

Article

Image Super-Resolution Algorithm Based on Dual-Channel Convolutional Neural Networks

Yuantao Chen ¹ , Jin Wang ^{1,2,*} , Xi Chen ¹, Arun Kumar Sangaiah ³ , Kai Yang ⁴ and Zhouhong Cao ⁵

¹ School of Computer and Communication Engineering & Hunan Provincial Key Laboratory of Intelligent Processing of Big Data on Transportation, Changsha University of Science and Technology, Changsha 410114, China; chenyt@csust.edu.cn (Y.C.); chentianjun@163.com (X.C.)

² School of Information Science and Engineering, Fujian University of Technology, Fujian 350118, China

³ School of Computing Science and Engineering, Vellore Institute of Technology University, Vellore 632014, India; arunkumarsangaiah@gmail.com

⁴ Technical Quality Department, Hunan ZOOMLION Heavy Industry Intelligent Technology Corporation Limited, Changsha 410005, China; yangkai@zoomlion.com

⁵ School of Hydraulic Engineering & Hunan Provincial Science and Technology Innovation Platform of Key Laboratory of Dongting Lake Aquatic Eco-Environmental Control and Restoration, Changsha University of Science and Technology, Changsha 410114, China; caozhouhong@csust.edu.cn

* Correspondence: jinwang@csust.edu.cn; Tel.: +86-180-1484-9250

Received: 17 March 2019; Accepted: 30 May 2019; Published: 5 June 2019



Abstract: For the image super-resolution method from a single channel, it is difficult to achieve both fast convergence and high-quality texture restoration. By mitigating the weaknesses of existing methods, the present paper proposes an image super-resolution algorithm based on dual-channel convolutional neural networks (DCCNN). The novel structure of the network model was divided into a deep channel and a shallow channel. The deep channel was used to extract the detailed texture information from the original image, while the shallow channel was mainly used to recover the overall outline of the original image. Firstly, the residual block was adjusted in the feature extraction stage, and the nonlinear mapping ability of the network was enhanced. The feature mapping dimension was reduced, and the effective features of the image were obtained. In the up-sampling stage, the parameters of the deconvolutional kernel were adjusted, and high-frequency signal loss was decreased. The high-resolution feature space could be rebuilt recursively using long-term and short-term memory blocks during the reconstruction stage, further enhancing the recovery of texture information. Secondly, the convolutional kernel was adjusted in the shallow channel to reduce the parameters, ensuring that the overall outline of the image was restored and that the network converged rapidly. Finally, the dual-channel loss function was jointly optimized to enhance the feature-fitting ability in order to obtain the final high-resolution image output. Using the improved algorithm, the network converged more rapidly, the image edge and texture reconstruction effect were obviously improved, and the Peak Signal-to-Noise Ratio (PSNR) and structural similarity were also superior to those of other solutions.

Keywords: super-resolution; dual-channel; residual block; convolutional kernel parameter; long-term and short-term memory blocks

1. Introduction

Because images are affected by both the image processing system and the transmission environment during the process of acquisition, the resolution of the original image is typically low; moreover, since key information is missing from these original low-resolution images, they are generally not capable of

meeting many actual user needs. Accordingly, the use of high-resolution images is required in some areas and fields of research. In order to solve the problems caused by low image quality, Single Image Super Resolution (SISR) technology is used to transform a single Low-Resolution (LR) image into a High-Resolution (HR) image containing rich high-frequency information. There are wide applications for this technology in the research fields of object detection, satellite image, medical image and face recognition [1–4].

Traditional *SISR* methods have included interpolation methods based on the sample extraction theory, such as Bicubic Interpolation [5] and Bilinear Interpolation [6]. The image reconstruction is based on methods including the Iterative Back Projection (IBP) method [7], the Projection Onto method (PO) [8], the Maximum A Posteriori method (MAP) [9], and so on. Based on learning methods such as embedded neighborhood [10], the regression or mapping relationship between HR and LR blocks has been understood by using the concept of geometric similarity. In sparse representation based on the interrelated approach, Yang et al. [11] and Yang et al. [12] reconstructed HR image blocks and HR images by strengthening the similarity between LR and HR image blocks and their corresponding real dictionaries, so that the sparse representation of the LR block and the super-completed HR dictionary can be used to reconstruct HR image blocks and then connect HR images. A complete high-resolution image is obtained like a block [13–16].

In recent years, deep learning has achieved remarkable results in the research field of image super resolution, benefiting from the powerful feature characterization [17] of deep learning, which is more effective than traditional methods. Dong et al. [18] first proposed the application of the Super Resolution using Convolutional Neural Networks (SRCNN) algorithm to super-resolution images. Compared with traditional methods, the simple network structure obtains the ideal super-resolution; however, there are limitations of the simple network structure. Firstly, it is dependent on the context information of small image blocks. Secondly, the training convergence is too slow, and the time complexity is high. Thirdly, the simple network only can be used for a single-scale super resolution (SR) procedure. Dong et al. [19] proposed the Fast Super-Resolution Convolutional Neural Network (FSRCNN) by reducing the speed training of the network parameters. FSRCNN used eight layers of network structure, making it deeper than SRCNN; moreover, instead of Bicubic Interpolation, the anti-coiling layer was used on the last layer of the network. Finally, FSRCNN has achieved success in the convergence and super-resolution reconstruction field. Considering the slow convergence and shallow network of SRCNN and FSRCNN networks, Wang et al. [20] proposed an image super-resolution algorithm (EEDS) based on end-to-end and shallow convolutional neural networks that has achieved better performance than others. However, because the deep network cannot fully extract the features of an *LR* image in the feature extraction stage, the loss of useful information and long-term memory content during the reconstruction process becomes serious when the feature of the up-sampling process is nonlinear mapping, as this causes the effect of super resolution to be reduced by the deep network. However, generally speaking, the shallow network master is the main problem. Moreover, when restoring the main components of LR images, the fast convergence of the network can be limited if too many parameters are used. Kim et al. [21] proposed a highly accurate single-image super-resolution method named Very Deep Networks for Super Resolution (VDSR). By using a very deep convolutional network of *VGG-net* [22] in image classification, the model employs cascaded small filters in a deep-network structure, using 20 weight-layers to efficiently utilize the context information of the large image region.

