Wide Receptive Field and Channel Attention Network for JPEG Compressed Image Deblurring

Donghyeon Lee*, Chulhee Lee*
Samsung Electronics
Hwaseong, Korea
donghyeon1223@gmail.com,
bbfreezer@gmail.com

Taesung Kim‡
Sunmoon University
Asan, Korea
ts7kim@sunmoon.ac.kr

Abstract

A motion blurred image stored in the joint photographic experts group (JPEG) image compression format contains both motion blur and JPEG artifacts. Therefore, it is very difficult to restore the original image from a blurred and JPEG-compressed image. To address this problem, this paper proposes two methods: a wide receptive field and channel attention network (WRCAN), and JPEG auto-encoder loss. First, the WRCAN utilizes a large receptive field and considers the interdependencies among channels of a feature map. Second, the proposed JPEG auto-encoder loss enables the WRCAN to learn prior knowledge of JPEG compression artifacts such that the proposed WRCAN can effectively restore the original image from JPEG-compressed images. The proposed methods are evaluated on the JPEG-compressed REDS dataset by participating in the NTIRE 2021 workshop challenges on Image Deblurring Track 2 JPEG artifacts. The WRCAN trained with the proposed loss ranked third with an output of 29.60dB on the REDS test set, indicating that the proposed methods provide state-of-the-art results. The source codes, model, and data are available at https://github.com/dhyeonlee/WRCAN-PyTorch.

1. Introduction

The target of single image deblurring is to reconstruct an image that contains detailed information of the original image from a blurred image. On the other hand, the joint photographic experts group (JPEG) image restoration aims to remove visual artifacts from a JPEG compressed image. These two problems are often combined in many cases because photographs captured with a camera are typically stored in JPEG format to reduce network traffic and storage. The JPEG compression algorithm partitions an image into 8×8 non-overlapping blocks and performs a discrete cosine transform (DCT) for each block, where each transformed coefficient is quantized to be effectively compressed via entropy coding. Hence, many high-frequency components in the image are lost by quantization, thereby resulting in ringing artifacts. In addition, pixel discontinuities, often referred to as blocking artifacts, occur at boundaries of the 8 × 8 pixel blocks because the transformation and quantization are performed on 8×8 pixel non-overlapping blocks. As a result, the image blurred and stored in JPEG format contains motion blurs and JPEG compression artifacts. Therefore, it is difficult to restore the original image from a blurred and JPEG-compressed image.

Recent studies regarding JPEG image restoration and deblurring are based on the convolutional neural network (CNN) owing to their superior performance. Previous studies in [1, 2, 3] demonstrate that the wide receptive field of CNNs result in better image restoration and deblurring because more information can be utilized by referring to larger image areas. However, these approaches do not focus on important channels in the feature maps. In other studies, the channel attention mechanism is employed to emphasize the informative channel of an input feature map [4, 5, 6]. The channel attention mechanism enables a deep learning model to focus on important features to improve performance. However, in the abovementioned studies, a large receptive field size is not considered when emphasizing the important channels of a feature map in a single basic block. For JPEG image restoration, prior knowledge pertaining to JPEG compression is used in recent neural network architectures, which comprises both pixel and DCT domains to account for the characteristics of the DCT in JPEG format [7, 8]. However, image deblurring cannot be integrated directly in these studies, because the general image deblurring framework differs from the JPEG image restoration framework.

In this paper, a new CNN architecture for JPEG compressed image deblurring is proposed. In addition, a new loss function for JPEG artifact reduction is proposed. The contributions of this study are as follows.

- This paper proposes the wide receptive field and channel attention network (WRCAN). The basic block of the WRCAN enables both a large local receptive

* denotes equally contributed authors, and ‡ denotes the corresponding author.

field and a channel attention mechanism to be utilized.

- JPEG auto-encoder loss ($L_{AE-JPEG}$) is proposed to enable the WRCAN effectively to learn the prior knowledge of JPEG compression artifacts without modifying the network. Hence, the WRCAN need not perform any additional computation for JPEG artifact reduction at the inference time.
- This study demonstrates that the WRCAN combined with the proposed JPEG auto-encoder loss can effectively solve the JPEG-compressed deblurring problem.

The proposed methods are evaluated on the JPEG-compressed REDS dataset by participating in the NTIRE 2021 workshop challenges on Image Deblurring Track 2 JPEG artifacts [9]. The evaluation results show that the WRCAN and L_{AE-JP} yield state-of-the-art results.

2. Related Works

2.1. Single Image Deblurring

Early studies pertaining to image deblurring focus on estimating blur kernels, which are modeled as uniform or non-uniform blurs. In studies focusing on uniform deblurring [10,11,12,13,14,15], it is assumed that a blurred image is convoluted with an unknown blur kernel, and the problem is defined as an optimization problem, the purpose of which is to accurately estimate the blur kernel. Fergus *et al.* [10] assume a camera movement as a single blur kernel, and the kernel is estimated using sparse gradients. Sun *et al.* [11] estimate a blur kernel using edge patches learned from training images. Michaeli *et al.* [12] use internal patch recurrence to estimate a blur kernel. Although these methods are effective for natural image deblurring, they are not effective for non-natural images or in low-light conditions. Hence, domain-specific priors are used for estimating a blur kernel such as deblurring low-light images [13], text images [14], and face images [15]. However, because these are domain-specific, their application scope is limited. In general, image deblurring cannot be estimated as a single blur kernel. Therefore, deblurring images based on assuming the non-uniform blurring is investigated in [16, 17]. Kim *et al.* [16] propose a deblurring framework by estimating a pixel-wise blur kernel and a latent image for dynamic scene deblurring. Bahat *et al.* [17] estimate the blur fields of an input image by analyzing the spectral contents of blurry image patches and achieve performance similar to those of CNN-based approaches.

