filter Flow network (PFF) [6] and Wang et al.'s Enhanced Super Resolution Generative Adversarial Network (ESRGAN) [10] (described below). PFF uses a CNN to predict near optimal filter flow operations as the basis of their model. Another notable work is the Progressive Super Resolution (ProSR) network of [11], which achieves near state of the art results while running 5 times faster than competing methods. ProSR focuses on high factor SR ($8\times$) and operates by progressively up-sampling the input image by $2\times$ at each level of the model. ProSR also adopts a GAN (as ProGanSR) to achieve impressive photo-realistic results at high SR factors. Training of ProSR is done by a variation of curriculum learning (a form of easy to hard progressive training) using an ℓ_1 norm based loss function.

Perceptual Loss for Super Resolution

Departing from strictly pixel loss based training, Johnson *et al.* use a perceptual loss function for super resolution [5]. Their key intuition was that not all pixels are equally important for a human when judging the perceived quality of an image. As an extreme example they point out that a perfectly resolved picture that is simply shifted one pixel in any direction will

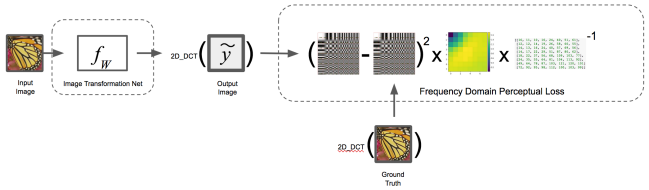

Figure 2. Visual overview of FDLP showing the squared difference in 8x8 DCT patches between target an output images being weighted by frequency importance, as determined by JPEG and relative difference frequencies computed over the training set.

suffer with a very poor pixel loss. To achieve a perceptual loss, they use a pre-trained VGG-16 loss network and compare an intermediate activation layer of that network with the output of their image transformation network, with loss measured as the mean squared error (MSE) between the two tensors.

In another seminal work on SR, Ledig et al. propose SRGAN, which combines the use of a GAN and a perceptual loss [7]. Their loss is formulated by measuring "content" reconstruction quality, which is also based on similarity to high level features of a VGG network. Again, this loss is measured in the pixel domain. ESRGAN is an improvement of SRGAN, that achieves excellent qualitative and state of the art quantitative results by addressing some of the shortcomings specific to SRGAN [10].

METHOD

We address the shortcomings of pixel-loss trained models by introducing the model agnostic Frequency Domain Perceptual Loss (FDPL). The key intuition of our method is that what makes a good SR model is the ability to do particularly well at restoring the high frequency components of a given image that are within the range of human perception (see Fig 1). Yet SR models trained on pixel loss target all frequencies equally: the low frequencies that are already given in the distorted image and the highest frequencies that are not perceptible to the human eye. If we can induce a model to focus on frequencies that are neither contained in the distorted image but are still important to human perception, then we will be able to achieve a higher perceptual quality, even if this doesn't translate to a higher PSNR (which is directly related to reducing pixel-

```
[[16, 11, 10, 16, 24, 40, 51, 61],
 [12, 12, 14, 19, 26, 58, 60, 55],
 [14, 13, 16, 24, 40, 57, 69, 56],
 [14, 17, 22, 29, 51, 87, 80, 62],
 [18, 22, 37, 56, 68, 109, 103, 77],
 [24, 35, 55, 64, 81, 104, 113, 92],
 [49, 64, 78, 87, 103, 121, 120, 101]
 [72, 92, 95, 98, 112, 100, 103, 99]]
```

Figure 3. An example JPEG Luminance channel quantization table, encoding assumptions of the relative importance of frequency components for human perception

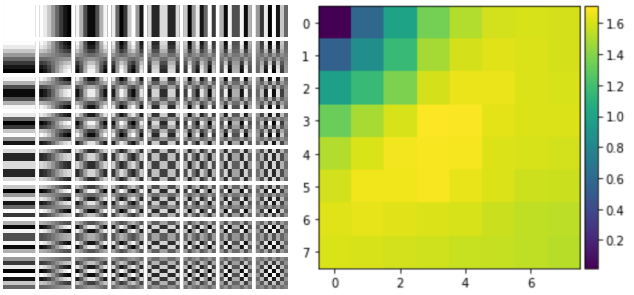

Figure 4. Left: a visualization of the DCT frequency components of an 8×8 image. Right: mean relative difference in frequency components between ground truth and distorted images in the training set.

loss MSE). This is the goal of FDPL. There are two main components in the FDPL, which we explain in turn prior to presenting the loss function: the Discrete Cosine Transform (DCT) and the JPEG luminance quantization table.

