A Matrix-in-matrix Neural Network for Image Super Resolution

Hailong Ma  Xiangxiang Chu  Bo Zhang  Shaohua Wan  Bo Zhang
Xiaomi AI
Beijing, China
{mahailong, chuxiangxiang, zhangboll, wanshaohua, zhangbo}@xiaomi.com

Abstract

In recent years, deep learning methods have achieved impressive results with higher peak signal-to-noise ratio in single image super-resolution (SISR) tasks by utilizing deeper layers. However, their application is quite limited since they require high computing power. In addition, most of the existing methods rarely take full advantage of the intermediate features which are helpful for restoration. To address these issues, we propose a moderate-size SISR network named matrixed channel attention network (MCAN) by constructing a matrix ensemble of multi-connected channel attention blocks (MCAB). Several models of different sizes are released to meet various practical requirements. Conclusions can be drawn from our extensive benchmark experiments that the proposed models achieve better performance with much fewer multiply-adds and parameters. Our models will be made publicly available at this URL 1.

1. Introduction

Single image super-resolution (SISR) attempts to reconstruct a high-resolution (HR) image from its low-resolution (LR) equivalent, which is essentially an ill-posed inverse problem since there are infinitely many HR images that can be downsampled to the same LR image.

Most of the works discussing SISR based on deep learning have been devoted to achieving higher peak signal noise ratios (PSNR) with deeper and deeper layers, making it difficult to fit in mobile devices [20, 21, 32, 41]. Out of many proposals, an architecture CARN has been released that is applicable in the mobile scenario, but it is at the cost of reduction on PSNR [1]. An information distillation network (IDN) proposed in [18] also achieves good performance at a moderate size. An effort that tackles SISR with neural architecture search has also been proposed [5, 6], their network FALSR surpasses CARN at the same level of FLOPS.

Still, there is a noticeable gap between subjective perception and PSNR, for which a new measure called perceptual index (PI) has been formulated [3]. Noteworthy works engaging perceptual performance are SRGAN [24] and ESRGAN [36], which behave poorly on PSNR but both render more high-frequency details. However, these GAN-based methods inevitably bring about bad cases that are intolerable in practice. Our work still focuses on improving PSNR, which is a well-established distortion measure. Furthermore, our proposed model can also serve as the generator of GAN-based methods.

To seek a better trade-off between performance and applicability, we design an architecture called Matrixed Channel Attention Network. We name its basic building block multi-connected channel attention block (MCAB), which is an adaptation of residual channel attention block (RCAB) from RCAN [41]. MCAB differs from RCAB by allowing hierarchical connections after each activation, in such way, multiple levels of information can be passed both in depth and in breadth.

In summary, our main contributions are as follows:

• We propose a matrixed channel attention network named MCAN for SISR. It scores higher PSNR and achieves the state-of-the-art results within a lightweight range.

• We introduce a multi-connected channel attention block to construct matrixed channel attention cell (MCAC), which makes full use of the hierarchical features. Then we use MCAC to build a matrix-in-matrix (MIM) structure that serves as a nonlinear mapping module.

• We devise an edge feature fusion (EFF) block, which can be used in combination with the proposed MIM structure. The EFF can better profit the hierarchical features of MIM from the LR space.

• We build three additional efficient SR models of different sizes, i.e., MCAN-M, MCAN-S, MCAN-T, which are respectively short for mobile, small, and tiny. Ex-

1https://github.com/macn3388/MCAN
experiments prove that all three models outperform the state-of-the-art models of the same or bigger sizes.

- We finally present MCAN-FAST to overcome the inefficiency of the sigmoid function on some mobile devices. Experiments show that MCAN-FAST has only a small loss of precision compared to MCAN.

2. Related Works

In recent years, deep learning has been applied to many areas of computer vision [11, 14, 26, 30, 39]. A pioneering work [7] has brought super-resolution into deep learning era, in which they proposed a simple three-layer convolutional neural network called SRCNN, where each layer sequentially deals with feature extraction, non-linear mapping, and reconstruction. The input of SRCNN, however, needs an extra bicubic interpolation which reduces high-frequency information and adds extra computation. Their later work FSRCNN [8] requires no interpolation and inserts a deconvolution layer for reconstruction, which learns an end-to-end mapping. Besides, shrinking and expanding layers are introduced to speed up computation, altogether rendering FSRCNN real-time on a generic CPU.