Recent studies of image deblurring are based on CNNs [3, 5, 18, 19, 20, 21, 22, 24]. Nah *et al.* [18] propose a multi-scale network architecture, in which deblurring is performed in a coarse-to-fine manner. The CNN architecture proposed by Tao *et al.* [19] uses images of different scales and propagates the result of a lower-scale network to a higher-scale network. Zhang *et al.* [20] present a hierarchical CNN inspired by spatial pyramid matching, in which deblurring is performed in fine-to-coarse grids. Brehm *et al.* [3] propose a residual block with atrous convolution, where the residual block contains four parallel paths of atrous convolution with different local receptive fields. Kaufman *et al.* [21] present analysis and synthesis networks for image deblurring, where the former estimates a two-dimensional blur kernel, and the latter creates a deblurred image using the estimated kernel and an input image. Qi *et al.* [5] propose a feature fusion block that consists of a channel attention module and a pixel attention module. Kupyn *et al.* [22] apply an adversarial loss for image deblurring, where a feature pyramid network is used as a generator and a relativistic discriminator [23] is adopted as a discriminator. Zhang *et al.* [24] present a combination of two generative adversarial network (GAN) models, i.e., the blur GAN and deblur GAN, where the former learns image blurring by generating and discriminating fake blur images, whereas the latter learns image deblurring by creating fake sharp images from the fake blur images.

2.2. JPEG artifact reduction

Early works for JPEG image restoration use deblocking filters to reduce discontinuities between non-overlapped 8×8 pixel blocks [25, 26]. To reduce blocking artifacts in JPEG-compressed images, Lee *et al.* [25] adaptively use various block predictors based on the frequency component in the DCT domain. Yoo *et al.* [26] classify blocks into flat or edge blocks and apply different deblocking filters based on the classification results. However, because JPEG-compressed images contain not only blocking artifacts, but also other artifacts such as ringing artifacts, the image restoration performance of the deblocking filter is not good.

Meanwhile, data-driven learnings are investigated for general image restoration [27, 28, 29]. First, dictionaries are learned from training image data. Second, an uncompressed image is reconstructed using the sparse representation of the learned dictionaries. Choi *et al.* [27] train different dictionaries based on the characteristics of the training images, and optimal dictionaries are automatically selected for JPEG artifact reduction. The method proposed by Rothe *et al.* [28] learns linear regressors from training data and regresses a test image to an artifact-free image by selecting learned regressors at the nearest anchoring points. Liu *et al.* [29] propose a dual-domain sparsity-based image restoration, where dictionaries are learned jointly in the DCT and pixel domains.

Extensive research has been conducted regarding JPEG image restoration based on CNNs because they outperform previous approaches [1, 2, 5, 7, 8, 30, 32, 33, 34]. Dong *et al.* [30] introduce a CNN that is a modified version of the super-resolution CNN for JPEG artifact reduction. Lie *et al.*

[1] utilize a large receptive field with reduced computational complexity by applying a wavelet transform to the U-Net architecture. Fu *et al.* [2] propose deep convolutional sparse coding architecture with atrous convolution [31] to obtain a high-level receptive field. Tai *et al.* [32] emphasize the importance of memorizing previous features and increase the depth of a CNN using dense connections. Zhang *et al.* [33] propose a residual dense network for the more effective usage of hierarchical features from an input image and utilize local features via densely connected local layers. Zhang *et al.* [5] present a residual non-local network which consists of residual local and non-local attention blocks. Galteri *et al.* [34] propose a fully convolutional residual network trained using a generative adversarial framework. Meanwhile, prior knowledge regarding JPEG compression is adopted for CNNs [7, 8]. Wang *et al.* [7] propose a dual-domain model, in which an input image is first processed in a DCT domain followed by a pixel domain. Zhang *et al.* [8] use auto-encoders in both the DCT and pixel domains, and the outputs of auto-encoders and input images are considered for artifact reduction.

3. Proposed WRCAN and JPEG auto-encoder loss

This section describes the proposed WRCAN, which not only utilizes a large receptive field, but also adaptively weights each channel of the feature map. The proposed JPEG auto-encoder loss function, which enables the model to extract and learn the characteristics of a JPEG compressed image, is described in this section as well.

3.1. Wide receptive field and channel attention network (WRCAN)

Overall Architecture

The base architecture of the proposed WRCAN follows that of the U-Net [35]. Therefore, the WRCAN utilizes a significant number of parameters with a reduced number of computations thanks to the pooling operation. Moreover, the WRCAN can precisely localize the output image by capturing the context of a high-resolution input image through a skip-connection. Figure 1 shows the proposed WRCAN. The WRCAN receives a blurred image (I^B) and outputs a sharpened image (I^S). Here, $Conv(C_{in}, C_{out}, s = 1, d = 1)$ represents a convolution layer that receives a feature map with C_{in} channels and outputs a C_{out}-channel feature map. The s and d denote the size of stride and atrous rate [36] of the convolution layer, respectively. The s and d values are 1, unless explicitly specified. The $Encoder_l(C_{in}, C_{out})$ and $Decoder_l(C_{in}, C_{out})$ represent the encoder and decoder layers, respectively, with C_{in} input channels and C_{out} output channels, respectively. Here, l represents the encoding level of the input feature map, which increases as a feature map passes an encoder block. $RG_n(C_{in})$ represents the residual group (RG) block, where n is its index. It is noteworthy that the RGs have the same number of input and output feature map channels.

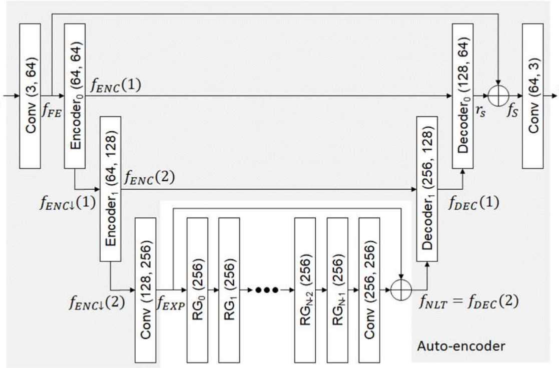


Figure 1. The proposed architecture of wide receptive field and channel attention network. Here, ⊕ represents element-wise addition of two feature maps.

The WRCAN uses one convolution layer for the feature extraction, i.e., $Conv(3, 64)$. Equation (1) expresses the feature extraction operation.

$$f_{FE} = Conv(3, 64) \circ I^B \tag{1}$$

In Equation (1), ∘ represents the operation between an operand (I^B in Equation (1)) and an operator ($Conv(3, 64)$ in Equation (1)). Hereinafter, this is also used for other operands and operators as well. Meanwhile, f_{FE}, which is the feature extracted from I^B , passes through two consecutive encoder blocks.

