Deep Wavelet Prediction for Image Super-resolution

Tiantong Guo, Hojjat Seyed Mousavi, Tiep Huu Vu, Vishal Monga
School of Electrical Engineering and Computer Science
The Pennsylvania State University, State College, PA, 16803
http://signal.ee.psu.edu

Abstract

Recent advances have seen a surge of deep learning approaches for image super-resolution. Invariably, a network, e.g. a deep convolutional neural network (CNN) or auto-encoder is trained to learn the relationship between low and high-resolution image patches. Recognizing that a wavelet transform provides a "coarse" as well as "detail" separation of image content, we design a deep CNN to predict the "missing details" of wavelet coefficients of the low-resolution images to obtain the Super-Resolution (SR) results, which we name Deep Wavelet Super-Resolution (DWSR). Out network is trained in the wavelet domain with four input and output channels respectively. The input comprises of 4 sub-bands of the low-resolution wavelet coefficients and outputs are residuals (missing details) of 4 sub-bands of high-resolution wavelet coefficients. Wavelet coefficients and wavelet residuals are used as input and outputs of our network to further enhance the sparsity of activation maps. A key benefit of such a design is that it greatly reduces the training burden of learning the network that reconstructs low frequency details. The output prediction is added to the input to form the final SR wavelet coefficients. Then the inverse 2d discrete wavelet transformation is applied to transform the predicted details and generate the SR results. We show that DWSR is computationally simpler and yet produces competitive and often better results than state-of-the-art alternatives.

1. Introduction

In image processing, reconstructing High-Resolution (HR) image from its corresponding Low-Resolution (LR) image is known as Super-Resolution (SR). The methods accomplishing this task are usually classified into two categories: multi-frame super-resolution and single image super-resolution (SISR). In multi-frame super-resolution, multiple LR images that are captured from the same scene are combined to generate the corresponding HR image [1, 2]. In SISR, it is very common to utilize examples from the historic data and form dictionaries of LR and HR image patches [3, 4]. These dictionaries are then used to transform each LR patch to the HR domain. For instance, [5, 6, 7, 8, 9] explored the similarity of self-examples, while others mapped the LR to HR patches with use of external samples [10, 11, 12, 13, 14, 15, 16, 17].

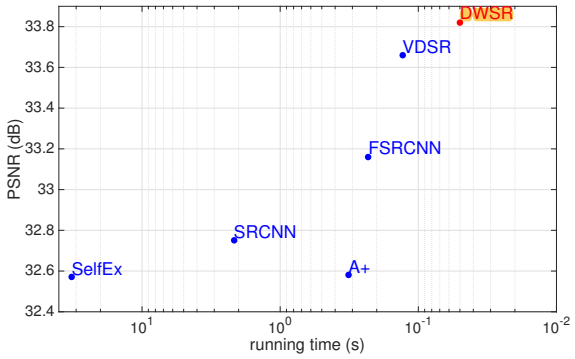

Figure 1: DWSR and other state-of-the-art methods reported PSNR with scale factor of 3 on Set5. For experimental setup see Section 4.4.

In this paper, we address the problem of single image super resolution, and we propose to apply super resolution in the wavelet domain for the reasons that we will justify later. Wavelet coefficients prediction for super-resolution has been applied successfully to multi-frames SR. For instance, [18, 19, 20, 21] used multi-frames images to interpolate the missing details in the wavelet sub-bands to enhance the resolution. Several different interpolation methods for wavelet coefficients in SISR were studied as well. [22] used straightforward bicubic interpolation to enlarge the wavelet sub-bands to produce SR results in spatial domain. [23] explored interlaced sampling structure in the low-resolution data for wavelet coefficients interpolation. [24] formed a minimization problem to learn the suitable wavelet interpolation with a smooth prior. Since the detailed wavelet sub-bands are often sparse, it is suitable to apply sparse coding methods to estimate detailed

wavelet coefficients and can significantly refine image details. Methods [25, 26, 27] used different interpolations related to sparse coding. Other attempts [28, 29] utilize Markov chains and [30] used nearest neighbor to interpolate wavelet coefficients. However, due to limited training and straightforward prediction procedures, these methods are not powerful enough to process general input images and fail to deliver state-of-the-art SR results, especially compared to more recent deep learning based methods for super resolution.

Deep learning promotes the design of large scale networks [31, 32, 33] for a variety of problems including SR. To this end, deep neural networks were applied to super resolution task. Among the first deep learning based super resolution methods, Dong *et al.* [34] trained a deep convolution neural network (SRCNN) to accomplish the image super-resolution task. In this work, the training set comprises of example LR inputs and their corresponding HR output images which were fed as training data to the SRCNN network. Combined with sparse coding methods, [35] proposed a coupled network structure utilizing middle layer representations for generating SR results which reduced training and testing time. In different approaches, Cui *et al.* [9] proposed a cascade network to gradually upscale LR images after each layer, while [17] trained a high complexity convolutional auto-encoder called Deep Joint Super Resolution (DJSR) to obtain the SR results. Self examples of images were explored in [36] where training sets exploit self-example similarity, which leads to enhanced results. However, similar to SRCNN, DJSR suffers from expensive computation in training and processing to generate the SR images.

Recently, residual net [37] has shown great ability at reducing training time and faster convergence rate. Based on this idea, a Very Deep Super-Resolution (VDSR) [38] method is proposed which emphasizes on reconstructing the residuals (differences) between LR and HR images rather than putting too much effort on reconstructing low frequency details of HR images. VDSR uses 20 convolutional layers producing state-of-the-art results in super resolution and takes significantly shorter training time for convergence; however, VDSR is massively parameterized with these 20 layers.

