NONLINEAR MULTI-SCALE SUPER-RESOLUTION USING DEEP LEARNING

Kenneth Tran[1*] *Ashkan Panahi*[2] *Aniruddha Adiga*[2*] *Wesam Sakla*[3] *Hamid Krim*[2*]

[1] Department of Computer Science, North Carolina State University, Raleigh, NC 27695
[2]Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, NC 27695
[3]Computational Engineering Division, Lawrence Livermore National Laboratory, Livermore, CA 94550

ABSTRACT

We propose a deep learning architecture capable of performing up to 8x single image super-resolution. Our architecture incorporates an adversarial component from the super-resolution generative adversarial networks (SRGANs) and a multi-scale learning component from the multiple scale super-resolution network (MSSRNet), which only together can recover smaller structures inherent in satellite images. To further enhance our performance, we integrate progressive growing and training to our network. This, aided by feed forwarding connections in the network to move along and enrich information from previous inputs, produces super-resolved images at scaling factors of 2, 4, and 8. To ensure and enhance the stability of GANs, we employ Wasserstein GANs (WGANs) during training. Experimentally, we find that our architecture can recover small objects in satellite images during super-resolution whereas previous methods cannot.

***Index Terms*—** super-resolution, remote sensing data, GANs, dilated convolutions

1. INTRODUCTION

Single image super-resolution (SISR) is the task of recovering a high resolution (HR) output from a low resolution (LR) input. Due to the improvement in hardware and availability of large data sets, SISR using deep learning is becoming increasingly popular in the computer vision community. Although [1, 2, 3, 4, 5] have led to remarkable results in 4x super-resolution of generic imagery, little work has been performed on remote sensing data. With the release of massive satellite imagery data sets such as SpaceNet [6] and the Functional Map of the World (fMoW) [7], it has become increasingly essential to explore how state-of-the-art SISR methods can be effectively applied to these data. With super-resolved satellite imagery, many remote sensing tasks such as detecting deforestation, or spotting undeclared nuclear power plants, would become more feasible.

A major difficulty in remote sensing applications is that different satellites capture images at varying temporal and spatial resolution. For example, WorldView-3 (WV3) captures images infrequently, with a re-visit period, or time elapsed between two successive views of the same area, of about 4.5 days [8], whereas PlanetScope (PS) has a re-visit period of less than a day [9]. However, WV3 has a spatial resolution of 30-cm/pixel, whereas PS has a spatial resolution of 3-m/pixel. Consequently, we are interested in the task of processing PS data based on learned models to artificially increase their spatial resolution, achieving high spatial and temporal resolution simultaneously. With this processed data, detection of small variations over time is possible, which is required for remote sensing applications such as those previously mentioned. The purpose of this work is to introduce a deep neural network architecture that leverages and improves on recent developments in super-resolution [2, 3] and training GANs [10, 11], resulting in a super-resolution method capable of capturing the finer details required for satellite imagery.

2. BACKGROUND AND RELATED WORKS

2.1. Super-Resolution with Deep Learning

In 2014, Dong et al. [1] introduced super-resolution convolutional neural networks (SRCNN), a method that closely followed the operations of traditional sparse-coding-based methods for super-resolution. Since then, several deep learning super-resolution methods have extended this idea. Yu and Porikli [12] applied GANs to perform 8x super-resolution on a dataset of faces using a standard pixel-wise loss function. Super-resolution GANs (SRGANs), proposed by Ledig et al. [2], applies both perceptual loss and GANs to super-resolution to gain a state-of-the-art performance over SRCNN. Perceptual loss is a dissimilarity metric computed by feeding the super-resolved image and the ground truth image independently into a pre-trained network and comparing their respective outputs with a standard metric such as the l_2 distance. Also, a new architecture for super-resolution called SRResNet that uses residuals connections is introduced. Dahl et al. combined a conditioning network and a prior network to develop pixel recursive super-resolution [4]. The conditioning network makes a prediction on the current pixel by using information from the LR image. The prior network follows from PixelCNN [13] and predicts the next pixel based on every other pixel that has already been generated. These two predictions are combined to produce the final

*This work was in part supported by DOE - National Nuclear Security Administration through CNEC-NCSU under Award DE-NA0002576.