Figure 2(a) shows the architecture of the encoder. The encoder receives the input feature map $f_{ENC}(l)$ and outputs two types of feature maps, i.e., $f_{ENC}(l + 1)$ and $f_{ENC\downarrow}(l + 1)$. Equations (2.1) and (2.2) show the operations of the encoder. Hereinafter, the C_{in} and C_{out} are omitted from the equations and are replaced with $\cdot$ for simplicity. The s in the $Conv(\cdot)$ denotes the stride size of the convolution layer, and ReLU represents a rectified linear unit [37].

$$f_{ENC}(l + 1) = ReLU \circ Conv(\cdot) \circ f_{ENC}(l) \tag{2.1}$$
$$f_{ENC\downarrow}(l + 1) = Conv(\cdot, s = 2) \circ f_{ENC}(l + 1) \tag{2.2}$$

In Equation (2.1), $f_{ENC}(l + 1)$ is generated by the convolution and nonlinear transformation of $f_{ENC}(l)$. In Equation (2.2), $f_{ENC\downarrow}(l + 1)$ is generated by a strided convolution with $f_{ENC}(l + 1)$. $f_{ENC}(l + 1)$ and $f_{ENC\downarrow}(l + 1)$ are passed to a decoder with the same encoding level and the next encoder, respectively.

After f_{FE} passes two encoders, $f_{ENC}(2)$ and $f_{ENC\downarrow}(2)$ are generated. $f_{ENC}(2)$ is passed to $Decoder_1$, and

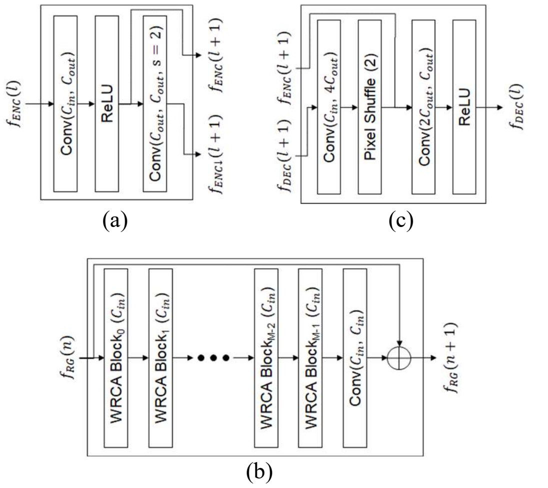

$f_{ENC}(l)$
Conv(C_{in}, C_{out})
ReLU
Conv(C_{out}, C_{out}, s = 2)
$f_{ENC}(l+1)$
$f_{ENC\downarrow}(l+1)$
(a)
$f_{DEC}(l+1)$
$f_{ENC}(l+1)$
Conv(C_{in}, $4C_{out}$)
Pixel Shuffle (2)
Conv($2C_{out}$, C_{out})
ReLU
$f_{DEC}(l)$
(c)
$f_{RG}(n)$
WRCA Block$_0$ (C_{in})
WRCA Block$_1$ (C_{in})
WRCA Block$_{M-2}$ (C_{in})
WRCA Block$_{M-1}$ (C_{in})
Conv(C_{in}, C_{in})
$f_{RG}(n+1)$
(b)

Figure 2. Detailed architecture of the wide receptive field and channel attention network. (a) Encoder architecture; (b) architecture of residual group composed of wide receptive residual blocks with channel attention and convolution layer; (c) decoder architecture.

$f_{ENC\downarrow}(2)$ is passed through a convolution layer. This convolution layer expands the number of channels of $f_{ENC\downarrow}(2)$ before $f_{ENC\downarrow}(2)$ enters the RGs. Equation (3) shows the expansion operation.

$$f_{EXP} = Conv(\cdot) \circ f_{ENC\downarrow}(2) \qquad (3)$$

f_{EXP} enters several RGs, which is $RG_n(\cdot)$, for nonlinear transformations and the following convolution layer. In this study, the proposed WRCAN consists of four RGs. Equation (4) shows the operation of the RGs. In this equation, f_{NLT} is the nonlinear transformed feature map from f_{EXP}, and $+$ represents the pixel-wise addition of the feature maps.

$$\begin{aligned} f_{NLT} = f_{EXP} + Conv(\cdot) \circ RG_3(\cdot) \circ RG_2(\cdot) \circ RG_1(\cdot) \\ \circ RG_0(\cdot) \circ f_{EXP} \end{aligned} \qquad (4)$$

Figure 2(b) shows the architecture of a single RG, which basically follows that of a RG in [6]. The proposed RG consists of M number of the wide receptive residual blocks with channel attention (WRCAs) and one convolution layer. The WRCA is newly proposed in this paper and its detailed architecture is discussed in the next section. Herein, the input and the output feature maps of the RG are represented as $f_{RG}(n)$ and $f_{RG}(n+1)$, respectively. In the proposed RG architecture, 16 WRCAs constitute one RG ($M = 16$). After this nonlinear transformation, the f_{NLT} passes through two decoder blocks.

Figure 2(c) shows the architecture of the decoder, where $f_{DEC}(l+1)$ is upscaled by two to match the spatial resolution of $f_{ENC}(l+1)$. One convolution layer and one pixel shuffle layer [38] double the height and width of the input feature map. The doubled $f_{DEC}(l+1)$, which is $f_{DEC\uparrow}(l+1)$, and $f_{ENC}(l+1)$ are concatenated and passed to a convolution layer and a ReLU to generate $f_{DEC}(l)$. Equations (5.1) and (5.2) show the operations for $f_{DEC\uparrow}(l)$ and $f_{DEC}(l)$, respectively. Here, $[\cdot,\cdot]$ in Equation (5.2) represents the concatenation operation of the two feature maps.