Moreover, Kim et al. [23] proposed the Deep Recursive Convolutional Network (DRCN) for image super resolution. The network uses a very deep recursive layer (as many as 16 recursions), as increasing the recursion depth can improve the performance without the need to introduce additional parameters to additional convolutions. In order to prevent the explosion and disappearance of the gradient, as well as to reduce the difficulty of training, the recursive monitoring and skipping connection methods are far more effective than previous methods. Recently, Ke et al. [24] proposed the Gradual Up-Sampling Network (GUN) method, which is based on a deep convolutional neural network. This method uses a gradual process to simplify the direct SR problem into a multi-step sampling task

that employs very small magnification at each step. The Enhanced Deep Residual Networks for Single-Image Super Resolution (EDSR) and the Multi-Scale Deep Super-Resolution (MDSR) network were proposed by Lim et al. [25] among others. The model is optimized by removing unnecessary modules from the residual network to significantly enhance the performance of the model. Moreover, by extending the size of the model to further improve the performance, *MDSR* can reconstruct HR images with different magnification factors using a network model. Tai et al. [26] proposed a very deep Memory Network (MemNet) for image restoration, which introduces memory blocks consisting of a recursive unit and a gate control unit that mine persistent memory through an adaptive learning process. The representation and output from previous memory blocks are connected and sent to the gate control unit. The gate control unit is adaptive to control memory [27,28] and controls how many previous states should be retained and how many current states should be stored to achieve superior performance in super-resolution tasks [29,30].

By exploring the above methods and combining them with MemNet [26] and Deep Residual Network (ResNet) [31], Nair et al. [32] proposed an enhanced algorithm of image super-resolution based on Dual-Channel Convolution Neural Network (DCCNN), related to SRCNN and EEDS to solve the above problems. The shallow channel is mainly used to restore the overall outline of the original image and to achieve fast convergence performance. By adjusting the parameters of the three-layer network from the shallow channel, it can quickly converge while ensuring the restoration of the main components from the image. By contrast, deep channels are used to extract detailed texture information from LR images. Deep channels are divided into three steps: feature extraction and mapping, up-sampling, and long-term and short-term memory block reconstruction. Because there are fewer network layers in the extraction stage of the original model, the local sensing field of the image is too small, and the full LR image feature extraction will lead to the final SR effect. In order to avoid loss of important high-frequency content, the proposed model increases the residual layer on the original basis by increasing the number of network layers in the process; it also reduces the LR feature mapping dimension, such that the residual layer can learn edge and texture information of the image better than the common stacked convolution, and the increased network depth avoids the network. It is difficult to train the problem, meaning that the feature can be directly transmitted to the lower level so as to optimize the gradient vanishing problem and make it easier for the network to enhance the training performance. During the up-sampling phase, because the sampling operation is an important part of the model, the goal is to increase the space span to the target of the HR size. In order to get good results, a 1×1 filter is used to increase the number of dimensions to 64 after the mapping is complete. In addition, deconvolution is used to achieve the sampling rather than manual designing. During memory block reconstruction in the long-term and short-term period, because the reconstruction stage directly determines the HR reconstruction effect of the deep channel, the long-term and short-term memory blocks made up of the residual block are used after up-sampling to further reduce the loss of high-frequency information, such that the reconstructed HR image texture information is more abundant. Finally, the deep and shallow passages are jointly optimized to obtain the final HR image. Experimental results show that the effect of network super resolution is better than that of bicubic interpolation, A+ [11], SRCNN [18], and EEDS [20] super-resolution reconstruction algorithms.

2. Related Works

2.1. The SCRNN Model

In the learning-based super-resolution image algorithm, *SRCNN* applied a convolutional neural network to the task of image super resolution for the first time. Compared with traditional methods, the method can directly learn the mapping relationship between LR images and HR images.

As shown in Figure 1, the process of the proposed algorithm was divided into three stages. The data are pre-processed, the training dataset of 91 images is taken to make up the image block of 14, and the LR image block after the bicubic interpolation pre-processing procedure is used as the input

for the network. The first layer uses 64 filters and the convolutional core of the size of 3 channels of image block performs feature extraction and representation; at this time, the number of channels is expanded from 3 to 64. The second layer uses 64 filters, and the convolution nucleus (of size 1×1) conducts the nonlinear mapping to extract features; at this time, the number of channels is reduced from 64 to 3. The third layer uses a convolution nuclear size of 5×5 to reconstruct the *HR* image block at this time; the number of channels decreases from 64 to 3. Finally, the mean squared error (MSE) corresponding to the original image and HR output image is constructed to optimize the model's parameters. In SRCNN, the experimental results show that the super-resolution effect is improved by using a large scale of dataset for ImageNet.

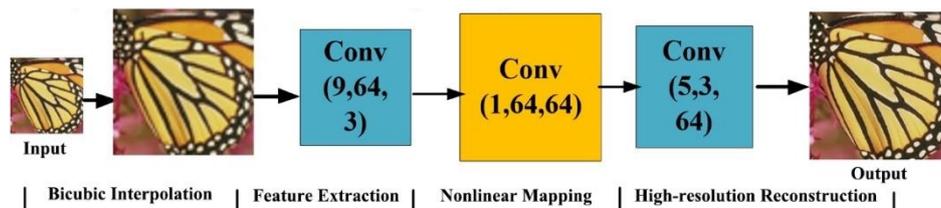


Figure 1. The processing procedure of Super Resolution using Convolutional Neural Networks (SRCNN) construction.

2.2. Image Super-Resolution Algorithm Based on Dual-Channel Convolutional Neural Networks

The image super-resolution algorithm is based on dual-channel convolutional neural networks, such as EEDS [21], and is also a learning-based SISR algorithm. The EEDS algorithm works to improve SRCNN and FSRCNN: its structure is deeper than those of SRCNN and FSRCNN, and the residual block with jump layer, the residual network because of the existence of the fast connection. Data transmission between the network is smoother, and the gradient is improved, resulting in the loss of fitting and making it easier for the network to converge. The network structure of EEDS is divided into two parts: the deep layer and the shallow layer. The deep network contains 13 layers, including a feature extraction layer, an up-sampling layer, and q multi-scale reconstruction layer. The shallow network contains three layers. The design idea comes from the three-layer model of SRCNN, in which the anti-coiling layer replaces the original SRCNN nonlinear mapping layer. Finally, using a deep network combined with the output of the shallow network, the final output of the HR image is obtained.