Motivations: Most of the deep learning based image super resolution methods work on spatial domain data and aim to reconstruct pixel values as the output of network. In this work we explore the advantages of exploiting transform domain data in the SR task especially for capturing more structural information in the images to avoid artifacts. In addition to this and motivated by promising performance of VDSR and residual nets in super resolution task, we propose our Deep Wavelet network for super resolution (DWSR). Residual networks benefit from sparsity of input and output, and the fact that learning networks with sparse activations is much easier and more robust. This motivates us to exploit spatial wavelet coefficients which are naturally sparse. More importantly, using residuals (differences) of wavelet coefficients as training data pairs further enhances the sparsity of training data resulting in more efficient learning of filters and activations. In other words, using wavelet coefficients encourages activation sparsity in middle layers as well as output layer. Consequently, residuals for wavelet coefficients themselves become sparser and therefore easier for the network to learn. In addition to this, wavelet coefficients decompose the image into sub-bands which provide structural information depending on the types of wavelets used. For example, Haar wavelets provide vertical, horizontal and diagonal edges in wavelet sub-bands which can be used to infer more structural information about the image. Essentially our network uses complementary structural information from other sub-bands to predict the desired high-resolution structure in each sub-band.

The **main contributions** of this paper are the following: 1) To the best of our knowledge, the proposed DWSR is the first approach to combine the complementarity of information (into low and high frequency sub-bands) in the wavelet domain with a deep CNN. Specifically, wavelets promote sparsity and also provide structural information about the image. 2) In addition to a wavelet prediction network, we built on top of residual networks which fit well to the wavelet coefficients due to their sparsity promoting nature and further enhancing it by inferring residuals. 3) Our network has multiple input and output channels which allows to learn different structures at different levels of the image. This complementary structural information in wavelet coefficients helps in better reconstruction of SR results with less artifacts. Extensive experimental results validate that our approach produces less artifacts around edges and outperforms many state-of-the-art methods.

2. 2D Discrete Wavelet Transformation (2dDWT)

To perform a 1D Discrete Wavelet Transformation, a signal $x[n] \in \mathbb{R}^N$ is first passed through a half band high-pass filter $G_H[n]$ and a low-pass filter $G_L[n]$, which are defined as (for Haar ("db1") wavelet):

$$G_H[n] = \begin{cases} 1, & n = 0 \\ -1, & n = 1 \\ 0, & \text{otherwise} \end{cases}, G_L[n] = \begin{cases} 1, & n = 0, 1 \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

After filtering, half of the samples can be eliminated according to the Nyquist rule, since the signal now has a frequency bandwidth of $\pi/2$ radians instead of π.

Any digital image x can be viewed as a 2D signal with index $[n, m]$ where $x[n, m]$ is the pixel value located at nth

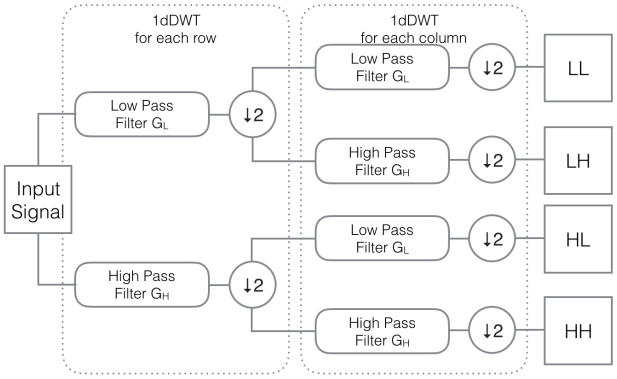

Figure 2: The procedure of 1-level 2dDWT decomposition.

column and mth row. The 2D signal $x[n, m]$ can be treated as 1D signals among the rows $x[n, :]$ at a given nth column and among the columns $x[:, m]$ at a given mth row. A 1-level 2D wavelet transform of an image can be captured by following the procedure in Figure 2 along rows and columns, respectively. As mentioned earlier, we are using Haar kernels in this work.

An example of 1-level 2dDWT decomposition with Haar kernels is shown in Figure 3. The right part of Figure 3 is the notation of each sub-band of wavelet coefficients. It is clear that the 2dDWT captures the image details in four sub-bands: average (LL), vertical(HL), horizontal(LH) and diagonal(HH) information, which are corresponding to each wavelet sub-bands coefficients. Note that after 2dDWT decomposition, the combination of four sub-bands always have the same dimension as the original input image.

The 2d Inverse DWT (2dIDWT) can trace back the 2dDWT procedure by inverting the steps in Figure 2. This allows the prediction of wavelet coefficients to generate SR results. Detailed wavelet decomposition introduction can be found in [39].

3. Proposed Method: Deep Wavelet Prediction for Super-resolution (DWSR)

The SR can be viewed as the problem of restoring the details of the image given an input LR image. This viewpoint can be combined with wavelet decomposition. As shown in Figure 3, if we treat the input image as an LL output of 1-level 2dDWT, predicting the HL, LH and HH sub-bands of the 2dDWT will give us the missing details of the LL image. Then one can use 2dIDWT to gather the predicted details and generate the SR results. With Haar

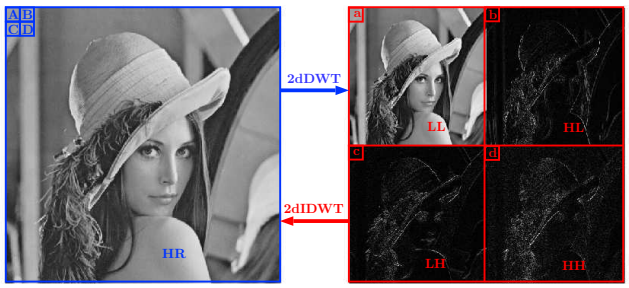

Figure 3: The 2dDWT and 2dIDWT. A, B, C, D are four example pixels located in a 2×2 grid at the top left corner of HR image. a, b, c, d are four pixels from the top left corner of four sub-bands correspondingly.

wavelet, the coefficients of 2dIDWT can be computed as:

$$\begin{cases} A = a + b + c + d \\ B = a - b + c - d \\ C = a + b - c - d \\ D = a - b - c + d \end{cases} \tag{2}$$

where A, B, C, D and a, b, c, d represent the pixel values from corresponding image/sub-bands.

Therefore, with the help of wavelet transformation, the SR problem becomes a wavelet coefficients prediction problem. In this paper, we propose a new deep learning based method to predict details of wavelet sub-bands from the input LR image. To the best of our knowledge, DWSR is the first deep learning based wavelet SR method.