value for the current pixel. This is repeated until the entire HR image is generated. Shi et al. proposed the multiple scale super-resolution network (MSSRNet) [3] which uses dilated convolution inception modules. In [14], a discrete dilated convolution $*_l$ with dilation factor l is defined as:

$$(F *_l k)[\mathbf{p}] = \sigma\left(\sum_{\mathbf{s}+l\mathbf{t}=p} F(\mathbf{s})k(\mathbf{t})\right), \quad (1)$$

where $F : \mathbb{Z}^2 \to \mathbb{R}$ is a discrete signal, $\Omega_r = [-r, r]^2 \cap \mathbb{Z}^2$, $k : \Omega_r \to \mathbb{R}$ is a discrete filter of size $(2r+1)^2$, and σ is a nonlinear activation function. It is noted that a traditional convolution operation is a dilated convolution with $l = 1$. These modules take multiple dilated convolution operations with different values of l on the same input to produce multiple feature maps. These feature maps are then concatenated before being fed into the next layer. With these modules, their architecture concurrently learns features that promote scale-invariance.

2.2. Generative Adversarial Networks

In GANs [15], G and D are two neural networks simultaneously trained with opposing objectives, shown in Eq. (2).

$$\min_G \max_D \mathbb{E}_{\tilde{x}\sim P_G}[log(1 - D(\tilde{x}))] + \mathbb{E}_{x\sim P_R}[log(D(x))] \quad (2)$$

The generator $G(\cdot)$ creates tentative samples from random noise. The objective of $G(\cdot)$ is to output samples $\tilde{x}$ with the same statistics as x_R. To adapt the generator to super-resolution, SRGANs inputs a downsampled x and super-resolves that image to produce $\tilde{x}$. The discriminator $D(\cdot)$ aims to correctly differentiate between $\tilde{x}$ and x_R. By alternating updates of the parameters of each network, both networks incrementally improve in performance. In [10], Arjovsky et al. argue that the original GANs formulation exhibits instability during training, and use of the Wasserstein distance to derive the loss function alleviates this problem. This resulted in WGANs, which has demonstrated improved performance and more stable training. Additionally, Karras et al. introduced progressive growing of GANs [11] to guide the generation of images. The idea behind progressive growing is that intermediate layers of the generator should produce tentative images at lower resolution, while the discriminator should be learning to distinguish between these lower resolution images and the downsampled training data. To achieve this, they propose an incremental addition of layers to the generator and discriminator while increasing the resolution of the output. At each increment, both networks are optimized. Using progressive growing ensures that each successive set of layers in the generator provides meaningful intermediate outputs, whereas traditional methods just train the whole network.

Our contribution: The purpose of this work is to demonstrate that SISR can be applied to upscale satellite images by a factor of 8, which is a step towards performing the 10x super-resolution required for the disparity in spatial resolution between the PS and WV3 satellites. Our experiments demonstrate the shortcomings of SRGANs and MSSRNet in that they cannot recover smaller objects in satellite imagery. We propose an amalgamation of the two architectures by replacing the convolutions in SRGANs with dilated convolution modules. To further enhance the quality of our images, we follow progressive growing of GANs by splitting our super-resolution task into three separate upscaling stages. Lastly, we carry forward all previous inputs and concatenate them to the input of future stages. Theoretically, this augments the information space of successive information inputs. Intuitively, this gives future stages nonlinear and rich information that may have been lost in previous layers. To secure improved training stability for our increased upscaling factor, we opted to use WGANs over GANs for its better numerical behavior. With WGANs, we also propose using a $\tanh(\cdot)$ activation on the discriminator output to make the adversarial loss scale properly with the content loss.

3. A NEW APPROACH TO SISR

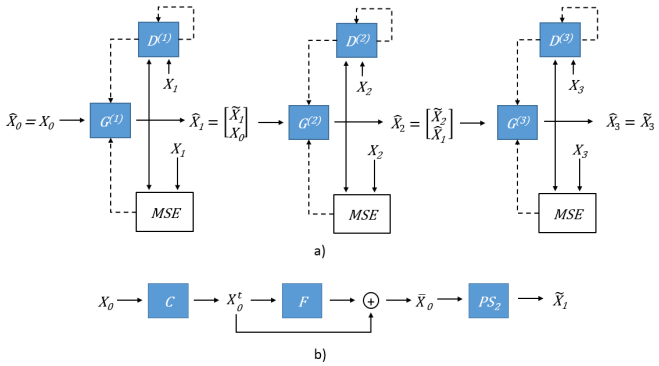

Fig. 1. **a)** A diagram of the super-resolution system. We denote $G^{(i)}(\hat{X}_{i-1})$ as $\tilde{X}_i$. Dotted lines: backpropagation, Solid lines: forward pass. **b)** $G^{(1)}(\cdot)$ decomposed into $C(\cdot)$, $F(\cdot)$, and $PS_2(\cdot)$.