In the training process, 91 training images are first scaled, rotated, and fragmented, and then sampled according to the required ratio. The obtained *LR* image blocks are input into the double-layer network. The MSE corresponds to the original image, the output HR image is constructed, and the model parameters are optimized.

The shortcomings of the network are as follows: because the deep network cannot fully extract the features of the *LR* image in the feature extraction stage, the nonlinear feature mapping of the up-sampling process leads to the loss of useful information and of long-term memory content in the reconstruction process, which causes the deep network to discount the effect of super-resolution. The shallow network is mainly used to restore the main components of LR images. Too many parameters will limit the fast convergence of the network.

3. Dual-Channel Convolutional Neural Networks

3.1. The Improved Ideas

Because the shallow network cannot adequately extract the features of the *LR* image, the effect produced by super resolution is not ideal. Although the deep network is superior to the shallow network in depth, the deepening architecture of the network will also cause the network to be difficult to train, and the gradient disappearance/explosion will affect the stability of the network. Therefore,

by combining with the two factors of width and depth of the network, the SRCNN and EEDS have been improved.

The network structure of SRCNN and EEDS is that of a three-layer network, that of the DCCNN is “deep and shallow”, using 13 tiers and 3 tiers, respectively, in a dual-channel network; the shallow channel is used to restore the overall outline of the image, while the deep channel is used to restore rich texture information. Therefore, the combination of these two channels can effectively improve the efficiency of training, enhance the feature-fitting ability, and reduce the computational complexity of the whole model. On the basis of the above factors, the present paper selects the parameters of the shallow channel with a convolutional kernel while adjusting the depth of the deep channel network, so that the shallow channel is mainly responsible for the convergence performance of the network to reduce the time complexity of the model; moreover, the deep channel is mainly responsible for more detailed texture recovery and for improving the restoration precision of the network. It is more efficient to learn the texture information at a high level, and the feature of fitting the image is more accurate, considering that the use of the residual block and the jump layer results in faster convergence than a simple increase in the number of network layers and also reduces the gradient dispersion and the loss of features.

Therefore, residual blocks and skip layers in deep channels are used in this paper. At the same time, as the depth increases, it is more difficult for the model to achieve long-term dependence at each stage; this leads to the reduction of dependence during the up-sampling component of the reconstruction phase and the increased loss of the important, higher-frequency information in the up-sampling stage. Accordingly, in this paper, three residual blocks are selected in the reconstruction stage to grow the short-term memory blocks to the up-sampling feature. The space is rebuilt. Finally, this paper proposes an image super-resolution algorithm based on DCCNN with a deep channel of 19 layers and a shallow channel of 3 layers.

3.2. The Network Structure of DCCNN

The image super-resolution algorithm based on DCCNN fully considers the nonlinear mapping relationship between the low-resolution image and the super-resolution image, and the characteristics of the dual-channel are equal to those of the proposed model. The corresponding weights for each channel are not shared in DCCNN. The shallow channel is mainly used to restore the overall outline of the image. The deep channel is used to extract detailed texture information of the LR image. In the phase of feature extraction and mapping with the deep channel, the input layer of the proposed network is the three-channel LR image, which is 48×48 size of units.

Figure 2 presents the dual-channel network constructed in this paper, which is divided into two sub-channels: the deep channel and the shallow channel. Firstly, the number of channels is increased to 64 through the convolutional kernel of 3×3 size, then entered into the residual block (see Figure 3). It is composed of *Conv*, *ReLU*, and *Conv*, and the *Conv* residual block size is 3×3 , while the step length is one and the padding is two. After three residual blocks, the output has 16 48×48 characteristic graphs. At this time, the semantic information in the feature map is richer than it was previously. In the up-sampling phase, as the most important part of the network, the goal is to increase the spatial span of the LR images to HR size. After the mapping, the dimension of 1×1 is compressed from 16 to 4. Instead of using the manual interpolation method, we used deconvolution (*DeConv*) to achieve the up-sampling. The size of *DeConv* is 9×9 . For two times, three times and four times, the different scales of up-sampling by setting different steps. After deconvolution, the feature map is increased to 64. Finally, the dimension of the 1×1 filter is mapped from 64 to 4, the parameters of the 1×1 filter are effectively reduced, and the calculation complexity is also reduced. In the stage of reconstructing long-term and short-term memory, as the last stage of the network, the up-sampling phase is also the most important part, as it determines the quality of the texture information recovery from the network.