Meantime, VDSR presented by [21] features a global residual learning to ease training for their very deep network. DRCN handles deep network recursively to share parameters [21]. DRRN builds two residual blocks in a recursive manner [32]. They all bear the aforementioned problem caused by interpolation. Furthermore, these very deep architectures undoubtedly require heavy computation.

The application of DenseNet in SR domain goes to SR-DenseNet [35], in which they argue that dense skip connections mitigate the vanishing gradient problem and can boost feature propagation. It achieves better performance as well as faster speed. However, results from [5] showed that dense connection might not be the most efficient and their less dense network FALSR is also competitive.

Later, a cascading residual network CARN is devised for a lightweight scenario [1]. The basic block of their architecture is called cascading residual block, whose outputs of intermediary layers are dispatched to each of the consequent convolutional layers. These cascading blocks, when stacked, are again organized in the same fashion.

There is another remarkable work RCAN [41], which has a great impact on our work. They have observed that low-frequency information is hard to capture by convolutional layers which only exploit a local region. By adding multiple long and short skip connections for residual dense blocks, low-frequency features can bypass the network and thus the main architecture focuses on high-frequency information. They also invented a channel attention mechanism via global average pooling to deal with interdependencies among channels.

3. Matrixed Channel Attention Network

3.1. Network Structure

The MCAN consists of four components: feature extraction (FE), matrix in matrix (MIM) mapping, edge feature fusion (EFF) and reconstruction, as illustrated in Figure 2.

Specifically, we utilize two successive $3 \times 3$ convolutions to extract features from the input image in the FE stage. Let $I_{LR}$ represent the input image and $I_{HR}$ be the output, this procedure can then be formulated as

$$F_0 = H_{FE}(I_{LR})$$

where $H_{FE} (\cdot)$ is the feature extraction function and $F_0$ denotes the output features.

The nonlinear mapping is constructed by what we call a matrix-in-matrix module (MIM). Similarly,

$$F_{EF} = H_{MIM}(F_0)$$
where $H_{MIM}(\cdot)$ is the mapping function, to be discussed in detail in Section 3.2. $F_{EF}$ stands for the edge features, so named for they are coming from the edge of our matrix block. Further feature fusion can be put formally as,

$$F_{EFF} = H_{EFF}(F_{EF}),$$

(3)

We will elaborate on EFF in Section 3.4.

Lastly, we upscale the combination of fused feature $F_{EFF}$ and $F_0$ to generate the high-resolution target via reconstruction,

$$I_{SR} = H_{UP}(F_{EFF} + F_0) + U(I_{LR}),$$

(4)

where $H_{UP}(\cdot)$ denotes the upsampling function and $U(\cdot)$ the bilinear interpolation.

### 3.2. Matrix in Matrix

The MIM block contains $D = 3$ matrixed channel attention cells (MCAC). A single MCAC is again a sequence of $M = 3$ MCABs. The $d$-th MCAC relays intermediate features to the next MCAC. In fact, each MCAC contains $K = 3$ heads, which are fed into different parts of the next MCAC. We recursively define $F_d$ as the outputs of a MCAC,

$$F_d = H_{MCAC}^d(F_{d-1}) = H_{MCAC}^d((F_1^{d-1}, F_2^{d-1}, \ldots, F_K^{d-1}))$$

(5)

Hence the output of $H_{MIM}(\cdot)$ can be composed by the combination of $K$-th outputs of all MCACs,

$$F_{EF} = (F_1^K, F_2^K, \ldots, F_D^K).$$

(6)

Therefore, we can regard MIM as a $D \times K$ matrix. If we look at its structural detail, a MCAC can again be seen as a matrix of $K \times M$, for this reason we call it matrix-in-matrix. The overall structure of MIM is shown in Figure 2.

### 3.3. Matrixed Channel Attention Cell

In super-resolution, skip connections are popular since it reuses intermediate features while relieving the training for deep networks [21, 27, 35]. Nevertheless, these skip connections between modules are point-to-point, where only the output features of a module can be reused, losing many
intermediate features. This can be alleviated by adding skip connections within the module, but as more intermediate features are concatenated, channels become very thick before fusion [42], which narrows transmission of information and gradients.

If we densely connect all intermediate features and modules like the SRDenseNet [35], it inevitably brings in redundant connections for less important features, while the important ones become indistinguishable and increases the training difficulty.