3.1. Network Structure

The structure of the proposed network is illustrated in Figure 4. The proposed network has a deep structure similar to the residual network [37] with two input and output layers with 4 channels. While most of deep learning based SR methods have only one channel for input and output, our network takes four input channels into consideration and produces four corresponding channels at the output. There are 64 filters of size $4 \times 3 \times 3$ in the first layer and 4 filters of size $64 \times 3 \times 3$ in the last layer. In the middle part of the network, the network has N same-sized hidden layers with $64 \times 3 \times 3 \times 64$ filters each. The output of each layer, except the output layer, is fed into ReLU activation function to generate a nonlinear activation map.

Usually, the CNN based SR methods only take valid regions into consideration while feeding forward the inputs. For example, in SRCNN [34], the network has three layers with filter size of 9×9, 1×1 then 5×5, from which we can compute the cropped out information width, which is $(9 + 1 + 5 - 3) = 12$ pixels. During the training process, SRCNN takes in sub-images of size 33×33, but only produce outputs of size 21×21. This procedure is

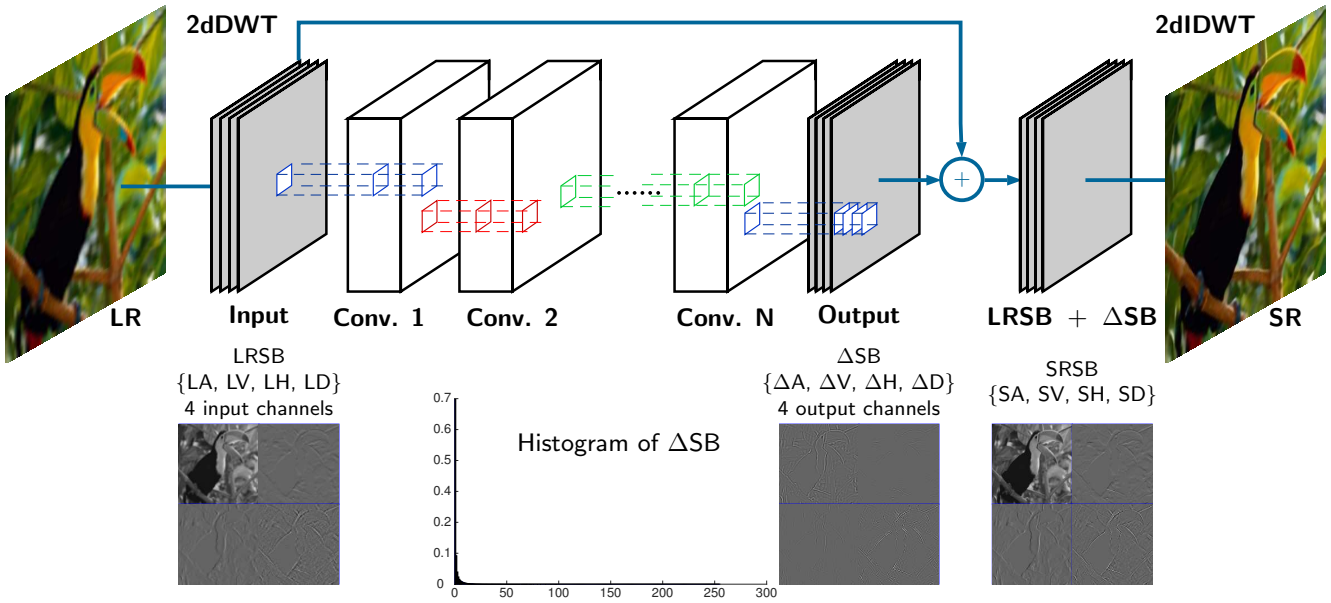

Figure 4: Wavelet prediction for SR network structure: there are input layers which takes four channels and output layers produce four channels. The network body has repeated N same-sized layers with ReLU activation functions. One example of the input LRSB and network output ΔSB are plotted. The histogram of all coefficients in ΔSB is drawn to illustrate the sparsity of the outputs.

unfavorable in our deep model since the final output could be too small to contain any useful information.

To solve this problem, we use zero padding at each layer to keep the outputs having the same sizes as the inputs. In this manner, we can produce the same size final outputs as the inputs. Later the experiments shows that with the special wavelet sparsity, the padding will not affect the quality of the SR results.

3.2. Training Procedure

To train the network, the low-resolution training images are enlarged by bicubic interpolation with the original downscale factor. Then the enlarged LR images are passed through the 2dDWT with Haar wavelet to produce four LR wavelet Sub-Bands (LRSB) which is denoted as:

$$\text{LRSB} = \{\text{LA, LV, LH, LD}\} := \text{2dDWT}\{\text{LR}\} \quad (3)$$

where the LA, LV, LH and LD are sub-bands containing wavelet coefficients for average, vertical, horizontal and diagonal details of the LR image, respectively. 2dDWT{LR} denotes the 2dDWT of the LR image.

The transformation is also applied on the corresponding HR training images to produce four HR wavelet Sub-Bands (HRSB):

$$\text{HRSB} = \{\text{HA,HV,HH,HD}\} := \text{2dDWT}\{\text{HR}\} \quad (4)$$

where the HA, HV, HH and HD denote the sub-bands containing wavelet coefficients for average, vertical, horizontal and diagonal details of the HR image, respectively.

Then the difference ΔSB (residual) between corresponding LRSB and HRSB is computed as:

$$\begin{aligned}\Delta\text{SB} &= \text{HRSB} - \text{LRSB} \\ &= \{\text{HA} - \text{LA}, \text{HV} - \text{LV}, \text{HH} - \text{LH}, \text{HD} - \text{LD}\} \\ &= \{\Delta\text{A}, \Delta\text{V}, \Delta\text{H}, \Delta\text{D}\}\end{aligned} \quad (5)$$

ΔSB is the target that we desire the network to produce with input LRSB. The feeding forward procedure is denoted as $f(\text{LRSB})$.