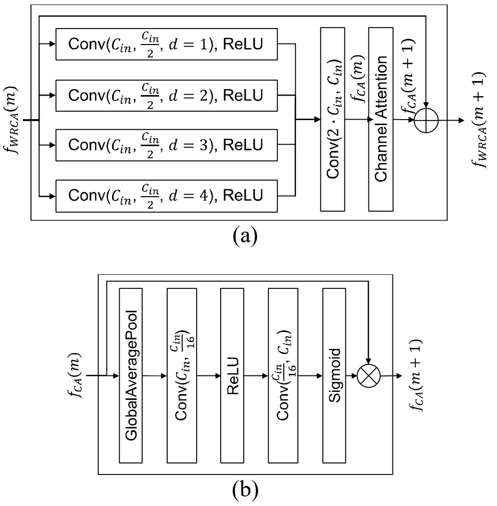
$$\begin{aligned} f_{DEC\uparrow}(l) &= Pixelshuffle(2) \circ Conv(\cdot) \circ f_{DEC}(l) && (5.1) \\ f_{DEC}(l) &= ReLU \circ Conv(\cdot) \\ &\quad \circ [f_{DEC\uparrow}(l+1), f_{ENC}(l+1)] && (5.2) \end{aligned}$$

Through the operations of the two decoder blocks, the residual feature r_S, i.e., $f_{DEC}(0)$, is generated. The sharpened feature f_S is generated by adding f_{FE} to r_S, and the final output I^S is generated from the f_S through one convolution layer.

Wide receptive residual block with channel attention (WRCA)

This section discusses the proposed WRCA as a basic block that composes the RG of the WRCAN. The purpose of the WRCA is to enable the WRCAN to extract features from a large receptive field and emphasize important channels from the extracted features. A residual block with parallel atrous convolution is known to be effective in extracting features with various receptive fields [3]. It is known that the channel attention mechanism can emphasize important channels of a feature map [6]. In this paper, to utilize the advantages of both architectures, a residual block with parallel atrous convolution and the channel attention mechanism are combined for the WRCA. Consequently, a series of WRCAs in each RG can extract important features in large image areas. In other words, the WRCAN utilizes feature maps both within and across the feature map channels. The detailed architecture of the WRCA is as follows. Figure 3(a) shows the architecture of the proposed WRCA. Here, $f_{WRCA}(m)$ is the input feature map of the m-th WRCA in each RG, and $f_{CA}(m)$ represents the input feature map of the channel attention module in the m-th WRCA. $f_{CA}(m+1)$ represents the output of the channel attention module of the m-th WRCA. $f_{WRCA}(m+1)$ is the output of the m-th WRCA, which becomes the input feature map of the $m+1$-th WRCA. Each WRCA extracts features with a large receptive field through multiple rates of atrous convolutions and then emphasizes the important channels of the atrous convolutions' outputs using the channel attention module.

The WRCA adopts four parallel convolutions with different atrous rates, i.e., 1, 2, 3, and 4, as used in [3]. However, the proposed method uses ReLU activation instead of a leaky rectified linear unit (LeakyReLU) [39] for atrous convolution layers to achieve fast convergence in



$f_{WRCA}(m)$
Conv(C_{in}, $\frac{C_{in}}{2}$, $d = 1$), ReLU
Conv(C_{in}, $\frac{C_{in}}{2}$, $d = 2$), ReLU
Conv(C_{in}, $\frac{C_{in}}{2}$, $d = 3$), ReLU
Conv(C_{in}, $\frac{C_{in}}{2}$, $d = 4$), ReLU
Conv($2 \cdot C_{in}$, C_{in})
$f_{CA}(m)$
Channel Attention
$f_{CA}(m+1)$
$f_{WRCA}(m+1)$
(a)
$f_{CA}(m)$
GlobalAveragePool
Conv(C_{in}, $\frac{C_{in}}{16}$)
ReLU
Conv($\frac{C_{in}}{16}$, C_{in})
Sigmoid
$f_{CA}(m+1)$
(b)

Figure 3. Architecture of basic block of WRCAN. (a) Architecture of wide receptive residual block with channel attention; (b) architecture of channel attention module.

training. One convolution layer with no activation is added after the concatenation of the four atrous convolutions. This convolution layer is working as a local feature extraction layer. The output of the local feature extraction layer, $f_{CA}(m)$, enters the channel attention module, as shown in Figure 3(b). This channel attention module extracts the weights of each channel via global average pooling, channel squeezing, and expansion; subsequently, it multiplies the extracted weights with the input feature map. Equation (6) shows the output of the m-th WRCA, $f_{WRCA}(m+1)$.

$$f_{WRCA}(m+1) = f_{WRCA}(m) + f_{CA}(m+1) \tag{6}$$

Table 1 shows an investigation into the WRCA architecture. In this table, WR represents the wide receptive field block with the atrous convolution layers, and CA represents the channel attention mechanism. The Case 1 in the table represents the residual block of [40]. For this investigation, the number of RG is reduced to one, and 300 epochs are trained with L1 loss. PSNR and SSIM are measured on REDS validation set. As shown in the Case 2 and Case 3, when only one of WR and CA is applied to the residual block, PSNR is improved by 0.15dB and 0.11dB, respectively. When both WR and CA are applied to the residual block, which is the proposed WRCA, PSNR is improved by 0.26dB as shown in Case 4. These experiments show that the proposed WRCA is an effective basic block.

The WRCA has an architecture similar to that of the RCA-ASPP [41]. However, RCA-ASPP uses a 1 × 1 convolution and a ReLU after the atrous convolutions to fuse the channel-wise feature map information, whereas the WRCA adopts a 3×3 convolution layer and no activation after atrous convolutions to extract local features. In addition, the purpose of RCA-ASPP is different from that of the WRCAN in that it is used to resolve the misalignment between depth maps and color guidance images.

Case	WR	CA	PSNR (dB)	SSIM
1	×	×	28.45	0.8030
2	✓	×	28.60	0.8061
3	×	✓	28.56	0.8057
4	✓	✓	28.71	0.8089

Table 1. An investigation into the WRCA architecture. ✓ and × indicate that each method is applied and not applied, respectively.