3.1. Generator and Discriminator Architecture

While most previous methods perform 4x super-resolution, we propose a deep learning architecture that is better suited for 8x super-resolution on satellite imagery. Our proposed framework is shown at a high level in Fig. 1a. All $G^{(i)}$ blocks are functions that super-resolve inputs by a factor of 2. All $D^{(i)}$ blocks seek to differentiate between the original higher resolution image and the super-resolved image. These blocks are modeled by neural networks. The MSE blocks represent the mean squared error between two inputs. X_u denotes the original image data of differing resolution determined by u. $\hat{X}_v$ is the concatenation of all previous inputs, i.e. $\hat{X}_1 = [\tilde{X}_1, X_0]$ and $\hat{X}_2 = [\tilde{X}_2, X_1]$.

Inspired by progressive training, we structured our generator training into three different stages, each upsampling by a factor of 2. The first stage parameters are trained to optimality and kept constant while training the second stage. Then, the first and second stage parameters are kept constant

to train the third stage. In Fig. 1b, we decomposed $G^{(1)}(\cdot)$ into multiple operations denoted by $C(\cdot)$, $F(\cdot)$, and $PS_2(\cdot)$. $C(\cdot)$ is a single convolutional layer given by Eq. (1) with $l = 1$ that acts as regularization for X_0 and outputs X_0^t. $F(\cdot)$ is a series of dilated convolutions modules with residual connections performed on X_0^t. A dilated convolution module is the concatenation of a series of dilated convolutions, defined by Eq. (1), with different l, followed by an additional convolution operation. We use $F(\cdot)$ to learn information about X_0^t necessary for super-resolution. Subsequently, X_0^t and the output of $F(\cdot)$ are summed to generate $\overline{X}_0$. From the spectral view point, LR images lack high-frequency components, corresponding to lower spatial sampling distance in a high-resolution image. The low-frequency components are further distorted due to the aliasing effect. Accordingly, the mechanism of deep CNNs is naturally appealing for the purpose of super-resolution as illustrated by the spectral study of the signal evolution over the forward path: the nonlinear function in Eq. (1) creates high-frequency components, which are further processed by a collection of filters in each layer, re-identifying the distorted spectrum by the aliasing effect. Employing dilated filters further provides a means of controlling the filters band width, pertaining to the multi-scale nature of our architecture. Unlike conventional signal processing approaches, the process of backpropagation in deep learning enables us to adjust the shape and spectral location of the filters in a supervised manner, based on the training data, leading to a highly adaptive multi-scale approach.

$PS_2(\cdot)$ is the pixel shuffler [5] operation, a learned interpolator, with an upscaling factor of 2. It is defined by:

$$Y[i,j] = \sum_{a\in\{0,1\},b\in\{0,1\}} (F *_1 k^{(ab)}) \left[\frac{i+a}{2}, \frac{j+b}{2}\right], \quad (3)$$

where Y is a discrete signal, and hence $Y[o,p] = 0$ for $o, p \notin \mathbb{Z}^+$. As noted in [16] and [17], feed forwarding the outputs of previous layers is effective because it increases accessible information and aides in mitigating vanishing gradients in deep networks. By concatenating successive stage inputs, our network achieves a dense set of nonlinear features. This ensures the persistent presence of nonlinearities at all scales, and therefore better fits all SR detailed features.

The discriminator $D(\cdot)$ clearly plays a critical role in the adaptivity of each scale-specific stage, and the desirable smoothness properties were investigated in [10]. Specifically, a Wasserstein-metric-based criterion was shown to be more adapted for GANs. This also yielded a markedly improved performance. Our proposed generator, shown in Fig. 2, utilizes one separate discriminator per stage, each including seven associated convolution layers and a fully connected layer.