The DCT is a Fourier transform that, like the Discrete Fourier Transform (DFT), transforms a signal into the frequency domain. Both the DCT and DFT perform a change of basis from the time basis (or pixel basis in our case) to the frequency basis. The main difference between the DCT and the DFT is the former's use of cosine wave bases, versus the sine wave bases used by the latter. Formally the 2D DCT (denoted from here on simply as *DCT*) is defined as follows [1]:

$$\text{DCT}(\mathbf{X}_{jk}) = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} x_{mn} \cos\left(\pi \frac{j}{M}(n+\frac{1}{2})\right) \cos\left(\pi \frac{k}{N}(m+\frac{1}{2})\right)$$

where $\mathbf{X}$ is an input image (i.e a matrix), M and N are the height and width of the image (respectively), m and n are the usual row and column indices so that x_{mn} is the value at row m column n, $0 \leq j \leq M-1$, and $0 \leq k \leq N-1$.

The intuition behind the JPEG compression standard is an important component of our loss function. Briefly, JPEG operates by performing 2D DCT operations over all non-overlapping 8×8 patches in a given image, divides the resulting coefficients by a quantization table, rounds these results to induce sparsity, and then applies run length and Huffman encoding to obtain the final result [9]. JPEG is an effective compression standard because it operates by removing the frequencies from an image that minimally harm perceived quality [9]. Even with significant compression, perceived quality is minimally harmed. JPEG operates on the principal that higher frequencies are less perceptible to the human eye than are lower frequencies. Thus, to preserve image quality during compression, we should start with eliminating the highest frequency components (bottom right in left side Fig. 3) and continuing towards the lower frequency components (top left in Fig. 3) until the desired level of compression is reached. JPEG encodes this assumption in the values of its quantization tables. An example of a quantization table used on the Luminance channel (when the image is in YCbCr format) is shown in Figure 4. Here we see that the elements corresponding to higher frequencies have larger values, which will more quickly result in zeroed out DCT coefficients during compression following the rounding operation mentioned above.

In this paper we compute the 2D DCT function over 8×8 non-overlapping patches of the given image, unless stated otherwise. The output of this function is a matrix of the same dimensions as the input, where each element is a coefficient indicative of the strength of the corresponding basis element. In Figure 4 we can see that the frequency of the components increase as we move from from the top left to the bottom right. The resulting coefficient matrix can be interpreted as the amount of each frequency, that when added together, makes up the initial input image.

Loss

We introduce Frequency Domain Perceptual Loss (FDPL) as a new loss function with which to train super resolution image transformation neural networks. Our method is model agnostic, so long as the model is trained via back-propagation. With the components described above (DCT and JPEG's quantization table) we can now define FDPL as follows:

$$\text{FDPL}(\mathbf{y}, \hat{\mathbf{y}}) = \|\text{DCT}(\mathbf{y}) - \text{DCT}(\hat{\mathbf{y}})\|_2^2 \odot (\mathbf{1} \oslash \mathbf{q}) \odot \mathbf{d}$$

where $\odot$ and $\oslash$ are the Hammarand (i.e. element wise) multiplication and division operators (respectively), $\mathbf{q}$ is the quantization table (Fig. 3), and $\mathbf{d}$ is the mean relative difference matrix computed over the training set. Note that the difference matrix shown in Figure 4 supports our assertions that the low frequencies are given in the distorted image and that it is the medium-high frequencies (those across the main diagonal) that are lost in image down-sampling. Our loss thus induces the network to focus on these missing frequency components, according to their relative importance to human perception as encoded by $\mathbf{q}$. See Figure 2 for a visual overview of a system using FDPL.

EXPERIMENTS

We investigate the impact of FDPL on super resolution performance by a conducting two initial experiments, where the only independent variable is the loss function and all other variables are controlled. For these experiments, quantitative evaluation is done by measuring PSNR and mean structural similarity index (SSIM); the traditional metrics for measuring SR performance. Our main experiment compares the performance of the SRCNN model when trained with FDPL verses the MSE

pixel loss described in [3]. Our second experiment tests a variant of FDPL where the weights in $\mathbf{q}$ are transposed around the anti-diagonal (FDPL-AT from here on). The rational for this latter experiment is that if restoring higher frequency elements is important for SR, then we should test our assumption that JPEG represents a good model of human perception by flipping importance to the highest frequencies.