The cost of the network outputs is defined as:

$$\text{cost} = \frac{1}{2}\|\Delta\text{SB} - f(\text{LRSB})\|_2^2 \quad (6)$$

The weights and biases can be denoted as $(\Theta, \boldsymbol{b})$. Then the optimization problem is defined as:

$$(\Theta, \boldsymbol{b}) = \arg\min_{\Theta, \boldsymbol{b}} \frac{1}{2}\|\Delta\text{SB} - f(\text{LRSB})\|_2^2 + \lambda\|\Theta\|_2^2 \quad (7)$$

where the $\|\Theta\|_2^2$ is the standard weight decay regularization with parameter λ.

Essentially, we want our network to learn the differences between wavelet sub-bands of LR and HR images. By adding these differences (residual) to the input wavelet sub-bands, we will get the final super resolution wavelet sub-bands.

3.3. Generating SR Results

To produce SR results, the bicubic enlarged LR input images are transformed by 2dDWT to produce LRSB as Equation (3). Then LRSB is fed forward through the trained network to produce ΔSB. Adding LRSB and ΔSB together generates four SR wavelet Sub-Bands (SRSB) denoted as:

$$\begin{aligned} \text{SRSB} &= \{\text{SA}, \text{SV}, \text{SH}, \text{SD}\} \\ &= \text{LRSB} + \Delta\text{SB} \\ &= \{\text{LA} + \Delta\text{A}, \text{LV} + \Delta\text{V}, \text{LH} + \Delta\text{H}, \text{LD} + \Delta\text{D}\} \end{aligned} \tag{8}$$

Finally, 2dIDWT generates the SR image results:

$$\text{SR} = \text{2dIDWT}\{\text{SRSB}\} \tag{9}$$

3.4. Understanding Wavelet Prediction

Training in wavelet domain can boost up the training and testing procedure. Using wavelet coefficients encourages activation sparsity in hidden layers as well as output layer. Moreover, by using residuals, wavelet coefficients themselves become sparser and therefore easier for the network to learn sparse maps rather than dense ones. The histogram in Figure 4 illustrates the sparse distribution of all the ΔSB coefficients. This high level of sparsity further reduces the training time required for the network resulting in more accurate super resolution results.

In addition, training a deep network is actually to minimize a cost function which is usually defined by $l2$ norm. This particular norm is used because it homogeneously describes the quality of the output image comparing to the ground truth. The image quality is then quantified by the assessment metric PSNR. However, SSIM [40] has been proven to be a conceptually better way to describe the quality of an image (comparing to the target) which unfortunately can not be easily optimized. Nearly all the SR methods use SSIM as final testing metric but it is not emphasized in the training procedure.

However, DWSR encourages the network to produce more structural details. As shown in Figure 4, the SRSB has more defined structural details than LRSB after adding the predicted ΔSB. With Haar wavelet, every fine detail has different intensity of coefficients spreading in all four sub-bands. Overlaying four sub-bands together can enhance the structural details the network taking in by providing additional relationships between structural details. At a given spatial location, the first sub-band gives the general information of the image, following three detailed sub-bands provide horizontal/vertical/diagonal structural information to the network at this location. The structural correlation information between the sub-bands helps the network weights forming in a way to emphases the fine details.

By taking more structural similarity into account while training, the proposed network increases both the PSNR and SSIM assessments to deliver a visually improved SR result. Moreover, benefiting from wavelet domain information, DWSR produces SR results with less artifacts while other methods suffers from misleading artificial blocks introduced by bicubic (see Section 4.5).

4. Experimental Evaluation

4.1. Data Preparation

During the training phase, the NTIRE [41] 800 training images are used without augmentation. The NTIRE HR images $\{Y_i\}_{i=1}^{800}$ are down-sampled by the factor of c. Then the down-sampled images are enlarged busing bicubic interpolation by the same factor c to form the LR training images $\{X_i\}_{i=1}^{800}$. Note that the image Y_i is cropped so that its width and height be multiple of c. Therefore X_i and Y_i have the same size where Y_i represents the HR training image, X_i represents the corresponding LR training image. X_i and Y_i are then cropped to 41×41 pixels sub-images with 10 pixels overlapping for training.

For each sub-image from X_i, the LRSB is computed as Equation (3). For each corresponding sub-image from Y_i, the HRSB is computed as Equation (4). Then the residual ΔSB is computed as Equation (5).

During the testing phase, several standard testing data sets are used. Specifically, Set5 [13], Set14 [42], BSD100 [43], Urban100 [36] are used to evaluate our proposed method DWSR.

Both training and testing phases of DWSR only utilize the luminance channel information. For color images, Cr and Cb channels are directly enlarged by bicubic interpolation from LR images. These enlarged chrominance channels are combined with SR luminance channel to produce color SR results.

4.2. Training Settings

During the training process, several training techniques are used. The gradients are clipped to 0.01 by norm clipping option in the training package. We use Adam optimizer as described in [44] to updates Θ and $\boldsymbol{b}$. The initial learning rate is 0.01 and decreases by 25% every 20 epochs. The weight regulator is set to 1×10^{-3} to prevent over-fitting. Other than input and output layers, the DWSR has $N = 10$ same-sized convolutional hidden layers with filter size of $64 \times 3 \times 3 \times 64$. This configuration results in a network with only half of parameters in VDSR [38].

The training scheme is implemented with TensorFlow [45] package with Python 2.7 interaction interface. We use one GTX TITAN X GPU 12 GB for both the training and testing.