In contrast to SRGANs, we adopt a more flexible loss function allowing for the use of multiple content losses. The overall loss is given by:

$$l^{SR} = \sum_{k=1}^{n} \alpha_k l_{C_k}^{SR} + \beta l_{adv}^{SR}, \quad (4)$$

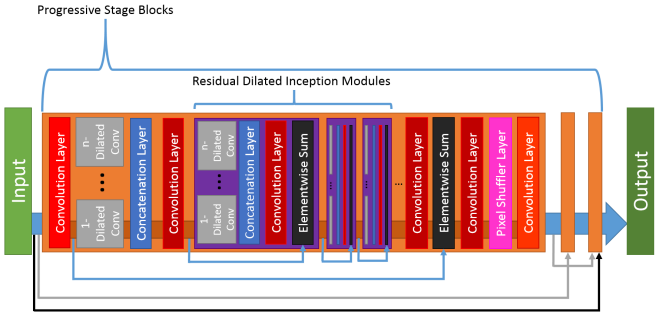

Fig. 2. The generator network for our proposed method.

where $l_{C_i}^{SR}$ denotes a content loss (MSE or perceptual loss) and l_{adv}^{SR} refers to the adversarial loss given by the discriminator. In our case, we use $n = 2$, with $l_{C_1}^{SR}$ being the perceptual loss [19] adapted from SRGANs and $l_{C_2}^{SR}$ being the MSE loss. The perceptual loss is computed by:

$$l_{C_1}^{SR} = \frac{1}{W_{q,r}H_{q,r}} \sum_{a=1}^{W_{q,r}} \sum_{b=1}^{H_{q,r}} (\phi_{q,r}(X_i)_{a,b} - \phi_{i,j}(G^{(i)}(\hat{X}_{i-1}))_{a,b})^2, \quad (5)$$

where $\phi_{q,r}$ is the q^{th} convolution after the r^{th} max-pooling layer in a pre-trained VGG-19 [20] network. In our case, we used $q = 0$ and $r = 5$. $W_{q,r}$ and $H_{q,r}$ represent the width and height of the feature maps.

WGANs is used for the adversarial loss. In our experiments, and supported by [21], we found that using the gradient penalty in [22] led to poorer super-resolved images. Instead, we follow the original WGANs implementation and clip the weights of our discriminator to [-0.01, 0.01] to enforce the 1-Lipschitz constraint. The WGANs loss is given by:

$$l_{WGANs} = \mathbb{E}_{\tilde{x}\sim\mathbb{P}_g}[D(\tilde{x})] - \mathbb{E}_{x\sim\mathbb{P}_r}[D(x)], \quad (6)$$

where $\mathbb{P}_g$ is the generated image distribution, $\mathbb{P}_r$ is the HR image distribution, and $D(\cdot)$ is a 1-Lipschitz function.

A problem with directly applying WGANs to SRGANs is that the discriminator is not bounded to [0,1]. To alleviate the potential of the adversarial loss overwhelming the content loss, we use the $\tanh(\cdot)$ activation on the discriminator output and thus bound the former loss. The new formulation for the super-resolution adversarial loss thus becomes:

$$l_{adv}^{SR} = \mathbb{E}_{\tilde{X}\sim\mathbb{P}_g}[\sigma(D(\tilde{X}))] - \mathbb{E}_{X\sim\mathbb{P}_r}[\sigma(D(X))], \quad (7)$$

where $\sigma(\cdot)$ is $\tanh(\cdot)$.

The loss we use for our training is given by $l^{SR} = \alpha_1 l_{C_1}^{SR} + \alpha_2 l_{C_2}^{SR} + \beta l_{adv}^{SR}$, where α_1, α_2, and β are tuned parameters dependent on the data set.

4. EXPERIMENTS

In our experiments, we compare our proposed method to SRGANs and MSSRNet. All methods were implemented in Ten-