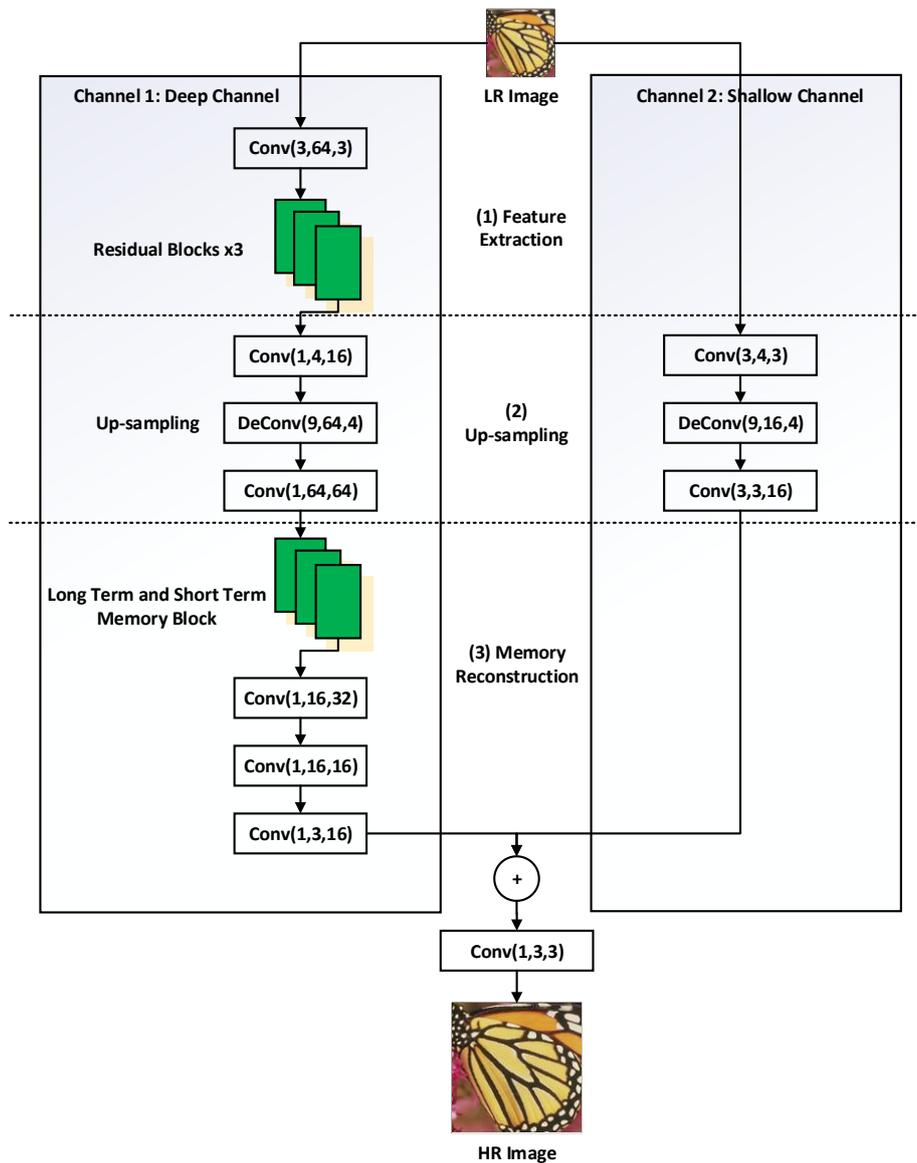


Figure 2. The processing procedure of Dual-Channel Convolutional Neural Networks (DCCNN).

Considering that it is difficult for the model to achieve long-term dependence at each stage, we used the multi-scale convolution to reconstruct the up-sampling feature space using long-term and short-term memory blocks at the beginning of the reconstruction. The long-term and short-term memory blocks, which are shown in Figure 4, consist of three residual blocks. The dimensions of the feature map range from 64 to 32, which further reduces the dimensions and enhances the nonlinear mapping ability. The size of *Conv* in the long-term and short-term memory block is 3×3 , the step length is one, the padding is two, and the output is 32 feature graphs. The 1×1 filter is then used to compress the dimensions from 32 to 16, so that the high-dimensional feature is extracted and the computational complexity is reduced. In order to effectively aggregate the local information of the 16 feature maps, multi-scale convolution is used for reconstruction. The multi-scale coiling layer contains four filters of different sizes, namely, 1×1 , 3×3 , 5×5 and 7×7 . The four filters' convolutions in the layer are parallel. Each filter has 4 outputs in the feature graph, and then the 16 feature graphs are combined. Finally, the 1×1 filter is used as the weighted combination of multi-scale texture features. At this time, the dimension of the feature map is from 16 to 1. Correspondingly, the use of deconvolutional networks to complete the up-sampling operation involves the use of a three-layer structure similar to that of SRCNN in shallow channels. The specific process is as follows: the three-channel LR image input to 48×48 is

input from the input layer to the network, and the number of channels is increased to four through the 3×3 filter. The space span of the LR image is increased to the HR size by means of deconvolution. The size of *DeConv* convolutional kernel is the same as that of the 9×9 deep learning network, and the feature map after the deconvolution is increased to 16. Finally, using the convolutional kernel of 3×3 size, the step size is one, the padding is two, and the output is a three-channel feature graph.

In order to avoid the problem of gradient disappearance, our proposed structure is deeper than that of the improved network; moreover, the characteristics of both the feature graph and the feature map of the lower layer are also different. Furthermore, in this paper, the *ReLU* [33] activation function is used in all convolution operations to improve the *PReLU* [34] activation function. All convolutional operations utilized in this paper can improve the high network's nonlinear modeling ability. At this point, the output feature graph of the shallow and deep network is optimized, the output of the two networks is added, the effective component is retained, and the texture information of the feature map is enriched. The feature graph is then input to a convolution layer of 1×1 size. Finally, the image output of HR is obtained, with the result that the image quality has greatly improved.

3.3. Residual Blocks and Long-Term and Short-Term Memory Block

(1) Residual Blocks

The residual blocks' network design is inspired by the 152-level ResNet network proposed by He et al. [31]. The recognition performance on the ImageNet dataset was improved with the increase of the number of network layers, and its performance on computer vision problems [22,23,26] from low to high tasks is excellent.

The original residual block, which is shown in Figure 3a, is composed of a feed-forward convolutional network and a jump around a number of layers. The stacked residuals form the final residual networks. Compared with a smooth network, the residual network exhibits lower convergence loss and a lack of overfitting due to the disappearance of the gradient, which makes the network easier to optimize. The dimensions of the feature map progressively increase to ensure the ability to express the output features.

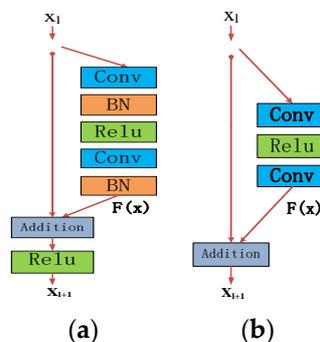


Figure 3. The processing construction with improved residual blocks described in the paper. (a) Original Residual Blocks; (b) Improved Residual Blocks in the Paper.