To address these problems, we propose a matrixed channel attention cell, which is composed of several multi-connected channel attention blocks.

### 3.3.1 Multi-connected Channel Attention Block

Previous works seldom discriminate feature channels and treat them equally. Till recently a channel attention mechanism using global pooling is proposed in RCAN to concentrate on more useful channels [41]. We adopt the same mechanism using global pooling is proposed in RCAN to concentrate on more useful channels [41]. We adopt the same mechanism using global pooling is proposed in RCAN to concentrate on more useful channels [41]. We adopt the same mechanism using global pooling is proposed in RCAN to concentrate on more useful channels [41]. We adopt the same mechanism using global pooling is proposed in RCAN to concentrate on more useful channels [41]. We adopt the same mechanism using global pooling is proposed in RCAN to concentrate on more useful channels [41]. We adopt the same mechanism using global pooling is proposed in RCAN to concentrate on more useful channels [41]. We adopt the same mechanism using global pooling is proposed in RCAN to concentrate on more useful channels [41].

**Channel Attention Mechanism.** We let $X = [x_1, \cdots, x_c, \cdots, x_C]$ denote an input that contains $C$ feature maps, and the shape of each feature map be $H \times W$. Then the statistic $z_c$ of $c$-th feature map $x_c$ is defined as

$$z_c = H_{GP}(x_c) = \frac{\sum_{i=1}^{H} \sum_{j=1}^{W} x_c(i,j)}{H \times W}, \quad (7)$$

where $x_c(i,j)$ denotes the value at index $(i,j)$ of feature map $x_c$, and $H_{GP}(\cdot)$ represents the global average pooling function. The channel attention of the feature map $x_c$ can thus be denoted as

$$s_c = f(W_D \delta(W_D z_c)), \quad (8)$$

where $f(\cdot)$ and $\delta(\cdot)$ represent the sigmoid function and the ReLU [29] function respectively, $W_D$ is the weight set of a $1 \times 1$ convolution for channel downscaling. This convolution reduces the number of channels by a factor $r$. Later after being activated by a ReLU function, it enters a $1 \times 1$ convolution for channel upscaling with the weights $W_U$, which expands the channel again by the factor $r$. The computed statistic $s_c$ is used to rescale the input features $x_c$.

$$\hat{x}_c = s_c \cdot x_c, \quad (9)$$

**Description of RCAB.** The RCAB is organized using the aforementioned channel attention mechanism. Formally it can be seen as a function $H_{RCAB}(\cdot)$ on the input features $I$,

$$F_{RCAB} = H_{RCAB}(I) = s_{X_I} \cdot X_I + I \quad (10)$$

where $s_{X_I}$ is the output of channel attention on $X_I$, which are the features generated from the two stacked convolution layers,

$$X_I = W_2 \delta(W_1 I). \quad (11)$$

The cascading mechanism from CARN [1] makes use of intermediate features in a dense way. In order to relax the redundancy of dense skip connections, our residual channel attention blocks are built in a multi-connected fashion, so called as MCAB, as shown in Figure 2. Each MCAB contains $M$ residual channel attention blocks (RCAB) and $M+1$ pointwise convolution operations for feature fusion ($H_F$), which are interleaved one after another.

### 3.3.2 MCAC Structure

The structure of MCAC can be considered as a matrix of $K = 3$ MCABs times $M = 3$ RCABs. In the $d$-th MCAC, we let the input and output of $k$-th MCAB be $IM^k_d$ and $OM^k_d$, and the $k$-th output of the last MCAB be $F^k_d$, we formulate $IM^k_d$ as follows,

$$\begin{align}
F^k_0 & \\
OM^k_1 & \\
F^k_d & \quad d = k = 0 \\
OM^k_{d-1} & \quad d = 0, k \in [0, K] \\
F^k_d & \quad d \in (0, D), k = 0 \\
OM^k_{d-1} & \quad d \in (0, D), k \in [0, K] 
\end{align} \quad (12)$$

In the case of $d \in (0, D), k \in [0, K], m \in [0, M]$, the $m$-th feature fusion convolution $H_F$ takes multiple inputs and fuses them into $F^k_d$, m as. Let the input of $m$-th RCAB be $IR^k_d,m$ and the output $OR^k_d,m$, we can write the input of $m$-th feature fusion convolution $IF^k_d,m$ as,

$$\begin{align}
\begin{cases}
F^k_{d-1}, m + 1 \quad m = 0 \\
OR^k_{d-1}, m + 1, F^k_{d-1}, m + 1 \quad m \in (0, M) \\
\vdots \\
OR^k_{d-1}, M, F^k_{d-1}, M \quad m = M + 1
\end{cases} \quad (13)
\end{align}$$