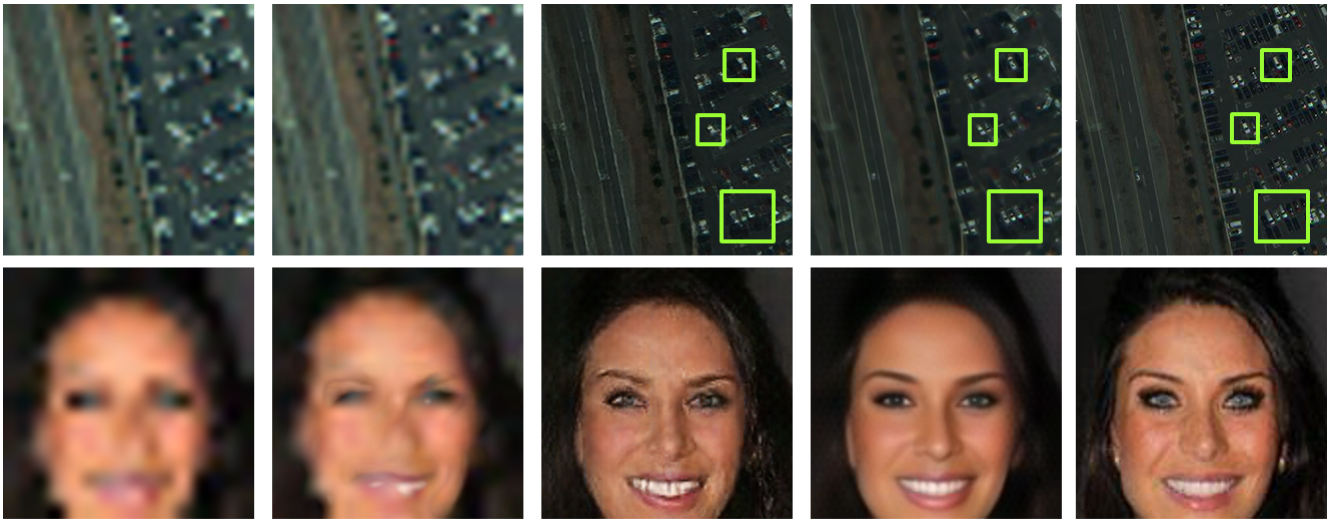

Fig. 3. Sample images from both WV3 images (top) and CelebA images (bottom). Perceptually, we observe that our method better captures the smaller structures such as the cars, indicated by boxes, in WV3 data and teeth/eyes in CelebA data. See [18] for source code and samples.

sorLayer, a wrapper for TensorFlow [23]. We slightly modified the hyper-parameters that weight the effect of the generator loss to better match the remote sensing content loss.

4.1. Datasets and Evaluation Metrics

We use the benchmark dataset CelebA [24], a collection of celebrity faces, as the initial test for our method, demonstrating its effectiveness on generic imagery. We also use HR remote sensing data collected from Digital Globe's WorldView-3 (WV3) satellites (partially taken from SpaceNet [6]) to showcase our intended application. The HR training data are downsampled to the appropriate sizes to train each stage of the generator. The peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) [25], two metrics used for image reconstruction quality, are computed to evaluate the performance of each method on both data sets. Since SRGANs and MSSRNet have already been shown to outperform bi-cubic interpolations and SRCNN [2, 3], we omit them from our comparison.

4.2. Training Details and Results

All of the networks are trained using an NVIDIA Tesla P100 GPU using a training split of a given data set. For CelebA, we take a random subset of 50,000 images for training. We start with 128x128 HR images for the face data set and downsample them to 16x16 for our training set. We have also experimented with remote sensing WV3 data using 10,000 images. Starting with 320x320 HR image patches, we downsample them to 40x40 images for training. As previously mentioned, we use a VGG-19 network previously trained on ImageNet for the perceptual loss. We use $\alpha_1 = 10^{-8}$, $\alpha_2 = .8$, and $\beta = 10^{-4}$ for CelebA data and $\alpha_1 = 3\text{x}10^{-8}$, $\alpha_2 = .6$, and $\beta = 3\text{x}10^{-4}$ for WV3 data. We show the PSNR and SSIM in Table 1. Moreover, sample results from our experiments are presented in Fig. 3. Our method achieved the highest PSNR and SSIM for both data sets. Visually, our method better recovers smaller objects, such as vehicles in satellite images and eyes/teeth in faces.

Method	CelebA		WorldView-3	
	PSNR	SSIM	PSNR	SSIM
SRGANs	22.612	0.6974	22.744	0.4332
MSSRNet	19.522	0.4625	19.243	0.2445
Our Method	**23.113**	**0.7189**	**22.931**	**0.4564**

Table 1. Peak signal-to-noise ratio and structural similarity of methods using 20,000 CelebA samples and 3,000 WV3 samples.