3.2. Auto Encoder Loss for JPEG artifacts

This section discusses the proposed JPEG auto-encoder loss, L_{AE-JP} . To effectively reduce JPEG artifacts, the WRCAN should be trained with an appropriate loss function that represents the characteristics of JPEG-compressed images. The proposed $L_{AE-JPEG}$ utilizes a JPEG-compressed ground-truth image, $I^{JPEG(HR)}$, which can be easily generated by compressing the ground-truth image in JPEG format. Equation (7) shows the proposed $L_{AE-JPEG}$, which uses the L2 norm for a stable training. The magnitude of the gradient of the L1 norm is 1 except for zero loss; however, that of the L2 norm decreases with $L_{AE-JPEG}$.

$$L_{AE-JPEG}(I^{AE-JPEG}, I^{HR}) = |I^{AE-JPEG}, I^{HR}|_2 \tag{7}$$

Here, I^{HR} represents the ground-truth image. $I^{JPEG(HR)}$ and $I^{AE-JPEG}$ are the input and output of the auto-encoder, respectively. The gray region in Figure 1 represents the auto-encoder for the proposed $L_{AE-JPEG}$. Therefore, the auto-encoder of the proposed WRCAN is the rest part of the WRCAN except the RGs and one following convolution layer of the RGs. In addition, the auto-encoder includes skip connections. Equation (8) shows the procedure for obtaining the I^{AE-JP} from a $I^{JPEG(HR)}$. In this equation, skip connections are excluded for simplicity.

$$\begin{aligned} I^{AE-JPEG} = {} & Conv \circ Decoder_0 \circ Decoder_1 \\ & \circ Conv \circ Encoder_1 \circ Encoder_0 \\ & \circ Conv \circ I^{JPEG(HR)} \end{aligned} \tag{8}$$

The auto-encoder loss [42] proposed by Kwak and Son, is for image super-resolution, and its purpose is to model generalization by inputting the same ground-truth images into the encoder and decoder. By contrast, the proposed $L_{AE-JPEG}$ is intended for training the WRCAN such that the original image can be restored from a JPEG-compressed image. Therefore, the encoders are trained to output meaningful features by considering the JPEG compression artifacts. The RGs and the decoders are trained to restore the original image with the feature map provided by the encoders. The proposed $L_{AE-JPEG}$ is a computationally efficient method for improving the performance of the WRCAN without increasing its size.

4. Experimental Results

4.1. Dataset and Evaluation method

The REDS dataset [43], which is the dataset for the NTIRE 2021 Image Deblurring Challenge Track 2. JPEG Artifacts [9], is used for training and validating of the proposed network. The training set of the dataset is composed of 240 image sequences. The validation and test set consist of 30 sequences of images, respectively. Each sequence of the training, validation, and test datasets consists of 100 ground-truth images (I^{HR}) and 100 blurred images (I^{B}) compressed in JPEG format using a quality factor of 25. To evaluate the proposed network, PSNR and SSIM between the restored image I^{S} and the I^{HR} are measured on RGB channels.

4.2. Training Details

Various types of data augmentations are applied to train the WRCAN. The training data are augmented with random horizontal flips and rotations. The RGB channels of I^{B} and I^{HR} are randomly permuted with a probability of 0.5. In addition, Gaussian random noise with a standard deviation of 2 is applied with a probability of 0.5. After these augmentations, the pixel values of the training images are normalized to the range of [0, 1]. The batch size is set to 16. The WRCAN is trained 400 epochs with a patch size of 256×256 or is trained 375 epochs with a patch size of 320×320. During the training, one patch is randomly fetched from each image. Hence, one epoch contains 24,000 pairs of patches. The Adam optimizer [44] is used for training the proposed model. The initial learning rate is set to 1×10^{-4}, and the learning rate is halved at 100, 200, 250, 300, and 350 epochs during the training. The weight decay parameter is set to 10^{-8} when L_{AE-JP} is applied.

Equation (9) shows the loss function for training the proposed network, where both the L1 loss and $L_{AE-JPEG}$ are used.

$$L(I^{S}, I^{AE-JPEG}, I^{HR}) = \lambda_0 \cdot L1(I^{S}, I^{HR}) + \lambda_1 \cdot L_{AE-JPEG}(I^{AE-JPEG}, I^{HR}) \qquad (9)$$

In this study, λ_0 and λ_1 are set to 1 and 0.1, respectively. To calculate $L_{AE-JPEG}$, $I^{JPEG(HR)}$ is generated by a JPEG compression of the I^{HR} with a quality factor of 25. After the training process, the model with the highest PSNR on the sequence 000 in the validation set is selected as the parameter of the WRCAN. The proposed WRCAN is implemented using PyTorch [38, 45] and trained using four NVIDIA RTX 2080Ti GPUs. For the training of the WRCAN, 9 days and 12 days are required when the network is trained with L1 loss and $L1 + L_{AE-JPEG}$ losses, respectively. The number of parameters of the WRCAN is 156,974,339 including bias.

Method	PSNR (dB)	SSIM
I^{LR}	24.91	0.7963
$WRN\ with\ L1$	28.90	0.8131
$WRCAN\ with\ L1$	28.93	0.8138
$WRCAN\ with\ L1 + L_{AE-JPEG}$	28.99	0.8154
$WRCAN\ with\ L1$ *	29.06	0.8160
$WRCAN\ with\ L1 + L_{AE-JPEG}$ *	29.12	0.8175
$WRCAN\ with\ L1 + L_{AE-JPEG}$ * $patch = 320$	29.20	0.8192

Table 2. Average PSNR (dB) and SSIM on REDS validation set. The $\boldsymbol{L1}$ loss and $\boldsymbol{L_{AE-JPEG}}$ represent the loss functions used for training WRCAN. Here, * indicates self-ensemble (×8). The 320 in the last row represents the training patch size is 320×320.