Now we give the complete definition of the output of $d$-th MCAC,

$$\begin{align}
F_d & = (F^1_d, \ldots, F^K_d) \\
& = H_{MCAC_d}(F_d) \\
& = H_{MCAC_d}(\{F^1_d, F^2_d, \ldots, F^K_d\}) \quad (14)
\end{align}$$

As mentioned above, the nonlinear mapping module of our proposed model can be seen as a matrix of $D \times K$. Thus its overall number of sigmoid functions can be calculated as,

$$\text{num}_f(\cdot) = D \times K \times M \times N_{ca},\quad (15)$$

where $f(\cdot)$ means the sigmoid function and $N_{ca}$ indicates the number of filters in the channel attention mechanism.
As we generate multiple features through MIM during different stage, we put forward an edge feature fusion (EFF) module to integrate these features hierarchically.

Table 1. Quantitative comparison with the state-of-the-art methods based on ×2, ×3, ×4 SR with bicubic degradation model. Red/blue text: best/second-best.

3.4. Edge Feature Fusion

Particularly, we unite the outputs of the last MCAB in each MCAC, which are nicknamed as the edge of the MIM
structure. In further detail, EFF takes a $3 \times 3$ convolution for fusion and another $3 \times 3$ convolution to reduce channel numbers.

\[
F_{EFF} = W_F W_R [F_d^{1,M+1}, \ldots, F_d^{K,M+1}],
\]

where $W_F$ and $W_R$ are the weights of fusion convolution and the channel reduction layer.

### 3.5. Comparison with Recent Models

**Comparison with SRDenseNet.** SRDenseNet uses dense block proposed by DenseNet to construct nonlinear mapping module [35]. This dense connection mechanism may lead to redundancy, in fact not all features should be equally treated. In our work, MIM and EFF can reduce dense connections and highlight the hierarchical information. Additionally, SRDenseNet connects two blocks from point to point, which refrains transmission and utilization of intermediate features. Our proposed multi-connected channel attention block (MCAB) mitigates this problem by injecting multiple connections between blocks.

**Comparison with CARN.** CARN uses a cascading block [1], which is also pictured in our MIM. Despite of this, MIM features multiple connections between MCACs, and the outputs of different stages are relayed between MCABs. Such an arrangement makes better use of intermediate information. Another important difference is that MCAN combines the hierarchical features before upsampling via edge feature fusion. This mechanism helps significantly for reconstruction.
4. Experimental Results

4.1. Datasets and Evaluation Metrics

We train our model based on DIV2K [34], which contains 800 2K high-resolution images for the training set and another 100 pictures for both validation and test set. Besides, we make comparisons across three scaling tasks (×2, ×3, ×4) on four datasets: Set5 [2], Set14 [38], B100 [28], and Urban100 [17]. The evaluation metrics we used are PSNR [15] and SSIM [37] on the Y channel in the YCbCr space.

4.2. Implementation Details

As shown in Figure 2, the inputs and outputs of our model are RGB images. We crop the LR patches by 64 × 64 for various scale tasks and adopt the standard data augmentation.

For training, we use Adam (β1 = 0.9, β2 = 0.999, and ε = 10⁻⁸) [22] to minimize $L_1$ loss within $1.2 \times 10^6$ steps with a batch-size of 64. The initial learning rate is set to $2 \times 10^{-4}$, halved every $4 \times 10^5$ steps. Like CARN [1], we also initialize the network parameters by $θ \sim U(-k, k)$, where $k = 1/\sqrt{cin}$ and $cin$ is the number of input feature maps. Inspired by EDSR [25], we apply a multi-scale training. Our sub-pixel convolution is the same as in ESPCN [31].

We choose network hyperparameters to build an accurate and efficient model. The first two layers in the FE stage contain $n_{FE} = \{64, 32\}$ filters accordingly. As for MIM, we set $D = K = M = 3$, its number of filters $n_{MIM} = 32$. Two EFF convolutions have $n_{EFF} = \{D \times 32, 32\}$ filters. The last convolution before the upsampling procedure has $n_l = 256$ filters. The reduction factor $r$ in channel attention mechanism is set to 8.