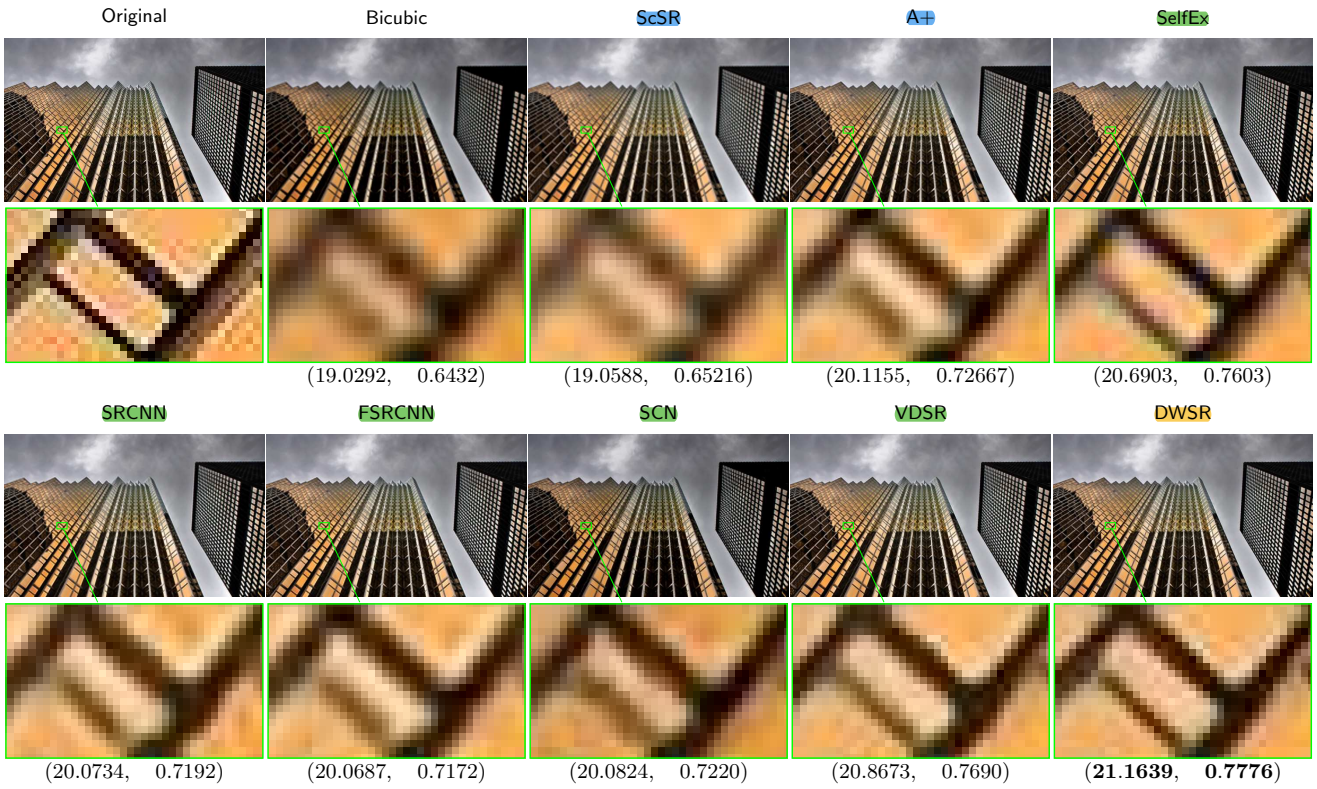

Figure 5: Test image No.19 in Urban100 data set. From top left to bottom right are results of: ground truth, bicubic, ScSR, A+, SelfEx, SRCNN, FSRCNN, SCN, VDSR, DWSR. The numeral assessments are labeled as (PSNR, SSIM). DWSR (bottom right) produces more defined structures with better SSIM and PSNR than state-of-the-art methods.

4.3. Convergence Speed

Since the gradients are clipped to a numerical large norm, with the high initial learning rate, DWSR reaches convergence with a really fast speed and produces practical results (see following reported evaluations). Figure 6 shows the convergence process during the training by plotting the evaluation of cost over training epochs. After 100 epochs, the network is fully converged and (Θ, b) is used for testing. The training procedure for 100 epochs takes about 4 hours to finish with one GPU.

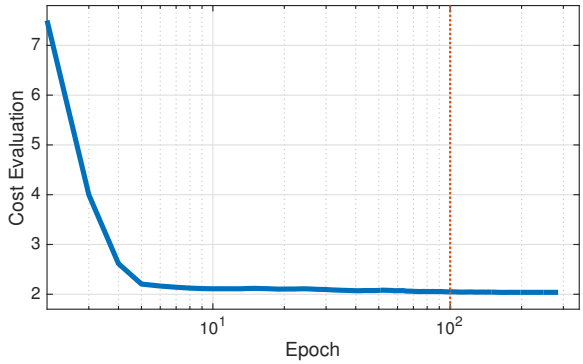

Figure 6: The evaluations of cost function (6) over training epochs for training scale factor 4. At 100 epoch, the network training convergences.

4.4. Comparison with State-of-the-Art

We compare DWSR with several state-of-the-art methods and use Bicubic as the baseline reference[1].

ScSR [4] and A+ [15] are selected to represent the sparse coding based and dictionary learning based methods. For deep learning based methods, DWSR is compared with SCN [46], SelfEx [36], FSRCNN [47], SRCNN [34] and VDSR [38]. We use publicly published testing codes from different authors, the tests are carried on GPU as mentioned above for deep learning based methods. For FSRCNN, SRCNN and sparse based methods we use their public CPU testing codes.

Table 1 shows the summarized results of PSNR and SSIM evaluations. The best results are shown in **red** and second best are shown in blue. DWSR has a clear

[1]Please refer to `http://signal.ee.psu.edu/DWSR.html` for high quality color images and to download our code.