5. DISCUSSION

When downsampling satellite images by 8x, humans can still identify buildings and larger objects, but smaller objects completely lose their structure. However, given the correct context, such as the place appearing to be a parking lot, humans can infer that a few pixels should form a car. Prior SR work has been primarily successful at the higher scale object level, but limited at smaller scale object recovery. Our proposed architecture has demonstrated the ability to produce small objects such as cars given ambiguous pixels, showing characteristics that may be similar to human contextual information. Future work will focus on further studying the role that intermediate feature maps play in providing context for HR details. Other research questions we will address include the quantification of the data quality propagation, so we can selectively modulate less important stages to reduce the number of parameters and improve computational efficiency.

6. REFERENCES

[1] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang, "Learning a deep convolutional network for image super-resolution," in *European conference on computer vision*. Springer, 2014, pp. 184–199.

[2] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew P Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al., "Photo-realistic single image super-resolution using a generative adversarial network.," .

[3] Wuzhen Shi, Feng Jiang, and Debin Zhao, "Single image super-resolution with dilated convolution based multi-scale information learning inception module," in *IEEE International Conference on Image Processing (ICIP)*. IEEE, 2017, pp. 977–981.

[4] Ryan Dahl, Mohammad Norouzi, and Jonathon Shlens, "Pixel recursive super resolution," in *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[5] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 1874–1883.

[6] Adam Van Etten, Dave Lindenbaum, and Todd M Bacastow, "Spacenet: A remote sensing dataset and challenge series," *arXiv preprint arXiv:1807.01232*, 2018.

[7] Gordon Christie, Neil Fendley, James Wilson, and Ryan Mukherjee, "Functional map of the world," *arXiv preprint arXiv:1711.07846*, 2017.

[8] Digital Globe, Inc., *DG2017 WorldView-3 DS*, 2017.

[9] Planet Labs, Inc., *Planet Combined Imagery Product Specs*, 2018.

[10] Martin Arjovsky, Soumith Chintala, and Léon Bottou, "Wasserstein generative adversarial networks," in *International Conference on Machine Learning (ICML)*, 2017, pp. 214–223.

[11] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," 2018.

[12] Xin Yu and Fatih Porikli, "Ultra-resolving face images by discriminative generative networks," in *European Conference on Computer Vision*. Springer, 2016, pp. 318–333.

[13] Aaron van den Oord, Nal Kalchbrenner, Lasse Espeholt, Oriol Vinyals, Alex Graves, et al., "Conditional image generation with pixelcnn decoders," in *Advances in Neural Information Processing Systems*, 2016, pp. 4790–4798.

[14] Fisher Yu and Vladlen Koltun, "Multi-scale context aggregation by dilated convolutions," *arXiv preprint arXiv:1511.07122*, 2015.

[15] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial nets," in *Advances in neural information processing systems (NIPS)*, 2014, pp. 2672–2680.

[16] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger, "Densely connected convolutional networks.," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[17] Tsung-Yi Lin, Piotr Dollár, Ross B Girshick, Kaiming He, Bharath Hariharan, and Serge J Belongie, "Feature pyramid networks for object detection.," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[18] Kenneth Tran, "Source code and sample results," https://github.com/ktran6/Nonlinear-Multi-Scale-Super-Resolution-Using-Deep-Learning.

[19] Justin Johnson, Alexandre Alahi, and Li Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *European Conference on Computer Vision*. Springer, 2016, pp. 694–711.

[20] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[21] Zhimin Chen and Yuguang Tong, "Face super-resolution through wasserstein gans," *arXiv preprint arXiv:1705.02438*, 2017.

[22] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville, "Improved training of wasserstein gans," in *Advances in Neural Information Processing Systems (NIPS)*, 2017, pp. 5767–5777.

[23] Hao Dong, Akara Supratak, Luo Mai, Fangde Liu, Axel Oehmichen, Simiao Yu, and Yike Guo, "TensorLayer: A Versatile Library for Efficient Deep Learning Development," *ACM Multimedia*, 2017.

[24] Shuo Yang, Ping Luo, Chen-Change Loy, and Xiaoou Tang, "From facial parts responses to face detection: A deep learning approach," in *IEEE International Conference on Computer Vision (CVPR)*, 2015, pp. 3676–3684.

[25] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.