Since the sigmoid function is inefficient on some mobile devices, especially for some fixed point units such as DSP. Therefore we propose MCAN-FAST by replacing the sigmoid with the fast sigmoid [10], which can be written as,

$$f_{fast}(x) = \frac{x}{1 + |x|}. \quad (17)$$

Experiments show that MCAN-FAST has only a small loss on precision, and it achieves almost the same level of metrics as MCAN.

| Models       | n_{FE} | n_{MIM} | n_{EFF} | n_l | r  |
|--------------|--------|---------|---------|-----|----|
| MCAN         | 64,32  | 32      | 96,32   | 256 | 8  |
| MCAN-M       | 64,24  | 24      | 72,24   | 128 | 8  |
| MCAN-S       | 32,16  | 16      | 48,16   | 64  | 8  |
| MCAN-T       | 16,8   | 8       | 24,8    | 8   | 4  |

Table 2. Network hyperparameters of our networks.

For more lightweight applications, we reduce the number of filters as shown in Table 2. Note in MCAN-T we also set the group as 4 in the group convolution of RCAB for further compression.

4.3. Comparisons with State-of-the-art Algorithms

We use Mult-Adds and the number of parameters to measure the model size. We emphasize on Mult-Adds as it indicates the number of multiply-accumulate operations. By convention, it is normalized on 1280 × 720 high-resolution images. Further evaluation based on geometric self-ensembling strategy [41] are marked with ‘+’.

Quantitative comparisons with the state-of-the-art methods are listed in Table 1. For fair comparison, we concentrate on models with comparative multi-adds and parameters.

Notably, MCAN outperforms CARN [1] with fewer multi-adds and parameters. The medium-size model MCAN-M [1] achieved better performance than the CARN-M, additionally, it is still on par with CARN with about half of its multi-adds. For short, it surpasses all other listed methods, including MoreMNAS-A [6] and FALSR-A [5] from NAS methods.

The smaller model MCAN-S emulates LapSRN [23] with much fewer parameters. Particularly, it has an average advantage of 0.5 dB on PSNR over the LapSRN on the ×2 task, and on average, MCAN-S still has an advantage of 0.4 dB. MCAN-S also behaves better than CARN-M on all tasks with half of its model size. It is worth to note that heavily compressed MCAN-S still exceeds or matches larger models such as VDSR, DRCN, DRRN and MemNet.

The tiny model MCAN-T is meant to be applied under requirements of extreme fast speed. It overtakes FSRCNN [8] on all tasks with the same level of mult-adds.
Table 3. Investigations of MIM and EFF. We record the best average PSNR(dB) values of Set5 & Set14 on ×4 SR task in 10^5 steps.

|          | MIM | EFF | Avg. PSNR |
|----------|-----|-----|-----------|
|          | ✓   | ✓   | 29.44     |
|          | ✓   | ✓   | 30.25     |
|          | ✓   | ✓   | 30.23     |
|          | ✓   | ✓   | 30.28     |

4.4. Ablation Study

In this section, we demonstrate the effectiveness of the MIM structure and EFF through ablation study.

Matrix in matrix. We remove the connections between MCACs and also the connections between MCABs. Hence the model comes without intermediate connections. As shown in Table 3, the MIM structure can bring significant improvements, PSNR improves from 29.44 dB to 30.25 dB when such connections are enabled. When EFF is added, PNSR continues to increase from 30.23 dB to 30.28 dB.

Edge feature fusion. We simply eliminate the fusion convolutions connected to MIM and consider the output of the last MCAB as the output of MIM. In this case, the intermediate features acquired by the MIM structure are not directly involved in the reconstruction. In Table 3, we observe that the EFF structure enhances PNSR from 29.44 dB to 30.23 dB. With MIM enabled, PSNR is further promoted from 30.25 dB to 30.28 dB.

5. Conclusion

In this paper, we proposed an accurate and efficient network with matrixed channel attention for the SISR task. Our main idea is to exploit the intermediate features hierarchically through multi-connected channel attention blocks. MCAB then acts as a basic unit that builds up the matrix-in-matrix module. We release three additional efficient models of varied sizes, MCAN-M, MCAN-S, and MCAN-T. Extensive experiments reveal that our MCAN family excel the state-of-the-art models of accordingly similar sizes or even much larger.