4.3. Evaluation of WRCAN

Quantitative and Qualitative Evaluations

Table 2 shows the average PSNR and SSIM values of the proposed network on the REDS validation set. In the third row of this table, the average PSNR value of the WRCAN without the channel attention mechanism, which is named as wide receptive field network (WRN), is shown. Compared with the WRCAN in the fourth row, the channel attention mechanism enables the network to achieve a PSNR that is higher 0.03dB. In addition, the WRCAN trained with the L1 loss and the proposed $L_{AE-JPEG}$ achieves a higher average PSNR than the model trained with only the L1 loss by 0.06dB. This shows that L_{AE-JP} enables the WRCAN to further improve the performance of the WRCAN for deblurring the JPEG-compressed image without additional computations at the inference time. Compared with the single model, the self-ensemble method [38] improves the PSNR by 0.13dB. Seven augmented input images are generated from the I^{B} via flipping and rotation. Eight input images, one I^{B}, and seven augmented images are input to the WRCAN. The eight output images are aligned to I^{B} and averaged pixel-wise. As shown in the last row in Table 2, the WRCAN trained with a patch size of 320×320 results in an output that is 0.08dB higher than that of the WRCAN trained with a patch size of 256×256 on validation images of the REDS dataset. This result shows that a larger-sized training patch improves the

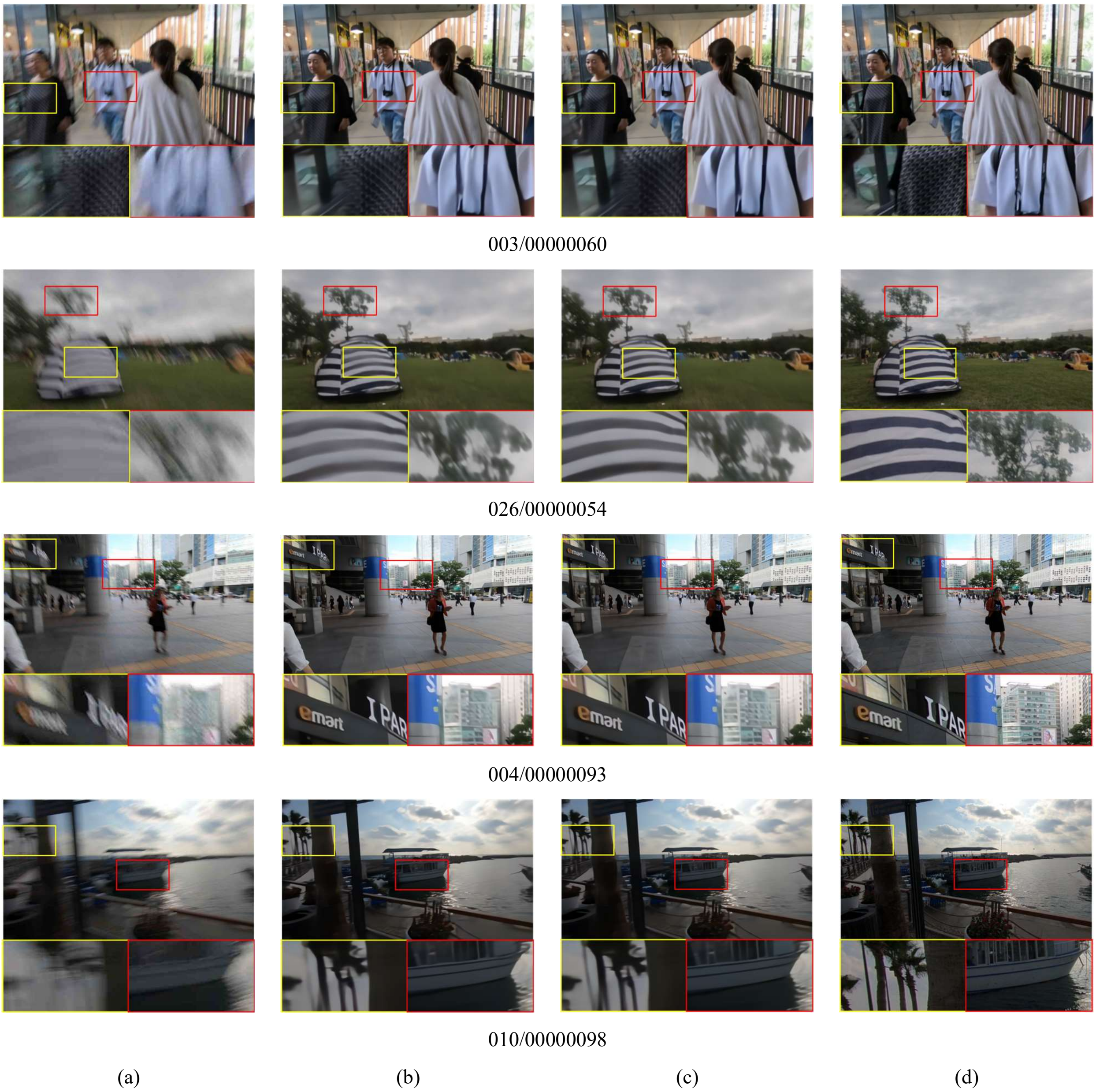


Figure 4. Examples of input, results, and ground truth sharp images from the REDS validation set. (a) Blurred input images; (b) Results of WRCAN with L1 loss; (c) Results of WRCAN with L1 and JPEG auto-encoder losses; (d) Ground truth images.

performance of the WRCAN.

The qualitative evaluation shows the effect of $L_{AE-JPEG}$ more clearly. Figure 4 shows a comparison of the result images of the WRCAN with the input and ground-truth images. A self-ensemble is applied to the WRCAN's results. In Figure 4, the "003" and "00000060" represents the names of the sequence and frames, respectively, in 003/00000060. In the case of the result of 003/00000060, the WRCAN trained with $L_{AE-JPEG}$ recovers the clothes and the camera strap, but the network trained with L1 loss doesn't. In addition, the stripe in image 026/00000054 is well recovered by the WRCAN trained with $L_{AE-JPEG}$, but the same position in the result of the WRCAN trained with the L1 loss is blurred. The other images indicate the same tendency. In the case of 004/00000093, the proposed WRCAN trained with $L_{AE-JPEG}$ is found to best recover details such as characters, and window frames the most effectively. In addition, in 010/00000098, the trees and