For our experiments we follow the basic protocol in [3]. We use the BSDS500 training (200 images) and test set (200 images) for our training set and the BSDS500 validation set (100 images) as our validation set. The training and target image pairs are prepared by taking 32×32 pixel patches with a stride of 13 over all of the images in the training and validation sets. This results in a training set of approximately 204800 image patches. To distort the input training images we down sample with bicubic interpolation by the SR factor. For all of our experiments we use an SR factor of 3, which means that the output image will have $3\times$ the resolution of the input image. Input images are then blurred with a Gaussian kernel with standard deviation of 1. Images are then up-sampled via bi-cubic interpolation by the SR factor as the sole pre-processing step. For simplicity, all experiments are done only on the Luminance channel of the YCbCr formatted images, as in [3, 5].

Training of the SRCNN models is done via standard Stochastic Gradient Descent (SGD) with a batch size of 128, a learning rate of 1×10^{-4} for the first two convolutional layers, and 1×10^{-5} for the last. Models are implemented using PyTorch [8] and trained on a single Nvidia GTX 1080 GPU for 15×10^6 back-propagation operations[1]. Each model takes approximately 3 days to train from scratch.

Results

While training SRCNN with MSE and FDPL, we monitor the average PSNR on the *Set5*[2] image set (as in [3]). We can see in Figure 5 that almost immediately SRCNN trained with FDPL (SRCNN-FDPL) overtakes the PSNR achieved by SRCNN trained with MSE (SRCNN-MSE). While in both cases average PSNR increases throughout the course of training, it is not smooth when trained with FDPL. This is an indication that while related to PSNR, FDPL is not optimizing this measure as directly as pixel domain MSE does. At the conclusion of training, we see that FDPL achieves a modest increase in average PSNR over MSE, giving empirical support to FDPL as a more suitable SR loss function.

More important to us than quantitative performance is the perceived quality of the super resolved images. In Figures 6 to 9 we present a comparison of the images produced by the competing loss functions. While we believe that the images coming from SSRN-FDPL are more visually pleasing, this remains to be evaluated via a formal user study. At this point we are also not sure if the higher perceived quality is a result of the perceptual qualities of FDPL, or because of the slightly higher PSNR it achieves; a question that requires further study.

[1] In [3] SRCNN is trained for 8×10^8 back-propagations. We found that we achieved good results for experimentation purposes much earlier.

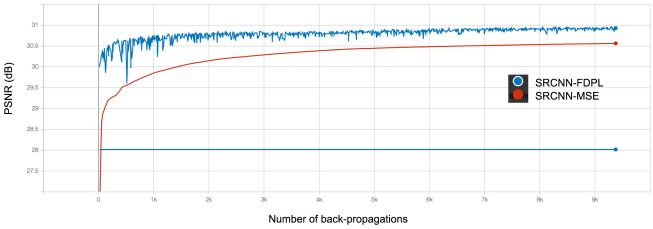

Figure 5. Average *Set5* PSNR increases over training progression for SRNCC-FDPL (blue) and SRCNN-MSE (red). PSNR after distorting image shown as constant.

Loss	*PSNR*	*SSIM*
Bi-cubic (baseline)	27.6340	0.8232
MSE	30.5917	0.8952
FDPL-AT	30.5946	0.8988
FDPL	**30.9399**	**0.9016**

Table 1. Average PSNR and SSIM achieved by SRCNN [3] when trained with standard pixel loss (MSE), FDPL with q transposed across the anti-diagonal (FDPL-AT), and the proposed FDPL.

CONCLUSION

In this paper we introduced the Frequency Domain Perceptual Loss function for training image transformation networks for super resolution. We conducted preliminary experiments evaluating the efficacy of this loss function, where we showed that its use in training SRCNN results in modest improvements in quantitative performance, as measured by PSNR and SSIM. We believe that the images resolved by the FDPL trained SRCNN are also of better perceptual (qualitative) quality. The main limitation of this result is that we can not say for certain that this qualitative improvement comes from the perceptually oriented FDPL loss function or simply as a result of the increased PSNR that it achieves.

