Balanced Two-Stage Residual Networks for Image Super-Resolution

Yuchen Fan*†, Honghui Shi*†, Jiahui Yu†, Ding Liu†, Wei Han†,
Haichao Yu†, Zhangyang Wang‡, Xinchao Wang†, and Thomas S. Huang†
†Beckman Institute, University of Illinois at Urbana-Champaign, Urbana, IL 61801
‡Department of Computer Science & Engineering, Texas A&M University, College Station, TX 77843
{yuchenf4, hshi10, dingliu2, weihan3, haichao3, xinchao, t-huang1}@uiuc.edu atlaswang@tamu.edu

Abstract

In this paper, balanced two-stage residual networks (BTSRN) are proposed for single image super-resolution. The deep residual design with constrained depth achieves the optimal balance between the accuracy and the speed for super-resolving images. The experiments show that the balanced two-stage structure, together with our lightweight two-layer PConv residual block design, achieves very promising results when considering both accuracy and speed. We evaluated our models on the New Trends in Image Restoration and Enhancement workshop and challenge on image super-resolution (NTIRE SR 2017). Our final model with only 10 residual blocks ranked among the best ones in terms of not only accuracy (6th among 20 final teams) but also speed (2nd among top 6 teams in terms of accuracy). The source code both for training and evaluation is available in https://github.com/ychfan/sr_ntire2017.

1. Introduction

Deep neural networks have achieved great success in the recent years for many key computer vision and image processing tasks such as image classification, object detection, and image super-resolution (SR) [11][16]. Recent work in training very deep networks suggests that, usually the deeper the network structure is, the better performance it can achieve. Many state-of-the-art approaches therefore focus on training deeper neural networks with techniques such as deep residual learning [11], batch normalization [13].

While deep models tend to yield high accuracy, they lead to very heavy computation in the task of searching the best deep network architecture as well as training and testing the deep neural networks. This is undesirable when computational resources are limited, and when the short training time or real-time testing performance are required.

In this paper, we focus on applying deep networks for single image super-resolution. We adopt the deep residual design to ensure the model accuracy but also constrain the model depth to only 10 residual blocks to ensure the efficiency for training and testing our models. Our experiments further show that we can achieve the best trade-offs between predicting accuracy and speed by using a relatively balanced two-stage structure

Our contribution is therefore novel balanced two-stage residual networks (BTSRN) with limited depths for single image SR. We explore the trade-offs between model accuracy and efficiency. Particularly, our models using 10 residual blocks perform the best with relatively balanced 6+4 two-stage structure for image SR task. Our models are tested in the 2017 New Trends in Image Restoration and Enhancement workshop and challenge on image super-resolution (NTIRE SR 2017) [33]. Our final model using only 10 residual blocks ranked among the top ones, both in accuracy (6th among 20 final teams) and speed (2nd among top 6 teams in terms of accuracy).

Besides the proposed the novel architecture for the challenge, we also perform extensive experiments in regard to up-sampling strategy, batch normalization, residual blocks etc., and discover a lightweight and efficient residual block design that best suites our proposed model.

The rest of the paper is organized as follows. Sec. 2 reviews the related work in image SR, Sec. 3 provides the details of our model, Sec. 4 describes our results and implementations, and Sec. 5 concludes the whole paper.

2. Related Work

Image SR has been studied in the research community over the past few decades, and a large number of papers have been published in this field. In this section we will focus on only single image SR and elaborate the neural network based approaches as a major trend towards solving this problem.

Single image SR is the task of recovering a HR image from only one LR observation. A recent comprehensive

*Authors contributed equally to this work

review for this task can be found in the work of Yang et al. [40]. Existing methods can be broadly classified into three categories: interpolation based [5], image statistics based [6, 32] and example based methods [42, 34, 36].

Interpolation based methods include linear, bicubic and Lanczos filtering [5], which usually run very fast due to the low complexity of algorithm. However, the simplicity of these methods leads to the failure of modeling the complex mapping between the LR feature space and the corresponding HR feature space, generating overly-smoothed unsatisfactory regions.