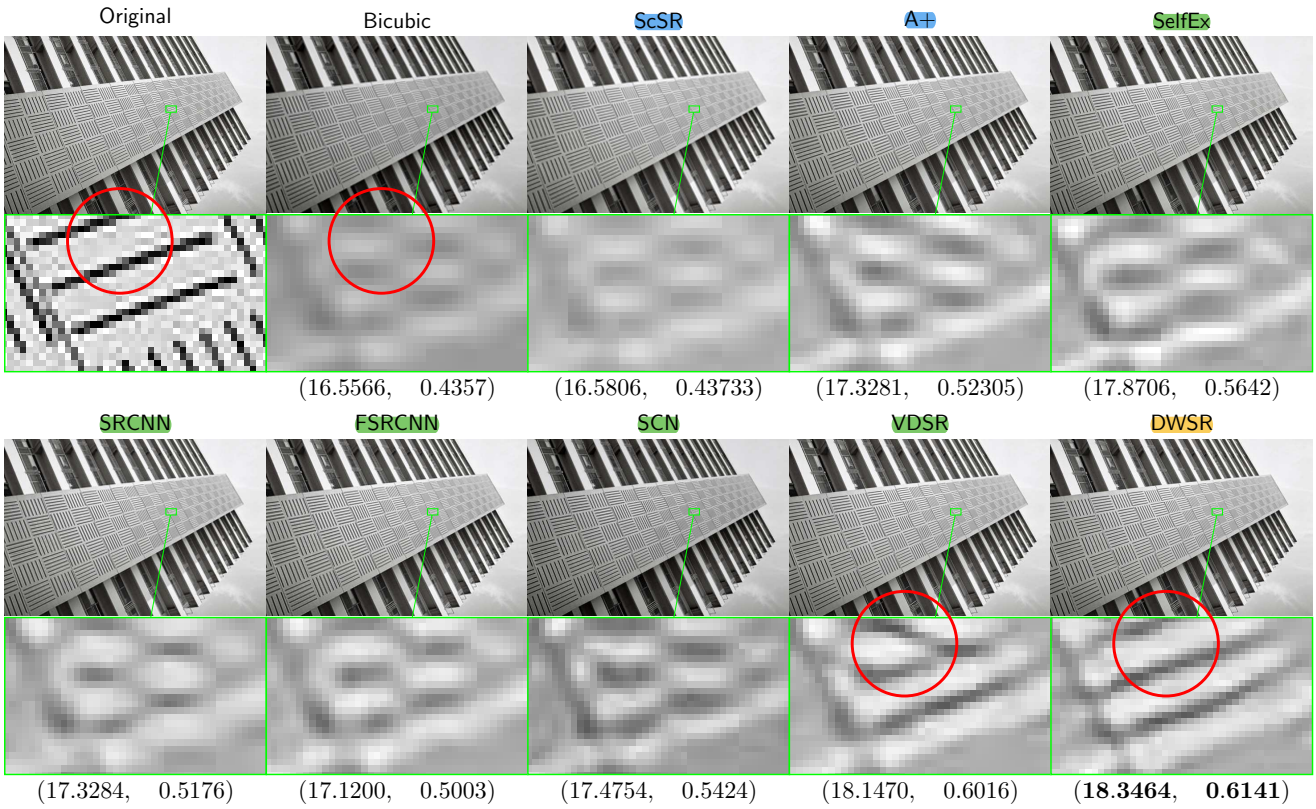

Figure 7: Test image No.92 in Urban100 data set. From top left to bottom right are results of: ground truth, bicubic, ScSR, A+, SelfEx, SRCNN, FSRCNN, SCN, VDSR, DWSR. The numeral assessments are labeled as (PSNR, SSIM). DWSR (bottom right) produces more fine structures with better SSIM and PSNR than state-of-the-art methods. Also note DWSR does not produce artifacts diagonal edges in the red circled region.

advantage on the large scaling factors owing to its reliance on incorporating the structural information and correlation from wavelet transform sub-bands. For large scale factors, DWSR delivers better results than the best known method (VDSR) with **only half** parameters benefiting from training in wavelet feature domain.

Table 2 shows the execution time of different methods. Since DWSR only has half of the parameters than the most parameterized method (VDSR) and benefiting from really sparse network activations, DWSR takes much less time to apply super-resolution. For 2K images in NTIRE testing set, DWSR takes less than 0.1s to produce the outputs of the network including the loading time from GPU.

Figure 5 shows SR results of a testing image from Urban100 dataset with scale factor 4. Overall, deep learning based methods produce better results than sparse coding based and dictionary learning based methods. Compared to SRCNN, DWSR produces more defined structures benefiting from training in wavelet domain. Compared to VDSR, DWSR results give higher PSNR and SSIM values using less than half parameters of VDSR with a faster speed. Visually, the edges are more enhanced in DWSR than other state-of-the-art methods and is clearly illustrated in the enlarged areas. The image generated by DWSR has less artifacts that are caused by initial bicubic interpolation of LR image and results in sharper edges which are consistent with the ground truth image. Also quite clearly, DWSR has an advantage on reconstructing edges especially diagonal ones due to the fact that these structural information are prominently emphasized with sub-bands in Haar wavelets coefficients.

4.5. Large Scaling Factor SR Artifacts

Figure 7 illustrates SR results from different methods with scale factor 4. DWSR produces more enhanced details than state-of-the-art methods. Moreover, since the scale factor is large for bicubic interpolations to keep the structural information, some artificial blocks are introduced during the bicubic enlargement. Meanwhile nearly all the deep learning based methods are utilizing the bicubic interpolations as the starting point, these artificial blocks get more pronounced during the SR enhancements. Eventually,

Table 1: PSNR and SSIM result comparisons with other approaches for 4 different datasets.

PSNR | SSIM		Bicubic [Baseline]		ScSR [TIP 10]		A+ [ACCV 14]		SelfEx [CVPR 15]		FSRCNN [ECCV 16]		SRCNN [PAMI 16]		VDSR [CVPR 16]		DWSR [ours]	
Set5	x2	33.64	0.9292	35.78	0.9485	36.55	0.9544	36.47	0.9538	36.94	0.9558	36.66	0.9542	**37.52**	**0.9586**	37.43	0.9568
	x3	30.39	0.8678	31.34	0.8869	32.58	0.9088	32.57	0.9092	33.06	0.9140	32.75	0.9090	33.66	0.9212	**33.82**	**0.9215**
	x4	28.42	0.8101	29.07	0.8263	30.27	0.8605	30.32	0.8640	30.55	0.8657	30.48	0.8628	31.35	0.8820	**31.39**	**0.8833**
Set14	x2	30.22	0.8683	31.64	0.8940	32.29	0.9055	32.24	0.9032	32.54	0.9088	32.42	0.9063	33.02	0.9102	**33.07**	**0.9106**
	x3	27.53	0.7737	28.19	0.7977	29.13	0.8188	29.16	0.8196	29.37	0.8242	29.28	0.8209	29.77	0.8308	**29.83**	**0.8308**
	x4	25.99	0.7023	26.40	0.7218	27.33	0.7489	27.40	0.7518	27.50	0.7535	27.40	0.7503	28.01	0.7664	**28.04**	**0.7669**
B100	x2	29.55	0.8425	30.77	0.8744	31.21	0.8864	31.18	0.8855	31.66	0.8920	31.36	0.8879	**31.85**	**0.8960**	31.80	0.8940
	x4	25.96	0.6672	26.61	0.6983	26.82	0.7087	26.84	0.7106	26.92	0.7201	26.84	0.7101	27.23	0.7238	**27.25**	**0.7240**
Urban100	x2	26.66	0.8408	28.26	0.8828	29.20	0.8938	29.54	0.8967	29.87	0.9010	29.50	0.8946	**30.76**	0.9140	30.46	**0.9162**
	x4	23.14	0.6573	24.02	0.7024	24.32	0.7186	24.78	0.7374	24.61	0.7270	24.52	0.7221	25.18	0.7524	**25.26**	**0.7548**