The areas of future work are numerous and in addition to expanding the results of this paper, are required to answer the open question just mentioned. More specifically, before training state of the art SR networks with FDPL, we must first train SRCNN with more SR factors. To match the related SR literature, this will require training for $2\times, 4\times$, and $8\times$ upscale factors. We would also like to evaluate the mean relative difference matrix component in FDPL in order to provide support for its use. Following an evaluation done in [3], we will also train our models with a subset of the ImageNet1K dataset in order to have a more diverse training set. Future work should also explore the performance of FDPL in the image compression artefact reduction task, where models and methods are similar to SR, but where our loss function may be more complementary given the use of the DCT in many compression algorithms. A logical place to start would be with the work of Yu et al. [12], which is a slightly modified version of SRCNN for the artefact reduction task. Additionally, we will investigate the use of the Discrete Fourier Transform as an alternative to the DCT, since it is known to distribute frequency information more equally across basis coefficients as compared to DCT, which concentrates information in the lower frequency elements. Finally, in this paper we only used

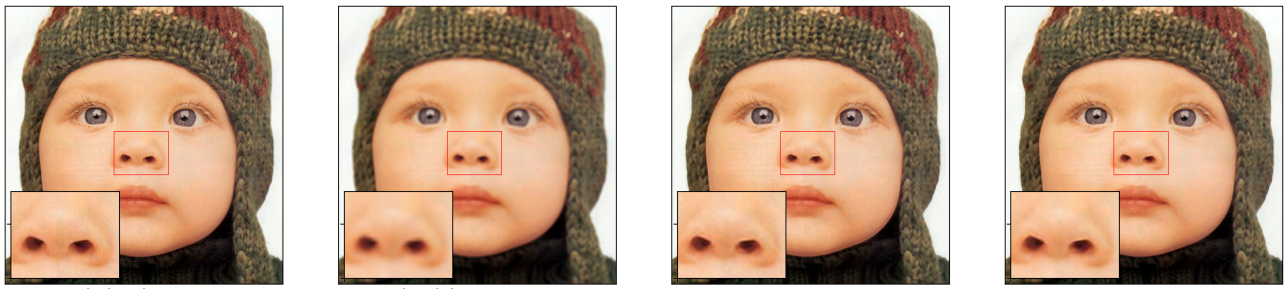

Figure 6. The *baby* image from *Set5*. Upscale factor: 3

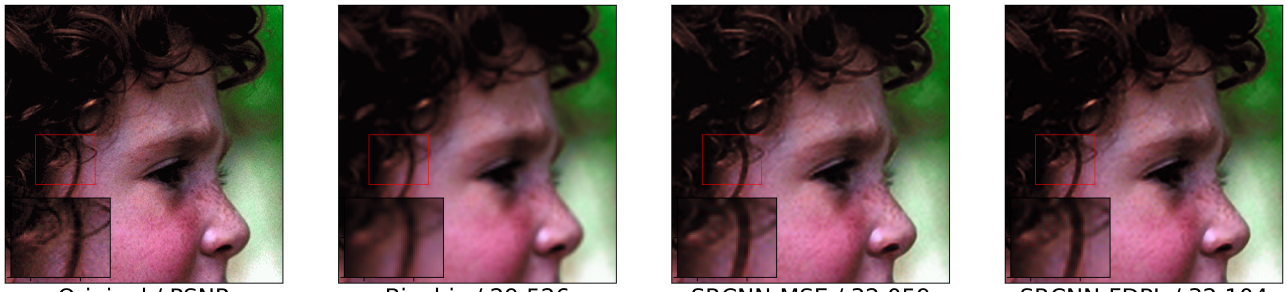

Figure 7. The *head* image from *Set5*. Upscale factor: 3

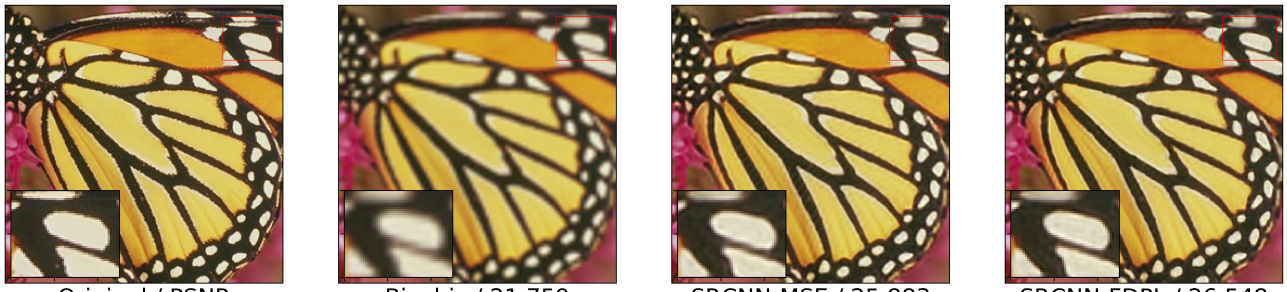

Figure 8. The *butterfly* image from *Set5*. Upscale factor: 3

Figure 9. The *bird* image from *Set5*. Upscale factor: 3

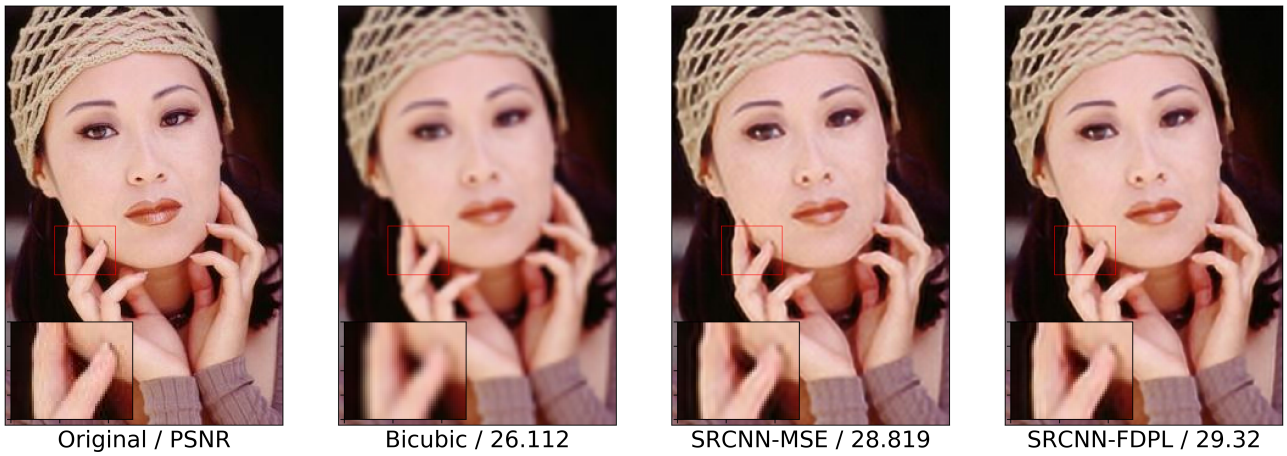

Figure 10. The *woman* image from *Set5*. Upscale factor: 3

a 5 image test set, which in the future should be expanded to include more standard benchmark image sets, such as *Set15*. Completing all of this future work will leave us well placed to expand our experiments to more models, including the current state of the art. If using FDPL to train these models results in even the modest performance increases seen in this paper, we will have achieved a new state of the art for the super resolution task.