Image statistics based methods utilize the statistical edge information to reconstruct HR images, for example, in [6, 32]. They rely on the priors of edge statistics in images while facing the shortcoming of losing high-frequency detail information especially in the case of large upscaling factors.

The current most popular and successful approaches are built on example based learning techniques, which aim to learn the correspondence between LR feature space and HR feature space through a large number of representative example pairs. The pioneer work in this area includes [8].

Given the origin of example pairs, these methods can be further categorized into three classes: *self-example based* [9, 7], *external-example based* methods [42, 34] and the joint of them [39]. Self-example based methods only exploit the single input LR image as references, and extract example pairs merely from the KR image across different scales to predict the HR image. This type of methods usually works well in the images containing repetitive patterns or textures but lacks the richness of image structures outside the input image and thus fails to generate satisfactory prediction for images of other classes. Huang et al. [12] extends the idea of self-example based SR, by building self dictionaries for handling geometric transformations.

External-example based methods first utilize the example pairs extracted from an external dataset, in order to learn the universal image characteristics between LR feature space and HR feature space, and then apply the learned the mapping for SR. Usually, the representative patches from external datasets are compactly embodied by pre-trained dictionaries. One representative approach is the sparse coding based method [42, 41, 38]. For example, in [41] two coupled dictionaries are trained for LR feature space and HR patch feature space, respectively, such that the LR patch over LR dictionary and its corresponding HR patch over HR dictionary share the same sparse representation. Although it is able to capture the universal LR-HR correspondence from external datasets and recover fine details and sharpened edges, it suffers the high computational cost for solving complicated non-linear optimization problems.

Timofte et al. [34, 35] propose a neighboring embedding approach for SR, and formulate the problem as optimizing a least square with l_2 norm regularization, which drastically reduces the computation complexity compared with [42, 41]. Neighboring embedding approaches approximate HR patches as a weighted average of similar training patches in a low dimensional manifold.

Random forest can be built for SR without dictionary learning in the work of [29, 28]. These methods embrace the fast inference time but usually suffer from the huge model size.

Recently, inspired by the achievement of many computer vision tasks obtained by deep learning, neural networks have been successfully applied for image SR. Dong et al. [3] first exploit a three layer convolutional neural network, termed SRCNN, to regress the complex non-linear mapping between the LR image and the HR counterpart. A neural network that encodes the sparse representation prior for image SR is designed by Wang et al. [37, 20] demonstrating the benefit of domain expertise from sparse coding in the task of image SR. Kim et al. [15] propose a very deep CNN with residual architecture to achieve outstanding SR performance, which utilizes broader contextual information with larger model capacity. Another network is designed by Kim et al. [16], which has recursive architectures with skip connection for image SR to boost performance while only exploiting a small number of model parameters. Liu et al. [19] utilize a mixture of network models to enhance the power of single network model.

Shi et al. [30] use a compact network model to conduct convolutions on LR images directly and learn an array of upscaling filters in the last layer, which considerably reduces the computation cost for real-time SR. Similarly, Dong et al. [4] adopt deconvolution layers to accelerate SRCNN in combination with smaller filter sizes and more convolution layers.

More recently, the research of various evaluation metric of SR has drawn increasing attention in the community. Different loss functions for neural networks have been studied. Since the conventional pixel-wise loss such as mean squared error (MSE) tends to find the average of possible HR candidates and leads to overly-smoothed results, Johnson et al. [14] propose perceptual loss for large upscaling SR in order to generate realistic details in HR prediction. In the works of [18, 25], this idea is combined with generative adversarial network (GAN) [10] to further enhance the details of HR prediction.

Among the aforementioned methods, the very deep residual networks proposed by Kim et al. [15] is the current state-of-the-art in terms of accuracy for image SR tasks. However, the expensive computational load for training and testing of such models encourages us to find more efficient yet accurate models towards real-time applications. Our proposed method is trying to find a better balanced solu-

tion between the trade-offs of accuracy and speed and will be introduced in the following sections.

3. Method

We propose balanced two-stage residual networks (BTSRN) for image super-resolution. The proposed model, as shown in Figure 1, mainly contains two stages: low resolution (LR) stage and high resolution (HR) stage. In the low and high resolution stages, the residual networks [11] are deployed with 6 and 4 residual blocks respectively. The two stages are connected with up-sampling layers.