Team	Image Deblurring Track 2. JPEG artifacts			
	PSNR↑	SSIM↑	LPIPS↓	Rank
The Fat, The Thin and The Strong	29.70	0.8403	0.2319	1
Noah_CVlab	29.62	0.8397	0.2304	2
CAPP_OB (Ours)	**29.60**	**0.8398**	**0.2302**	**3**
Baidu	29.59	0.8381	0.2340	4
SRC-B	29.56	0.8385	0.2322	5
Mier	29.34	0.8355	0.2546	6
VIDAR	29.33	0.8565	0.2222	7
DuLang	29.17	0.8325	0.2411	8
TeamInception	29.11	0.8292	0.2449	9
Giantpandacv	29.07	0.8286	0.2499	10
Maradona	28.96	0.8264	0.2506	11
LAB FHD	28.92	0.8259	0.2424	12
SYJ	28.81	0.8222	0.2546	13
Dseny	28.26	0.8081	0.2603	14
IPCV IITM	27.91	0.8028	0.2947	15
DMLAB	27.84	0.8013	0.2934	16
Blur Attack	27.41	0.7887	0.3124	17

Table 3. Comparison of the methods on REDS test set blurred and JPEG compressed. CAPP_OP represents the result of WRCAN. Here, ↑ indicates the higher value is the better result and ↓ means the lower value is the better result.

window frame of the boat are clearly recovered by the WRCAN trained with L_{AE-JP} . Based on the quantitative and qualitative evaluations, the proposed $L_{AE-JPEG}$ improves the performance of the WRCAN by enhancing the ability to restore the original image from a JPEG-compressed image without additional computations at the inference time.

Experimental Results on NTIRE 2021 Challenge

Table 3 shows the results of NTIRE 2021 Image Deblurring Challenge Track 2. JPEG artifacts [9], where the WRCAN is trained with a patch size of 320×320 and 375 epochs. The rank is determined based on the PSNR. The SSIM and LPIPS, which is learned perceptual image patch similarity [46], are considered when the difference in the PSNR is not significant.

The result of the WRCAN trained with the proposed loss function is shown as CAPP_OB. The proposed method ranked third. In terms of PSNR, the teams ranked from second to fourth are similar. However, the SSIM result of the WRCAN is the second highest, and it also has a significant margin compared with the team ranked in the fourth. In addition, the WRCAN demonstrates the second-best result in terms of the LPIPS. These results show that the WRCAN provides state-of-the-art performances in the field of deblurring images compressed by JPEG.

5. Conclusion

A state-of-the-art CNN architecture for JPEG image deblurring is proposed herein. The proposed WRCAN has a large receptive field and emphasizes the importance of feature map channels. For a large receptive field, the residual atrous convolution layer constitutes the basic block. In addition, the channel attention mechanism adaptively weights the highly informative channels of the feature map after the parallel atrous convolution layers. Hence, the proposed WRCAN can capture and utilize meaningful features from the input image. In addition to the WRCAN, L_{AE-JP} is proposed. $L_{AE-JPEG}$ enables the JPEG artifacts to be aware of the original image restoration function of the auto-encoder. This paper shows that adopting $L_{AE-JPEG}$ is an effective method to train the WRCAN for deblurring of JPEG-compressed images. Experimental results show that the proposed network and loss function results in an output of 29.60dB on the REDS test set and ranked third on Image Deblurring Track 2 of the NTIRE 2021 Challenges. This indicates that the proposed methods provide state-of-the-art results and are effective for deblurring images compressed by JPEG.

Acknowledgement

This work was supported by the Sun Moon University Research Grant of 2021.

References

[1] Pengju Liu, Hongzhi Zhang, Kai Zhang, Liang Lin, and Wangmeng Zuo, Multi-level Wavelet-CNN for Image Restoration, In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 773-782, 2018.

[2] Xueyang Fu, Zheng-Jun Zha, Feng Wu, Xinghao Ding, and John Paisley, JPEG Artifacts Reduction via Deep Convolutional Sparse Coding, In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2501-2510, 2019.

[3] Stephan Brehm, Sebastian Scherer, and Rainer Lienhart, High-resolution Dual-stage Multi-Level Feature Aggregation for Single Image and Video Deblurring, In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 458–459, 2020.

[4] Yulun Zhang, Kunpeng Li, Kai Li, Bineng Zhong, and Yun Fu, Residual Non-Local Attention Networks for Image Restoration, In *Proceedings of the International Conference on Learning Recognition (ICLR)*, 2019.

[5] Qing Qi, Jichang Guo, and Weipei Jin, Attention Network for Non-Uniform Deblurring, *IEEE Access*, pages 100044–100057, 2020.

[6] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu, Image Super-Resolution Using Very Deep Residual Channel Attention Networks, In *Proceedings*

of the European Conference on Computer Vision (ECCV), pages 286–301, 2018.

[7] Zhangyang Wang, Ding Liu, Shiyu Chang, Qing Ling, Yingzhen Yang, and Thomas S. Huang, D3: Deep Dual-Domain Based Fast Restoration of JPEG-Compressed Images, In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[8] Xiaoshuai Zhang, Wenhan Yang, Yueyu Hu, and Jiaying Li-u. DMCNN: Dual-Domain Multi-Scale Convolutional Neural Network for Compression Artifacts Removal, In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, 2018.

[9] Seungjun Nah, Sanghyun Son, Suyoung Lee, Radu Timofte, and Kyoung Mu Lee, NTIRE 2021 Challenge on Image Deblurring, In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops,* 2021.

[10] Rob Fergus, Barun Singh, Aaron Hertzmann, Sam T. Roweis, and William T. Freeman, Removing Camera Shake from a Single Photograph, *ACM Transactions on Graphics*, 25(3), pages 787–794, 2006.

[11] Libin Sun, Sunghyun Cho, Jue Wang, and James Hays, Edge-based Blur Kernel Estimation using Patch Priors, *IEEE International Conference on Computational Photography (ICCP)*, pages 1-8, 2013.

[12] Tomer Michaeli and Michal Irani, Blind Deblurring using Internal Patch Recurrence, In *Proceedings of the European conference on computer vision (ECCV),* pages 783-798, 2014.

[13] Zhe Hu, Sunghyun Cho, Jue Wang, and Ming-Hsuan Yang, Deblurring Low-light Images with Light Streaks, In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3382-3389, 2014.

[14] Jinshan Pan, Zhe Hu, Zhixun Su, and Ming-Hsuan Yang, Deblurring Text Images via l0-regularized Intensity and Gradient Prior, In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2901-2908, 2014.