To deal with the inefficiency of the sigmoid function on some mobile devices, we benefit from the fast sigmoid to construct MCAN-FAST. The result confirms that MCAN-FAST has only a small loss of precision when compared to MCAN, and it can still achieve better performance with fewer multi-adds and parameters than the state-of-the-art methods.

References

[1] N. Ahn, B. Kang, and K.-A. Sohn. Fast, accurate, and, lightweight super-resolution with cascading residual network. arXiv preprint arXiv:1803.08664, 2018.

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

[3] Y. Blau, R. Mechrez, R. Timofte, T. Michaeli, and L. Zelnik-Manor. The 2018 pirm challenge on perceptual image super-resolution. In European Conference on Computer Vision, pages 334–355. Springer, 2018.

[4] J.-S. Choi and M. Kim. A deep convolutional neural network with selection units for super-resolution. In Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on, pages 1150–1156. IEEE, 2017.

[5] X. Chu, B. Zhang, H. Ma, R. Xu, J. Li, and Q. Li. Fast, accurate and lightweight super-resolution with neural architecture search. arXiv preprint arXiv:1901.07261, 2019.

[6] X. Chu, B. Zhang, R. Xu, and H. Ma. Multi-objective reinforced evolution in mobile neural architecture search. arXiv preprint arXiv:1901.01074, 2019.

[7] 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.

[8] 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.

[9] Y. Fan, H. Shi, J. Yu, D. Liu, W. Han, H. Yu, Z. Wang, X. Wang, and T. S. Huang. Balanced two-stage residual networks for image super-resolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pages 161–168, 2017.

[10] G. Georgiou. Parallel Distributed Processing in the Complex Domain. PhD thesis, Tulane, 1992.

[11] R. Girshick. Fast r-cnn. In Proceedings of the IEEE international conference on computer vision, pages 1440–1448, 2015.

[12] 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.

[13] S. Han, H. Mao, and W. J. Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. arXiv preprint arXiv:1510.00149, 2015.

[14] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In Proceedings of the IEEE international conference on computer vision, pages 2961–2969, 2017.

[15] A. Hore and D. Zhou. Image quality metrics: Psnr vs. ssim. In 2010 20th International Conference on Pattern Recognition, pages 2366–2369. IEEE, 2010.

[16] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861, 2017.

[17] 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.
[18] Z. Hui, X. Wang, and X. Gao. Fast and accurate single image super-resolution via information distillation network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 723–731, 2018.

[19] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size. arXiv preprint arXiv:1602.07360, 2016.

[20] 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.

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

[22] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In IEEE Conference on Computer Vision and Pattern Recognition, volume 2, page 5, 2017.

[23] 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, 2017.

[24] B. Lim, S. Son, H. Kim, S. Nah, and K. Mu Lee. Enhanced deep residual networks for single image super-resolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pages 136–144, 2017.

[25] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector. In European conference on computer vision, pages 21–37. Springer, 2016.

[26] X. Mao, C. Shen, and Y.-B. Yang. Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections. In Advances in neural information processing systems, pages 2802–2810, 2016.

[27] 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 null, page 416. IEEE, 2001.

[28] Y. Tai, J. Yang, and X. Liu. Image super-resolution via deep recursive residual network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, volume 1, page 5, 2017.

[29] Y. Tai, J. Yang, X. Liu, and C. Xu. Memnet: A persistent memory network for image restoration. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 4539–4547, 2017.

[30] R. Timofte, E. Agustsson, L. Van Gool, M.-H. Yang, and L. Zhang. Ntire 2017 challenge on single image super-resolution: Methods and results. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pages 114–125, 2017.

[31] T. Tong, G. Li, X. Liu, and Q. Gao. Image super-resolution using dense skip connections. In Proceedings of the IEEE International Conference on Computer Vision, pages 4799–4807, 2017.

[32] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, Y. Qiao, and C. C. Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In European conference on Computer Vision, pages 63–79. Springer, 2018.

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

[34] 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.

[35] R. Zhang, P. Isola, and A. A. Efros. Colorful image colorization. In European conference on computer vision, pages 649–666. Springer, 2016.

[36] X. Zhang, X. Zhou, M. Lin, and J. Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 6848–6856, 2018.

[37] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu. Image super-resolution using very deep residual channel attention networks. In European Conference on Computer Vision, pages 294–310. Springer, 2018.

[38] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu. Residual dense network for image super-resolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 2472–2481, 2018.