3.1. Balanced Two Stages

Deep residual structures are necessary in both the low and high resolution stages. In the low resolution stages, the feature maps have relatively small size. Receptive fields are extended effectively to capture enough spatial context and high level information with stacked convolution layers. In the high resolution stages, the bigger feature maps contain more information and are more correlated to output images. Multiple residual blocks refine interaction between neighbor pixels and reduce checkerboard artifacts [22] by up-sampling. However, the feature maps need s^2 times processing time and memory space in each layer for super-resolution with magnification factor s. The balance of residual block numbers between low and high resolution stages is expected to achieve a good trade-off between accuracy and speed. Compared with VDSR [15], the proposed approach takes low resolution image as input and reduces the computational redundancy; compared with ESPCN [30], SRGAN [18] and EnhanceNet [25], the proposed networks perform better refinement in the high resolution space and yield fewer artifacts.

3.2. Up-sampling

For the up-sampling layers, the element sum of nearest neighbor up-sampling and deconvolution is employed. Deconvolution can up-sample feature maps with linear kernels. The kernels may overlap for up-sampled feature maps. The overlap may be uneven with unpaired stride and size of kernels. The uneven overlap will cause checkerboard artifacts [22] in both output and gradient. To reduce the artifacts, the stride and size of kernels are equal to scaling factor for x2 and x3, and two x2 up-sampling are applied for x4 scaling. The skip connections in up-sampling layers are achieved by nearest neighbor up-sampled feature maps. The gradient can bypass deconvolution and be directly feedbacked to low resolution stage.

3.3. Residual Blocks

Residual networks are stacked by residual blocks with skip connections. The design of residual blocks decides the performance of the networks. Multiple settings of the residual blocks were investigated in Figure 1, including residual blocks in PixelCNN [24], gated convolution blocks in advanced PixelCNN [23], gated convolution blocks in PixelCNN++ [26], and the proposed projected convolution (PConv) structure.

In PixelCNN, the original proposed structure consists of stacked convolutional layers with ReLU as non-linear activation. The three layers are 1x1, 3x3, and 1x1 convolutions respectively. Later the gated convolotional structure is proposed in PixelCNN for better performance. After the first 3x3 convolutional layer, channels are divided into two branches. Hyperbolic tangent and sigmoid operations are applied to the feature map respectively and appended with element-wise multiplication. One difference in PixelCNN++ is that the hyperbolic tangent branch is replaced by identity mapping. All of these structures are designed for specific tasks and the details can be find in Figure 1. We proposed a simple and efficient residual block structure called projected convolution (PConv) that has 1x1 convolution as feature map projection to reduce input size of 3x3 convolution. The proposed model achieves good trade-off between the accuracy and the speed.

3.4. Batch Normalization

Batch normalization [13] technique is not adopted in our proposed model architecture. Batch normalization is first introduced in [13] and designed to reduce internal covariate shift. It turns out that batch normalization can reduce the gradient vanishing problems thus makes training much faster. And its widely used in previous super-resolution models [18] [31]. However, we found batch normalization is not suitable for super-resolution task. As shown in [1], batch normalization makes the networks invariant to data re-centering and re-scaling. Because super-resolution is a regressing task, the target outputs are highly correlated to inputs first order statistics. Actually, weight normalization networks are still sensitive to input mean and variance. But our experiments showed that there are no noticeable differences in performance between models with and without weight normalization [27].