[15] Jinshan Pan, Zhe Hu, Zhixun Su, and Ming-Hsuan Yang, Deblurring Face Images with Exemplars, In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 47-62, 2014.

[16] Tae Hyun Kim, Byeongjoo Ahn, and Kyoung Mu Lee, Dynamic Scene Deblurring, In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3160-3167, 2013.

[17] Yuval Bahat, Netalee Efrat, and Michal Irani, Non-Uniform Blind Deblurring by Reblurring, In *Proceedings of the IEEE International Conference on Computer Vision (ICCV),* pages 3286-3294, 2017.

[18] Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee, Deep Multi-Scale Convolutional Neural Network for Dynamic Scene Deblurring, In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3883–3891, 2017.

[19] Xin Tao, Hongyun Gao, Xiaoyong Shen, Jue Wang, and Jiaya Jia, Scale-recurrent Network for Deep Image Deblurring, In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8174-8182, 2018.

[20] Hongguang Zhang, Yuchao Dai, Hongdong Li, and Piotr Koniusz, Deep Stacked Hierarchical Multi-Patch Network for Image Deblurring, In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5978-5986, 2019.

[21] Adam Kaufman and Raanan Fattal, Deblurring Using Analysis-Synthesis Networks Pair, In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5811-5820, 2020.

[22] Orest Kupyn, Tetiana Martyniuk, Junru Wu, and Zhangyang Wang, DeblurGAN-v2: Deblurring (Orders-of-Magnitude) Faster and Better, In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8878-8887, 2019.

[23] Alexia Jolicoeur-Martineau, The Relativistic Discriminator: a Key Element Missing from Standard Gan, *arXiv preprint*, arXiv:1807.00734, 2018.

[24] Kaihao Zhang, Wenhan Luo, Yiran Zhong, Lin Ma, Bjorn Stenger, Wei Liu, and Hongdong Li, Deblurring by Realistic Blurring, In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2737-2746, 2020.

[25] Kiryung Lee, Dong Sik Kim, and Taejeong Kim, Regression-based Prediction for Blocking Artifact Reduction in Jpeg-compressed Images, *IEEE Transactions on Image Processing*, 14(1), pages 36–48, 2005.

[26] Seok Bong Yoo, Kyuha Choi, and Jong Beom Ra, Post-processing for Blocking Artifact Reduction based on Inter-Block Correlation, *IEEE Transactions on Multimedia*, 15(6), pages 1536–1548, 2014.

[27] Inchang Choi, Sunyeong Kim, Michael S. Brown, and Yu-Wing Tai, A Learning-Based Approach to Reduce JPEG Artifacts in Image Matting, In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 2880-2887, 2013.

[28] Rasmus Rothe, Radu Timofte, and Luc Van Gool, Efficient Regression Priors for Reducing Image Compression Artifacts, In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, 2015.

[29] Xianming Liu, Xiaolin Wu, Jiantao Zhou, and Debin Zhao, Data-Driven Sparsity-Based Restoration of JPEG-Compressed Images in Dual Transform-Pixel Domain, In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5171-5178, 2015.

[30] Chao Dong, Yubin Deng, Chen Change Loy, and Xiaoou Tang, Compression Artifacts Reduction by a Deep Convolutional Network, In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 576-584, 2015.

[31] Fisher Yu and Vladlen Koltun, Multi-scale Content Aggregation by Dilated Convolutions, In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2016.

[32] Ying Tai, Jian Yang, Xiaoming Liu, and Chunyan Xu, MemNet: A Persistent Memory Network for Image Restoration, In *Proceedings of the IEEE International Conference on Computer Vision (ICCV),* pages 4539-4547, 2017.

[33] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu, Residual Dense Network for Image Restoration, *IEEE Transactions on Pattern Recognition and Machine Intelligence (PAMI)*, 2020.

[34] Leonardo Galteri, Lorenzo Seidenari, Marco Bertini, and Alberto Del Bimbo, Deep Generative Adversarial Compression Artifact Removal, In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 4826-4835, 2017.

[35] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, U-Net: Convolutional Networks for Biomedical Image Segmentation, In *proceedings of the Internal Conference on Medical Image Computing and Computer-Assisited Intervention (MICCAI)*, pages 234–241, 2015.

[36] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille, DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs, *IEEE Transactions on Pattern Anaysis and Machine Intelligence (PAMI)*, 40(4), pages 834–848, 2018.

[37] Vinod Nair and Geoffrey E. Hinton, Rectified Linear Units Improve Restricted Boltzmann Machines, In *Proceedings of the International Conference on Machine Learning (ICML)*, 2010.

[38] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P. Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang, Real-time Single Image and Video-Super Resolution using an Efficient Sub-pixel Convolutional Neural Network, In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1874–1883, 2016.

[39] Andrew L. Maas, Awni Y. Hannun, and Andrew Y. Ng, Rectifier Nonlinearities Improve Neural Network Acoustic Models, In *Proceedings of the International Conference on Machine Learning (ICML)*, 30(1), 2013.

[40] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee, Enhanced Deep Residual Networks for Single Image Super-resolution, In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 136–144, 2017.

[41] Tao Li, Xiucheng Dong, and Hongwei Lin, Guided Depth Map Super-Resolution Using Recumbent Y Network, *IEEE Access* (8), pages 122695–122708, 2020.

[42] Junhyung Kwak and Donghee Son, Fractal Residual Network and Solutions for Real Super-resolution, In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2019.

[43] Seungjun Nah, Sungyong Baik, Seokil Hong, Gyeongsik Moon, Sanghyun Son, Radu Timofte, and Kyoung Mu Lee, NTIRE 2019 Challenge on Video Deblurring and Super-Resolution: Dataset and Study, In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2019.

[44] Diederik P. Kingma and Jimmy Ba, A Method for Stochastic Optimization, In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2015.

[45] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala, Pytorch: An Imperative Style, High-performance Deep Learning Library, In *Proceedings of Advances in Neural Information Processing Systems (NeuralIPS)*, 2019.

[46] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang, The Unreasonable Effectiveness of Deep Features as a Perceptual Metric, In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 586–595, 2018