Since the original batch normalization layer (BN) [35] is used to normalize the characteristics of the coiling output layer, this will affect the distribution of features learned by the convolution layer and cause the loss of important information from the feature graph. Moreover, the batch positive layer has the same number of parameters as the previous convolutional layer and thus consumes a lot of memory. In their image deblurring task, Zeiler et al. [36] deleted the *BN* layer in the residual block, with the result that the network performance was greatly improved. Therefore, in this paper, we used the residual block to delete the batch regularization layer in order to reduce the color's offset in the output, while maintaining the training stability. Each residual block in the present paper contained two 3×3 convolutional layers and the *ReLU* layer. The structure of the residual block in the present paper is shown in Figure 3b.

The residual block can be expressed by Equation (1):

$$X_{l+1} = X_l + F(X) \tag{1}$$

Here, X_l and X_{l+1} represent the input and output vectors of residual blocks, respectively. The function $F(X)$ denotes residual mapping. The residual block in this paper contained only the convolutional layer and the ReLU layer. The modified linear unit (ReLU) has unilateral suppression and sparsity. In most cases, the ReLU gradient is a constant term, avoiding the problem of gradient disappearance to a certain extent. The relevant mathematical expression can be expressed by Equation (2):

$$f(x) = \max(0, x_i) \tag{2}$$

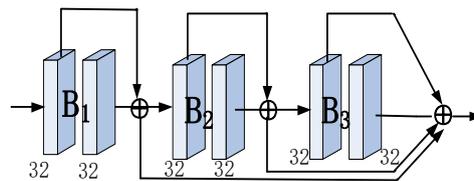


Figure 4. The processing construction of long-term and short-term memory blocks.

In this paper, the activation function of the convolutional layer outside the residual block is the Parametric Rectified Linear Unit (PReLU) [34]. The use of PReLU is mainly designed to avoid the “dead angle” [37] caused by the zero gradient in the ReLU. It is increased by the correction of parameters to a certain extent. It can have a regularizing effect and can also improve the generalization ability of the model. The difference between the proposed model and *ReLU* is mainly reflected in the negative part, and the mathematical expression is shown in Equation (3):

$$f(x) = \max(0, x_i) + a_i \min(x_i, 0) \tag{3}$$

Here, x_i is the input signal of the i th layer, and a_i is the coefficient of the negative part. In Equation (3), the parameter a_i is set to zero, but the negative part of PReLU can be learned. Finally, the output of the activation function can be expressed by Equation (4):

$$f_i(x) = f(W_l * f_{l-1}(x) + B_l) \tag{4}$$

Here, f_l is the final output feature graph and B_l is the offset of the l th layer.

(2) Long-Term and Short-Term Memory Block

It is difficult to achieve long-term dependence at each stage, resulting in lower dependence on the up-sampling phase in the reconstruction phase and more loss of important high-frequency information in the up-sampling phase. In this paper, three residual blocks (B_1, B_2, B_3) were used to synthesize the long-term and short-term memory block for the up-sampling feature space at the beginning of reconstruction. The design of the long-term and short-term memory block was inspired by He et al. [31], who proposed a very deep persistent MemNet. The construction of our long-term and short-term memory blocks is presented in Figure 4.

In this paper, we used three residual blocks to learn recursively in the long-term and short-term memory blocks. We used the eigenvector x of the up-sampling phase as input; the residual block B_i can be expressed by Equation (5):

$$B_i = F(B_{i-1}, w_i) + B_{i-1} \tag{5}$$

In Equation (5), i is set to one, two, and three. $B_1, B_2,$ and $B_3,$ respectively represent the output of the corresponding residual block. When $i = 1, B_{i-1} = x.$ F represents the residual mapping, and w_i

represents the weight vector of the residual block to learn. Since each residual block consists of two volume layers and ReLU activation functions, Equation (5) can be further expressed as Equation (6):

$$F(B_{i-1}, w_i) = w_i^2 \text{ReLU}(w_i^1 \text{ReLU}(B_{i-1})) \tag{6}$$

Here, ReLU represents the activation function, while w^1 and w^2 are the two weight vectors of the volume layer, respectively. In the interest of simplicity, the bias is omitted in the above equations.

Finally, unlike the traditional leveling network, the present paper uses cascading methods to combine the output features of the three residual blocks, which effectively avoids content loss from the previous stage. The process of calculation is shown in Equation (7):

$$B_{out} = [B_1, B_2, B_3] \tag{7}$$

Here, B_{out} represents the final output and passes to the next layer.

3.4. Loss Function and Evaluation Standard

(1) Loss Function

By minimizing the loss cost between the super-resolution image and the real high-resolution image, the network constantly adjusts the network parameters $\Theta = \{w_i, b_i\}$. For a group of real high-resolution images X_j and a group of super-resolution images, $F^j(Y; \Theta)$, is reconstructed by the network. This paper uses MSE as the cost function:

$$L(\Theta) = \frac{1}{n} \sum_{i=1}^n \|F^j(Y; \Theta) - X_j\|^2 \tag{8}$$

where n represents the number of training samples. Because the weights of the dual-channel network are not shared, they are converted to a dual-channel cost function problem:

$$L_{EDC} = \min[L_d(\Theta) + L_s(\Theta)] \tag{9}$$

Here, $L_d(\Theta)$ and $L_s(\Theta)$ are the loss costs of the deep channel and shallow channel respectively. The network uses the Adam optimization method and back-propagation algorithm [38] to minimize MSE in order to adjust the network parameters, and the update process of the network weights is as in Equation (10):

$$\Delta_{k+1} = 0.9 \times \Delta_k - \eta \times \frac{\partial L}{\partial W_k^l}, W_{k+1}^l = W_k^l + \Delta_{k+1} \tag{10}$$

Δ_k represents the updating value of the last weight, l represents the number of layers of the network, and k represents the number of iterations from the network; η is the learning rate, W_k^l represents the weight of the k th iteration in level l , $\frac{\partial L}{\partial W_k^l}$ represents the corresponding weight of the cost function and derivation of the derivative. The weights are randomly initialized according to a Gaussian distribution with mean value of zero and variance of 0.001. The model can automatically adjust the learning rate in the range of training, making the learning of the parameters more stable.