3.5. Training

The deep networks predict the residual images between the ground truth (high-resolution images) and the bicubic-upscaled images, as shown in Figure 1. Patches from training image pairs are batched and fed into the neural networks. The loss function is defined as the mean square error on RGB channels between the neural network outputs $F(X^{LR}; \Theta)$ and corresponding residual images $(X^{HR} -$

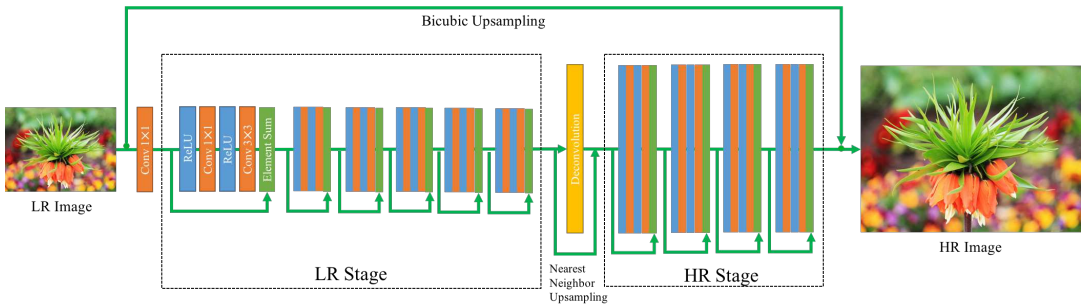
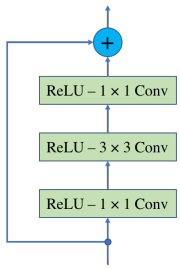
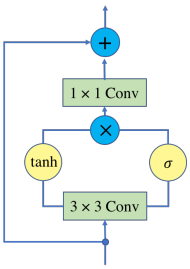


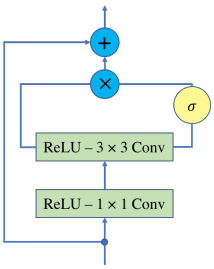
Figure 1: Architecture of the proposed network



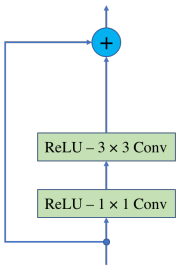
(a) Residual block in PixelCNN



(b) Gated conv in PixelCNN



(c) Gated conv in PixelCNN++



(d) PConv block in proposed BTSRN model

Figure 2: Comparison of different residual block designs

$B(X^{LR})$):

$$L(\Theta) = \frac{1}{mn}\sum_{i=1}^{n}\sum_{j=1}^{m}||F(X_{ij}^{LR};\Theta) - (X_{ij}^{HR} - B(X_{ij}^{LR}))||^2 \quad (1)$$

where B represents bicubic operation over low resolution image patches, n is the number of training samples and m is the number of patches of one training sample.

4. Experiments

4.1. Implementation

The proposed method is evaluated in the super-resolution challenge of NTIRE 2017 [33]. The challenge takes DIV2K dataset as benchmark, which includes 1000 DIVerse 2K resolution RGB images. 800 images are for training, 100 are for validation and 100 are for testing. The challenge contains two tracks with different down-sampling method (i.e. bicubic and unknown). For each track, there are three competitions with different down-sampling scale (i.e. x2, x3, x4). The models are evaluated in term of Peak Signal-to-Noise Ratio (PSNR) on RGB channels. $6 + s$ boundary pixels are ignored in the evaluation, where s is the magnification factor,

During training, all the images are cropped into 108x108-pixel patches with random flipping and rotation for augmentation. Each step takes 32 image patches as a batch. Adam optimizer [17] is used with the initial learning rate of 0.001. The learning rate is exponentially decreased by a fixed factor 0.6 after each iteration for faster convergence. For each track, we train separate models for each scale (i.e. x2, x3, x4) on pairs of high-resolution and downscaled low-resolution images.

During testing, data augmentations of flipping and rotation are used and averaged on the float data type. The float images are then rounded to uint8 data type and saved. For single image and its augmented images, the testing is performed on 8 GPUs synchronously, which are fully paralleled and achieve nearly 8x speed up.

The system is implemented in Python with Tensorflow, and runs on modern GPUs. Because the networks are trained with small patches, they can be fitted into regular GPUs with 8GBs memory. Our training is performed on single GPU, enabling us to have better utilization of GPUs and compare different network topologies and hyperparameters.

4.2. Results

To find the optimal model structure, we trained models on DIV2K 800-image training set and evaluated them on 100-image validation set.