REFERENCES

[1] Nasir Ahmed, T_ Natarajan, and Kamisetty R Rao. 1974. Discrete cosine transform. *IEEE transactions on Computers* 100, 1 (1974), 90–93.

[2] Marco Bevilacqua, Aline Roumy, Christine Guillemot, and Marie Line Alberi-Morel. 2012. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. (2012).

[3] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. 2015. Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence* 38, 2 (2015), 295–307.

[4] Katarzyna Janocha and Wojciech Marian Czarnecki. 2017. On loss functions for deep neural networks in classification. *arXiv preprint arXiv:1702.05659* (2017).

[5] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. 2016. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*. Springer, 694–711.

[6] Shu Kong and Charless Fowlkes. 2018. Image reconstruction with predictive filter flow. *arXiv preprint arXiv:1811.11482* (2018).

[7] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and others. 2017. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4681–4690.

[8] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, and others. 2019. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*. 8024–8035.

[9] Gregory K Wallace. 1992. The JPEG still picture compression standard. *IEEE transactions on consumer electronics* 38, 1 (1992), xviii–xxxiv.

[10] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. 2018b. Esrgan: Enhanced super-resolution generative adversarial networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 0–0.

[11] Yifan Wang, Federico Perazzi, Brian McWilliams, Alexander Sorkine-Hornung, Olga Sorkine-Hornung, and Christopher Schroers. 2018a. A fully progressive approach to single-image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 864–873.

[12] Ke Yu, Chao Dong, Chen Change Loy, and Xiaoou Tang. 2016. Deep convolution networks for compression artifacts reduction. *arXiv preprint arXiv:1608.02778* (2016).