(2) Evaluation Standards

In this paper, the difference between the generated image quality and the quality of the original high-resolution image is measured by means of two common evaluation indexes, namely the Peak Signal-to-Noise Ratio (PSNR) and the Structural Similarity Index (SSIM) [29,30].

PSNR is used as an objective evaluation index of image quality, which is measured by calculating the error between corresponding pixels. The PSNR's unit is decibel (dB) [16]. The larger the value, the smaller the image distortion. The calculating equation is Equation (11):

$$PSNR = 10 \log_{10} \left(\frac{(2^n - 1)^2}{MSE} \right) \quad (11)$$

Here, MSE is the direct Mean Squared Error of the original image and the super-resolution image, $(2^n - 1)^2$ is the signal maximum square, and n is the number of bits per sampling value.

The SSIM measures image similarity in terms of three aspects: brightness, contrast ratio, and structure. The range of SSIM is [0,1], and its value is closer to one. The distortion effect is smaller. The calculation equations are as follows:

$$SSIM(X, Y) = l(X, Y) \cdot c(X, Y) \cdot s(X, Y) \quad (12)$$

$$l(X, Y) = \frac{2\mu_X\mu_Y + C_1}{\mu_X^2 + \mu_Y^2 + C_1} \quad (13)$$

$$c(X, Y) = \frac{2\sigma_X\sigma_Y + C_2}{\sigma_X^2 + \sigma_Y^2 + C_2} \quad (14)$$

$$s(X, Y) = \frac{2\sigma_{XY} + C_3}{\sigma_X\sigma_Y + C_3} \quad (15)$$

X is the super-resolution image of the LR image obtained through network training, Y is the original HR image. The variances of μ_X and μ_Y are represented by X and Y , respectively, while σ_X and σ_Y represent the variances of the super-resolution image and of the original high-resolution image, respectively, and σ_{XY} represents the covariance of the super-resolution image and the original high-resolution image. C_1, C_2, C_3 are constant terms. In order to avoid a zero in the denominator, the usual practice is to take $C_1 = (K_1 \times L)^2$, $C_2 = (K_2 \times L)^2$, $C_3 = C_2/2$ and, generally, $K_1 = 0.01$, $K_2 = 0.03$, $L = 255$.

4. Experimental Results and Analysis

4.1. Parameter Settings

The experiment used 91 pictures by Bevilacqua et al. [39] and one hundred 2K high-definition images selected from the DIV2K dataset. In short, a total of 191 images were used as training datasets to train the network model. Considering that dataset size directly affects network performance, two methods of data expansion were adopted for the image, based on the original training dataset. The image was amplified in two ways: (1) Scaling: each image was zoomed in proportion to 0.9, 0.8, 0.7, and 0.6; (2) Rotating: each image was rotated by 90 degrees, 180 degrees, and 270 degrees. Each image was used 20 times, such that 3820 images were eventually available for the training process. In this process, the sub-sampling size was 48×48 , the initial learning rate of the network was set to 0.001, and the Adam optimization method was adopted to automatically adjust the learning rate so that the network parameters could be learned smoothly. The number of images per batch was set to 64, and the network was trained 1000 times. The testing dataset comprised the internationally common datasets "Set5" [40,41] and "Set14" [42,43]. The GPU was NVIDIA GeForce 1080 T_i, the experimental environment was Keras, and Python 3.5 and OpenCV 3.0 were applied to carry out the simulation experiments. The results of the network training were compared with those of existing methods in terms of three aspects: subjective visual effect, objective evaluation index, and efficiency comparison.

4.2. Experimental Results and Comparative Analysis

In order to verify the effectiveness of the proposed image super-resolution algorithm based on DCCNN, the present paper used a trained model to reconstruct the *LR* image at “2×”, “3×”, and “4×” [44] the super resolution. The performance of the proposed DCCNN method was evaluated on the Set5 dataset and Set14 dataset, and the results were compared with the results of the existing bicubic interpolation, A+ [11], SRCNN [19], and EEDS [20] algorithms.

Because of the different experimental environments of each algorithm, the contrast images could differ from the original ones. However, the overall trend of the comparison results would not be affected. In order to ensure the rationality and objectivity of the experimental results, two representative datasets were selected to test and contrast the images with rich texture details. The testing results are presented in Figures 5–7, which compare the results of the bicubic interpolation, A+, SRCNN, and EEDS methods for different reconstruction times of the butterfly image, zebra image, and comic image, and select the whole panorama and more obvious parts of the wing texture of the butterfly, the head markings of the zebra, and the cheek and shoulder of the comic. A subjective visual evaluation was carried out.

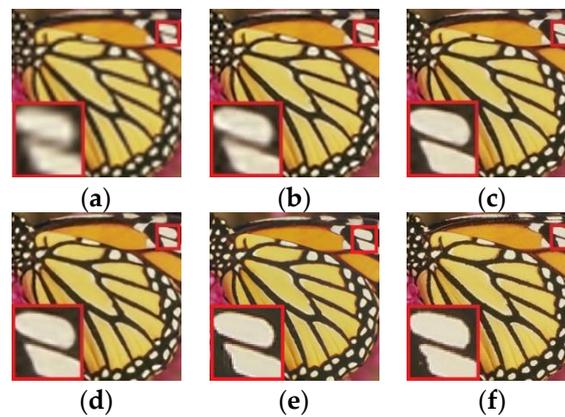


Figure 5. Super-resolution reconstruction results of the image “Butterfly” with “×4” scale factor. (a) Bicubic [5]; (b) A+ [11]; (c) SRCNN [18]; (d) EEDS [20]; (e) The Proposed Method’s; (f) Original Image.

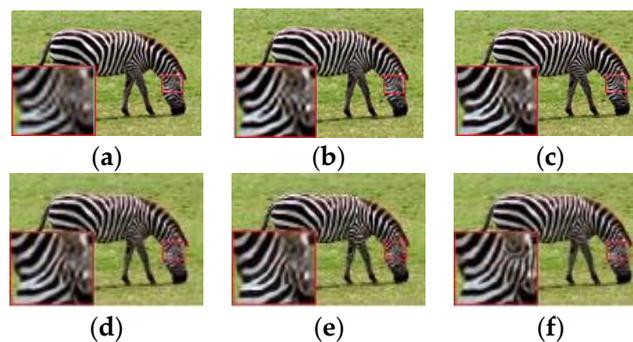


Figure 6. Super-resolution reconstruction results of the image “Zebra” with “×3” scale factor. (a) Bicubic [5]; (b) A+ [11]; (c) SRCNN [18]; (d) EEDS [20]; (e) The Proposed Method; (f) Original Image.