High Stage	PSNR (dB)
3	34.14024
4	34.16181
5	34.12123

Table 1: Comparison of blocks in high resolution stage given low stage with 6 blocks

Low Stage	PSNR (dB)
5	34.15799
6	34.16181
7	34.15126
8	34.16568
9	34.18892

Table 2: Comparison of blocks in low resolution stage given high stage with 4 blocks

4.2.1 Balanced Two Stages

Different combinations of number of blocks in low and high resolution stages are tried to verify the advances of balanced structure.

First, by fixing the low resolution stages to 6 blocks, performance of different number of high resolution stage is shown in Table 1. The results show that 4 blocks in high resolution stage is adequate for image refinement, additional blocks will cause the models slow to converge.

Then, by fixing the high resolution stages to 4 blocks, performance of different number of high resolution stage is shown in Table 2. The results show that the more blocks (less than 10) in low resolution, the better the performance is. And networks with 6 low resolution blocks are good compromise between accuracy and speed.

In addition, by fixing the total number of blocks in low and high resolution stages to 10, which equals to fix the number of model parameters, performance is shown in Figure 3. The results show that networks, with 7 and 3 layers for low and high resolution stages respectively, achieve the best performance.

By quantitatively comparing results between different architectures, we can draw the conclusion that the proposed balanced two-stage residual networks (BTSRN) yield best trade-off between accuracy and speed.

4.2.2 Residual Blocks

To compare different design of residual blocks, including residual block in PixelCNN 2a, gated convolution in PixelCNN++ 2c and proposed PConv blocks 2d, we fix the input of residual blocks to 128 nodes and input 1x1 convolution

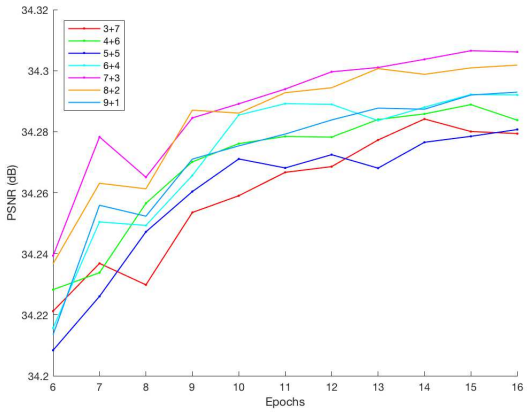


Figure 3: Comparison of low and high stages residual blocks combination (low+high)

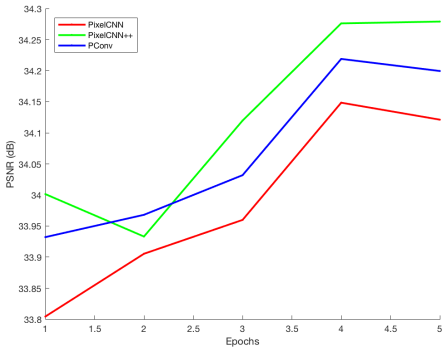


Figure 4: Comparison of residual blocks design

to 32 nodes, and results are shown in Figure 4. Compared to residual block in PixelCNN, the proposed PConv blocks achieve better performance, need the same time and less memory for training. Although gated convolution in PixelCNN++ has better performance, it needs nearly double time and memory for training compared to proposed residual blocks. So the proposed PConv blocks make good trade-off between accuracy and efficiency.

4.3. NTIRE 2017 Challenge

The submitted models for NTIRE 2017 super-resolution challenge are based on the proposed balanced two stage residual networks (BTSRN) with 6 and 4 residual blocks in low and high resolution stage respectively. The proposed PConv blocks are employed with 128 nodes as input and 64 nodes after 1x1 convolution layer. The networks are trained with training and validation dataset, totally 900 images, and evaluated on the 100-image testing set. The results are shown in Table 3.