Table 2: Results of the execution time comparison to other approaches

		ScSR [TIP 10]	A+ [ACCV 14]	SelfEx [CVPR 15]	FSRCNN [ECCV 16]	SRCNN [PAMI 16]	VDSR [CVPR 16]	DWSR [ours]
Set5	x2	80.22	0.58	45.76	0.30	2.56	0.13	**0.06**
	x3	82.67	0.32	32.28	0.23	2.63	0.13	**0.05**
	x4	84.88	0.24	29.32	0.26	2.16	0.12	**0.06**
Set14	x2	86.12	0.85	112.3	0.32	4.52	0.25	**0.07**
	x3	91.52	0.59	76.02	0.42	4.25	0.26	**0.08**
	x4	89.25	0.32	66.06	0.39	4.68	0.25	**0.07**
B100	x2	98.03	0.60	62.02	0.32	2.65	0.16	**0.09**
	x4	100.43	0.26	36.67	0.39	2.98	0.26	**0.12**
Urban100	x2	1021.06	2.96	663.66	2.23	23.2	0.98	**0.33**
	x4	1282.33	1.21	662.68	2.35	25.6	1.07	**0.38**

the enhancements on the artificial blocks produce artificial edges in the SR results. For instance, in Figure 7, these blocks and artificial edges are labeled within red circles for bicubic and VDSR. The diagonal edges are introduced by SR enhancement on the artificial blocks from bicubic enlargement, which are not present in the ground truth image.

However, DWSR utilizes wavelet coefficients to take in more structural correlation information into account which does not enhance the artificial blocks and produces edges more similar to the ground truth.

5. Conclusion

Our work presents a deep wavelet super resolution (DWSR) technique that recovers the "missing details" by using (low-resolution) wavelet sub-bands as inputs. DWSR is significantly economical in the number of parameters compared to most state-of-the-art methods and yet achieves competitive or better results. We contend that this is because wavelets provide an image representation that naturally simplifies the mapping to be learned. While we used the Haar wavelet, effects of different wavelet basis can be examined in future work. Of particular interest could be to learn the "optimal" wavelet basis for the SR task.

6. Acknowledgment

This work is supported by NSF Career Award to V. Monga.

References

[1] S. C. Park, M. K. Park, and M. G. Kang, "Super-resolution image reconstruction: a technical overview," *Signal Processing Magazine, IEEE*, vol. 20, no. 3, pp. 21–36, 2003.

[2] S. Farsiu, M. D. Robinson, M. Elad, and P. Milanfar, "Fast and robust multiframe super resolution," *Image processing, IEEE Transactions on*, vol. 13, no. 10, pp. 1327–1344, 2004.

[3] J. Yang, J. Wright, T. Huang, and Y. Ma, "Image super-resolution as sparse representation of raw image patches," in *Computer Vision and Pattern Recognition, IEEE Conference on*, pp. 1–8, 2008.

[4] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *Image Processing, IEEE Transactions on*, vol. 19, no. 11, pp. 2861–2873, 2010.

[5] D. Glasner, S. Bagon, and M. Irani, "Super-resolution from a single image," in *Computer Vision, IEEE International Conference on*, pp. 349–356, 2009.

[6] G. Freedman and R. Fattal, "Image and video upscaling from local self-examples," *ACM Trans. Graph.*, vol. 28, no. 3, pp. 1–10, 2010.

[7] J. Yang, Z. Lin, and S. Cohen, "Fast image super-resolution based on in-place example regression," in *Computer Vision and Pattern Recognition, IEEE Conference on*, pp. 1059–1066, 2013.

[8] S. Minaee, A. Abdolrashidi, and Y. Wang, "Screen content image segmentation using sparse-smooth decomposition," *arXiv preprint arXiv:1511.06911*, 2015.

[9] Z. Cui, H. Chang, S. Shan, B. Zhong, and X. Chen, "Deep network cascade for image super-resolution," in *Computer Vision, ECCV*, pp. 49–64, Springer, 2014.

[10] W. T. Freeman, E. C. Pasztor, and O. T. Carmichael, "Learning low-level vision," *International journal of computer vision*, vol. 40, no. 1, pp. 25–47, 2000.

[11] H. Chang, D.-Y. Yeung, and Y. Xiong, "Super-resolution through neighbor embedding," in *Computer Vision and Pattern Recognition, IEEE Conference on*, vol. 1, pp. I–I, 2004.

[12] K. I. Kim and Y. Kwon, "Single-image super-resolution using sparse regression and natural image prior," *Pattern Analysis and Machine Intelligence, IEEE transactions on*, vol. 32, no. 6, pp. 1127–1133, 2010.

[13] M. Bevilacqua, A. Roumy, C. Guillemot, and M. L. Alberi-Morel, "Low-complexity single-image super-resolution based on nonnegative neighbor embedding," 2012.

[14] R. Timofte, V. De, and L. Van Gool, "Anchored neighborhood regression for fast example-based super-resolution," in *Computer Vision, IEEE International Conference on*, pp. 1920–1927, 2013.