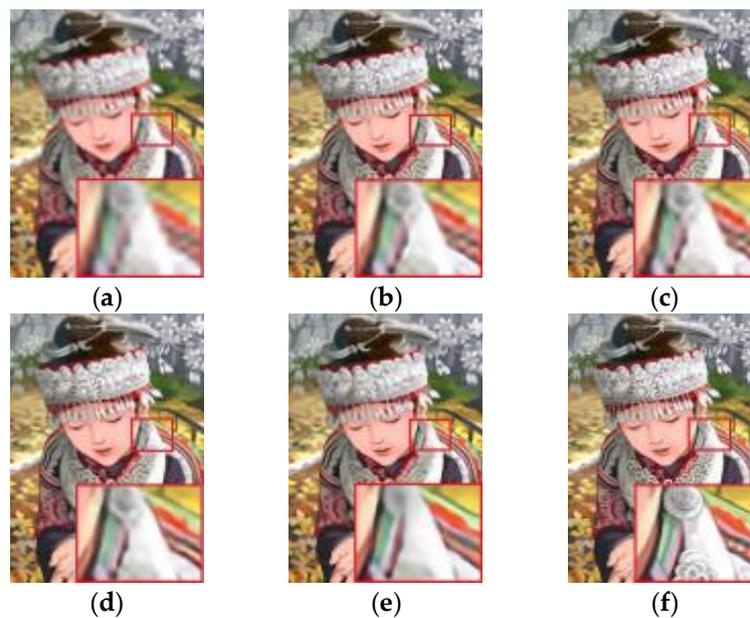


Figure 7. Super-resolution reconstruction results of the image “Comic” with “ $\times 3$ ” scale factor. (a) Bicubic [5]; (b) A+ [11]; (c) SRCNN [18]; (d) EEDS [20]; (e) The Proposed Method; (f) Original Image.

Figure 5a–d present four super-resolution images of four contrast models from left to right. Figure 5e is the result of reconfiguration. Figure 5f concentrates on the Set5 testing of the original HR image. The butterfly wing edge of the image produced by the proposed method is sharper relative to the other methods: both the edge and the image are more complete, and the texture is also clearer.

In Figures 6 and 7, from left to right, the reconfiguration of the four contrast models is also three times that of the super-resolution effect diagram. Figure 6e presents the reconstruction result of the proposed method, while Figure 6f is the original HR image of the Set14 testing dataset. It was found that the reconstruction effect of the zebra image was more prominent, the reconstruction of the cheek edge from the comic image was sharper, the edge preservation was better, and the details of the shoulder texture were more abundant. The average PSNR and SSIM objective testing indexes under various experimental conditions are presented in Table 1. The best experimental results in the table are marked in bold.

Table 1. Average Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM) at different reconstruction scales on Set5 and Set14 datasets.

Dataset	Reconstruction Multiple	Bicubic [5]	A+ [11]	SRCNN [18]	EEDS [20]	Proposed DCCNN
		PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM
Set5	$\times 2$	33.64/0.9296	36.55/0.9543	36.67/0.9541	37.30/0.9578	37.43/0.9603
	$\times 3$	30.38/0.8681	32.57/0.9089	32.76/0.9091	33.46/0.9190	33.59/0.9204
	$\times 4$	28.41/0.8106	30.29/0.8602	30.49/0.8627	31.15/0.8782	31.32/0.8842
Set14	$\times 2$	30.23/0.8687	32.29/0.9058	32.43/0.9062	32.82/0.9104	32.95/0.9115
	$\times 3$	27.54/0.7743	29.14/0.8187	29.29/0.8208	29.61/0.8283	29.70/0.8307
	$\times 4$	26.01/0.7028	27.31/0.7492	27.48/0.7502	27.81/0.7625	28.13/0.7696

As can be seen from the testing results presented in Table 2 below, the results of the proposed algorithm were better than those of the improved algorithm in terms of average PSNR and SSIM, thereby proving the effectiveness of the proposed algorithm.

Table 2. Comparison of computational complexity with phases.

Method	Feature Extraction/ms	Up-Sampling/ms	Reconstruction/ms	Shallow Channel/ms
EEDS	38,015	4112	154,834	7265
DCCNN	19,151	24,895	70,500	5231

4.3. Efficiency Comparison

To further illustrate the effectiveness of the proposed algorithm and evaluate the network performance, the paper analyzed the time complexity [45,46] of the dual channels and compared them in turn with those of the improved network. The specific parameters are shown in Table 2. In the paper, the time complexity of the shallow network is $O(f_1^2 n_1) + O(n_1 f_2^2 n_2) + O(n_2 f_3^2)$, while the time complexity of the deep channel is the same as that of the shallow layer. It can be seen from Table 2 that the amount of parameter computation per iteration was smaller than that of EEDS, meaning that a single iteration training consumed less time. With the same number of iterations, the network training of our proposed model was better than those of SRCNN and EEDS, while the computational complexity of our model was also greatly reduced relative to others. In summary, the efficiency of our proposed method is better than that of the EEDS algorithm.

5. Conclusions

This paper proposed the image super-resolution algorithm based on DCCNN. The deep channel was used to extract the detailed texture information of an image and increase the local receptive field of the image. The shallow channel was mainly used to restore the overall outline of the image. Experimental results showed that the simplified model parameters could not only enhance the ability of the network model to fit the model characteristics but also enable the network model to be trained at a higher learning rate, improving the model's convergence speed. At the same time, the long-term and short-term memory blocks constructed by the residual blocks in the network performed better than the single mapping output network using only the residual blocks. The quantity of image recovery was better, and the performance improved, which proves the necessity of using long-term and short-term memory blocks. Improvement could be observed in both subjective visual effect and objective evaluation parameters, as well as in efficiency, which proves the practicability of the proposed method.