Qualitative comparison results can be found in Figure

Scale	BTSRN	Bicubic
x2	34.19	31.01
x3	30.44	28.22
x4	28.49	26.65

Table 3: Final results in NTIRE 2017 super-resolution challenge

Dataset	Scale	BTSRN	VDSR	Bicubic
Set5	x2	37.75	37.53	33.64
	x3	34.03	33.66	30.38
	x4	31.85	31.35	28.42
Set14	x2	33.20	33.03	30.22
	x3	29.90	29.77	27.51
	x4	28.20	28.01	25.95
B100	x2	32.05	31.90	29.55
	x3	28.97	28.82	27.20
	x4	27.47	27.29	25.97
Urban100	x2	31.63	30.76	26.87
	x3	27.75	27.14	24.46
	x4	25.74	25.18	23.14

Table 4: Benchmark results in PSNR

5 and the results clearly demonstrate our method achieves much sharper super-resolved images in general and significant smoother results in edges compared with bicubic interpolation method.

4.4. Benchmarks

The proposed balanced two stage residual networks (BTSRN) are further evaluated on benchmarks including Set5 [2], Set14 [43], B100 [21, 35] and Urban100 [12]. To make the results comparable with state-of-the-art methods, we follow the same evaluation procedure by calculating PSNR on luminance channel and ignoring two boundary pixels. The results are shown in Table 4. Compared with state-of-the-art VDSR [15] approach, our proposed BTSRN achieves significant improvements in PSNR on all the benchmarks.

5. Conclusion

In this work, we proposed novel balanced two-stage residual networks (BTSRN) with lightweight yet efficient two-layer PConv residual blocks. The experiments show that our proposed model achieves well-balanced results in terms of both model accuracy and efficiency. For future work, we will further explore the proposed model structure in two directions: speed-up without losing accuracy and deeper models with better performance.

References

[1] J. L. Ba, J. R. Kiros, and G. E. Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016. 3

[2] M. Bevilacqua, A. Roumy, C. Guillemot, and M. L. Alberi-Morel. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. 2012. 6

[3] C. Dong, C. C. Loy, K. He, and X. Tang. Learning a deep convolutional network for image super-resolution. In *European Conference on Computer Vision*, pages 184–199. Springer, 2014. 2

[4] C. Dong, C. C. Loy, and X. Tang. Accelerating the super-resolution convolutional neural network. In *European Conference on Computer Vision*, pages 391–407. Springer, 2016. 2

[5] C. E. Duchon. Lanczos filtering in one and two dimensions. *Journal of Applied Meteorology*, 18(8):1016–1022, 1979. 2

[6] R. Fattal. Image upsampling via imposed edge statistics. In *ACM Transactions on Graphics (TOG)*, volume 26, page 95. ACM, 2007. 2

[7] G. Freedman and R. Fattal. Image and video upscaling from local self-examples. *ACM Transactions on Graphics (TOG)*, 30(2):12, 2011. 2

[8] W. T. Freeman, T. R. Jones, and E. C. Pasztor. Example-based super-resolution. *IEEE Computer graphics and Applications*, 22(2):56–65, 2002. 2

[9] D. Glasner, S. Bagon, and M. Irani. Super-resolution from a single image. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 349–356. IEEE, 2009. 2

[10] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014. 2

[11] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 1, 3

[12] J.-B. Huang, A. Singh, and N. Ahuja. Single image super-resolution from transformed self-exemplars. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5197–5206, 2015. 2, 6

[13] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015. 1, 3

[14] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision*, pages 694–711. Springer, 2016. 2

[15] J. Kim, J. Kwon Lee, and K. Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1646–1654, 2016. 2, 3, 6

[16] J. Kim, J. Kwon Lee, and K. Mu Lee. Deeply-recursive convolutional network for image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1637–1645, 2016. 1, 2

[17] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014. 4

[18] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. *arXiv preprint arXiv:1609.04802*, 2016. 2, 3

[19] D. Liu, Z. Wang, N. Nasrabadi, and T. S. Huang. Learning a mixture of deep networks for single image super-resolution. In *ACCV*, pages 145–156, 2016. 2

[20] D. Liu, Z. Wang, B. Wen, J. Yang, W. Han, and T. S. Huang. Robust single image super-resolution via deep networks with sparse prior. *IEEE Transactions on Image Processing*, 25(7):3194–3207, 2016. 2

[21] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proc. 8th Int'l Conf. Computer Vision*, volume 2, pages 416–423, July 2001. 6

[22] A. Odena, V. Dumoulin, and C. Olah. Deconvolution and checkerboard artifacts. *Distill*, 2016. 3