[15] R. Timofte, V. De Smet, and L. Van Gool, "A+: Adjusted anchored neighborhood regression for fast super-resolution," in *Computer Vision, ACCV*, pp. 111–126, Springer, 2014.

[16] K. Jia, X. Wang, and X. Tang, "Image transformation based on learning dictionaries across image spaces," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 35, no. 2, pp. 367–380, 2013.

[17] Z. Wang, Y. Yang, Z. Wang, S. Chang, W. Han, J. Yang, and T. S. Huang, "Self-tuned deep super resolution," *arXiv preprint arXiv:1504.05632*, 2015.

[18] M. E.-S. Wahed, "Image enhancement using second generation wavelet super resolution," *International Journal of Physical Sciences*, vol. 2, no. 6, pp. 149–158, 2007.

[19] H. Ji and C. Fermüller, "Robust wavelet-based super-resolution reconstruction: theory and algorithm," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 4, pp. 649–660, 2009.

[20] H. Demirel, S. Izadpanahi, and G. Anbarjafari, "Improved motion-based localized super resolution technique using discrete wavelet transform for low resolution video enhancement," in *Signal Processing, IEEE European Conference on*, pp. 1097–1101, 2009.

[21] M. D. Robinson, C. A. Toth, J. Y. Lo, and S. Farsiu, "Efficient fourier-wavelet super-resolution," *Image Processing, IEEE Transactions on*, vol. 19, no. 10, pp. 2669–2681, 2010.

[22] G. Anbarjafari and H. Demirel, "Image super resolution based on interpolation of wavelet domain high frequency subbands and the spatial domain input image," *ETRI journal*, vol. 32, no. 3, pp. 390–394, 2010.

[23] N. Nguyen and P. Milanfar, "An efficient wavelet-based algorithm for image superresolution," in *Image Processing. IEEE International Conference on*, vol. 2, pp. 351–354, 2000.

[24] C. Jiji, M. V. Joshi, and S. Chaudhuri, "Single-frame image super-resolution using learned wavelet coefficients," *International journal of Imaging systems and Technology*, vol. 14, no. 3, pp. 105–112, 2004.

[25] S. Mallat and G. Yu, "Super-resolution with sparse mixing estimators," *Image Processing, IEEE Transactions on*, vol. 19, no. 11, pp. 2889–2900, 2010.

[26] M. F. Tappen, B. C. Russell, and W. T. Freeman, "Exploiting the sparse derivative prior for super-resolution and image demosaicing," in *Statistical and Computational Theories of Vision, IEEE Workshop on*, Citeseer, 2003.

[27] W. Dong, L. Zhang, G. Shi, and X. Wu, "Image deblurring and supe r-resolution by adaptive sparse domain selection and adaptive regularization," *Image Processing, IEEE Transactions on*, vol. 20, no. 7, pp. 1838–1857, 2011.

[28] K. Kinebuchi, D. D. Muresan, and T. W. Parks, "Image interpolation using wavelet based hidden markov trees," in *Acoustics, Speech, and Signal Processing, IEEE International Conference on*, vol. 3, pp. 1957–1960, 2001.

[29] S. Zhao, H. Han, and S. Peng, "Wavelet-domain hmt-based image super-resolution," in *Image Processing, IEEE International Conference on*, vol. 2, pp. II–953, 2003.

[30] H. Chavez-Roman and V. Ponomaryov, "Super resolution image generation using wavelet domain interpolation with edge extraction via a sparse representation," *IEEE Geoscience and Remote Sensing Letters*, vol. 11, no. 10, pp. 1777–1781, 2014.

[31] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[32] Y. Bengio, P. Lamblin, D. Popovici, H. Larochelle, *et al.*, "Greedy layer-wise training of deep networks," *Advances in neural information processing systems*, vol. 19, p. 153, 2007.

[33] C. Poultney, S. Chopra, Y. L. Cun, *et al.*, "Efficient learning of sparse representations with an energy-based model," in *Advances in neural information processing systems*, pp. 1137–1144, 2006.

[34] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *Computer Vision, ECCV*, pp. 184–199, Springer, 2014.

[35] T. Guo, H. S. Mousavi, and V. Monga, "Deep learning based image super-resolution with coupled backpropagation," in *Signal and Information Processing, IEEE Global Conference on*, pp. 237–241, 2016.

[36] J.-B. Huang, A. Singh, and N. Ahuja, "Single image super-resolution from transformed self-exemplars," in *Computer Vision and Pattern Recognition, IEEE Conference on*, pp. 5197–5206, 2015.

[37] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Computer Vision and Pattern Recognition, IEEE Conference on*, pp. 770–778, 2016.

[38] J. Kim, J. K. Lee, and K. M. Lee, "Accurate image super-resolution using very deep convolutional networks," in *Computer Vision and Pattern Recognition, IEEE Conference on*, June 2016.

[39] S. Mallat, *A wavelet tour of signal processing: the sparse way*. Academic press, 2008.

[40] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *Image Processing, IEEE Transactions on*, vol. 13, no. 4, pp. 600–612, 2004.

[41] R. Timofte, E. Agustsson, L. Van Gool, M.-H. Yang, L. Zhang, *et al.*, "Ntire 2017 challenge on single image super-resolution: Methods and results," in *Computer Vision and Pattern Recognition Workshops, IEEE Conference on*, July 2017.

[42] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse-representations," in *International conference on curves and surfaces*, pp. 711–730, Springer, 2010.

[43] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proc. 8th Int'l Conf. Computer Vision*, vol. 2, pp. 416–423, July 2001.

[44] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[45] M. Abadi, A. Agarwal, and P. B. et. al., "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015. Software available from tensorflow.org.

[46] Z. Wang, D. Liu, J. Yang, W. Han, and T. Huang, "Deep networks for image super-resolution with sparse prior," in *Computer Vision, IEEE International Conference on*, pp. 370–378, 2015.

[47] C. Dong, C. C. Loy, and X. Tang, "Accelerating the super-resolution convolutional neural network," in *Computer Vision, ECCV*, pp. 391–407, Springer, 2016.