[23] A. v. d. Oord, N. Kalchbrenner, L. Espeholt, O. Vinyals, A. Graves, et al. Conditional image generation with pixelcnn decoders. In *Advances in Neural Information Processing Systems*, pages 4790–4798, 2016. 3

[24] A. v. d. Oord, N. Kalchbrenner, and K. Kavukcuoglu. Pixel recurrent neural networks. *arXiv preprint arXiv:1601.06759*, 2016. 3

[25] M. S. M. Sajjadi, B. Schölkopf, and M. Hirsch. Enhancenet: Single image super-resolution through automated texture synthesis. *CoRR*, abs/1612.07919, 2016. 2, 3

[26] T. Salimans, A. Karpathy, X. Chen, and D. P. Kingma. Pixelcnn++: Improving the pixelcnn with discretized logistic mixture likelihood and other modifications. *arXiv preprint arXiv:1701.05517*, 2017. 3

[27] T. Salimans and D. P. Kingma. Weight normalization: A simple reparameterization to accelerate training of deep neural networks. In *Advances in Neural Information Processing Systems*, pages 901–901, 2016. 3

[28] J. Salvador and E. Pérez-Pellitero. Naive bayes super-resolution forest. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 325–333, 2015. 2

[29] S. Schulter, C. Leistner, and H. Bischof. Fast and accurate image upscaling with super-resolution forests. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3791–3799, 2015. 2

[30] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1874–1883, 2016. 2, 3

[31] C. K. Sønderby, J. Caballero, L. Theis, W. Shi, and F. Huszár. Amortised map inference for image super-resolution. *arXiv preprint arXiv:1610.04490*, 2016. 3

[32] J. Sun, Z. Xu, and H.-Y. Shum. Gradient profile prior and its applications in image super-resolution and enhancement. *IEEE Transactions on Image Processing*, 20(6):1529–1542, 2011. 2

[33] 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 *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, July 2017. 1, 4

[34] R. Timofte, V. De Smet, and L. Van Gool. Anchored neighborhood regression for fast example-based super-resolution. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1920–1927, 2013. 2

[35] R. Timofte, V. De Smet, and L. Van Gool. A+: Adjusted anchored neighborhood regression for fast super-resolution. In *Asian Conference on Computer Vision*, pages 111–126. Springer, 2014. 2, 6

[36] R. Timofte, R. Rothe, and L. Van Gool. Seven ways to improve example-based single image super resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1865–1873, 2016. 2

[37] Z. Wang, D. Liu, J. Yang, W. Han, and T. Huang. Deep networks for image super-resolution with sparse prior. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 370–378, 2015. 2

[38] Z. Wang, J. Yang, H. Zhang, Z. Wang, Y. Yang, D. Liu, and T. S. Huang. *Sparse Coding and its Applications in Computer Vision*. World Scientific, 2015. 2

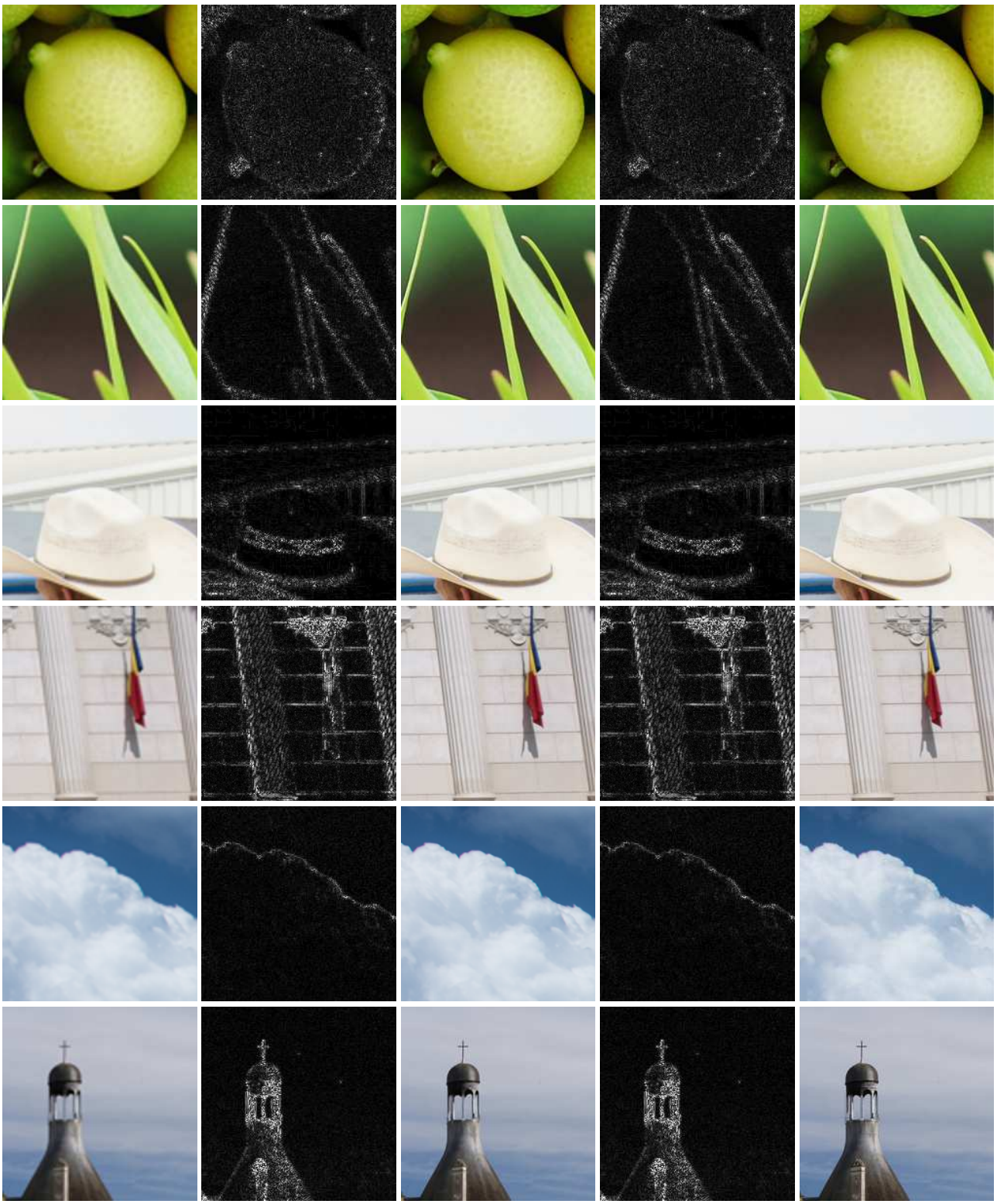
[39] Z. Wang, Y. Yang, Z. Wang, S. Chang, J. Yang, and T. S. Huang. Learning super-resolution jointly from external and internal examples. *IEEE Transactions on Image Processing*, 24(11):4359–4371, 2015. 2

[40] C.-Y. Yang, C. Ma, and M.-H. Yang. Single-image super-resolution: A benchmark. In *European Conference on Computer Vision*, pages 372–386. Springer, 2014. 2

[41] J. Yang, Z. Wang, Z. Lin, S. Cohen, and T. Huang. Coupled dictionary training for image super-resolution. *IEEE Transactions on Image Processing*, 21(8):3467–3478, 2012. 2

[42] J. Yang, J. Wright, T. S. Huang, and Y. Ma. Image super-resolution via sparse representation. *IEEE transactions on image processing*, 19(11):2861–2873, 2010. 2

[43] R. Zeyde, M. Elad, and M. Protter. On single image scale-up using sparse-representations. In *International conference on curves and surfaces*, pages 711–730. Springer, 2010. 6



(a) (b) (c) (d) (e)

Figure 5: Visual comparison results on 200×200 image patches: (a) output using bicubic interpolation, (b) absolute difference summed over rgb channels between bicubic interpolation's output and ground truth, (c) output using our proposal method, (d) absolute difference summed over rgb channels between our proposed method's output and ground truth, (e) ground truth. As can be seen from the visual results, our method produces much sharper results in general and significantly smoother results on image edges.