Author Contributions: Conceptualization Y.C.; Methodology, J.W.; Software X.C.; Validation, A.K.S.; Formal Analysis, Y.C.; Investigation J.W.; Resources Z.C.; Data Curation, K.Y. and Z.C.; Supervision J.W. and A.K.S.; Funding Acquisition, J.W. and K.Y. Y.C. provided extensive support in the overall research; J.W. conceived and designed the presented idea, developed the theory, performed the simulations, and wrote the paper; K.Y. and X.C. verified the analytical methods and encouraged to investigate various relevant aspects of the proposed research; A.K.S. and Z.C. provided critical feedback and helped shape the research, analysis, and manuscript. All authors discussed the results and contributed to the final manuscript.

Funding: This research was funded by the National Natural Science Foundation of China [61811530332, 61811540410], the Open Research Fund of Hunan Provincial Key Laboratory of Intelligent Processing of Big Data on Transportation [2015TP1005], the Open Research Fund of Hunan Provincial Science and Technology Innovation Platform of Key Laboratory of Dongting Lake Aquatic Eco-Environmental Control and Restoration [2018DT04], the Changsha Science and Technology Planning [KQ1703018, KQ1706064, KQ1703018-01, KQ1703018-04], the Research Foundation of Education Bureau of Hunan Province [17A007, 16A008], Changsha Industrial Science and Technology Commissioner [2017-7], the Junior Faculty Development Program Project of Changsha University of Science and Technology [2019QJCZ011].

Acknowledgments: We are grateful to our anonymous referees for their useful comments and suggestions. The authors also thank Jingbo Xie, Ke Gu, Yan Gui, Jian-Ming Zhang, Runlong Xia and Li-Dan Kuang for their useful advice during this work.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Gunturk, B.K.; Batur, A.U.; Altunbasak, Y.; Hayes, M.H.; Mersereau, R.M. Eigenface-Domain Super-Resolution for Face Recognition. *IEEE Trans. Image Process.* **2003**, *12*, 597–606. [[CrossRef](#)] [[PubMed](#)]
2. Li, S.; Fan, R.; Lei, G.Q.; Yue, G.H.; Hou, C.P. A Two-Channel Convolutional Neural Network for Image Super-Resolution. *Neurocomputing* **2018**, *275*, 267–277. [[CrossRef](#)]
3. Zhang, L.P.; Zhang, H.Y.; Shen, H.F.; Li, P.X. A Super-Resolution Reconstruction Algorithm for Surveillance Images. *Signal Process.* **2010**, *90*, 848–859. [[CrossRef](#)]
4. Shi, W.Z.; Caballero, J.; Ledig, C.; Zhuang, X.H.; Bai, W.J.; Bhatia, K.K.; Marvao, A.M.M.D.; Dawes, T.; O'Regan, D.P.; Rueckert, D. Cardiac Image Super-Resolution with Global Correspondence using Multi-Atlas Patchmatch. In Proceedings of the 2013 Medical image computing and computer-assisted intervention: MICCAI, Nagoya, Japan, 22–26 September 2013; pp. 9–16.
5. Chen, Y.T.; Wang, J.; Xia, R.L.; Zhang, Q.; Cao, Z.H.; Yang, K. The Visual Object Tracking Algorithm Research Based on Adaptive Combination Kernel. *J. Ambient Intell. Humaniz. Comput.* **2019**, 1–19. [[CrossRef](#)]
6. Chen, Y.T.; Xiong, J.; Xu, W.H.; Zuo, J.W. A Novel Online Incremental and Decremental Learning Algorithm Based on Variable Support Vector Machine. *Clust. Comput.* **2018**, 1–11. [[CrossRef](#)]
7. Zhang, J.M.; Jin, X.K.; Sun, J.; Wang, J.; Sangaiah, A.K. Spatial and Semantic Convolutional Features for Robust Visual Object Tracking. *Multimed. Tools Appl.* **2018**, 1–21. [[CrossRef](#)]
8. Wang, J.; Gao, Y.; Liu, W.; Wu, W.B.; Lim, S.J. An Asynchronous Clustering and Mobile Data Gathering Schema based on Timer Mechanism in Wireless Sensor Networks. *Comput. Mater. Contin.* **2019**, *58*, 711–725. [[CrossRef](#)]
9. Timofte, R.; De Smet, V.; Van Gool, L. Anchored Neighborhood Regression for Fast Example-Based Super-Resolution. In Proceedings of the 2013 IEEE International Conference Computer Vision, Sydney, Australia, 1–8 December 2013; pp. 1920–1927.
10. Timofte, R.; De Smet, V.; Van Gool, L. A+: Adjusted Anchored Neighborhood Regression for Fast Super-Resolution. In Proceedings of the 2014 Asian Conference Computer Vision, Singapore, Singapore, 1–5 November 2014; pp. 111–126.
11. Yang, J.C.; Wright, J.; Huang, T.S.; Ma, Y. Image Super-Resolution as Sparse Representation of Raw Image Patches. In Proceedings of the 2008 IEEE Conference Computer Vision and Pattern Recognition, Anchorage, AK, USA, 24–26 June 2008; pp. 1–8.
12. Yang, J.; Wright, J.; Huang, T.S.; Ma, Y. Image Super-Resolution Via Sparse Representation. *IEEE Trans. Image Process.* **2010**, *19*, 2861–2870. [[CrossRef](#)]
13. Wang, J.; Gao, Y.; Liu, W.; Sangaiah, A.K.; Kim, H.J. An Intelligent Data Gathering Schema with Data Fusion Supported for Mobile Sink in WSNs. *Int. J. Distrib. Sens. Netw.* **2019**, *15*. [[CrossRef](#)]
14. Chen, Y.T.; Xia, R.L.; Wang, Z.; Zhang, J.M.; Yang, K.; Cao, Z.H. The Visual Saliency Detection Algorithm Research Based on Hierarchical Principle Component Analysis Method. *Multimedia Tools Appl.* **2019**, *78*. [[CrossRef](#)]
15. Yang, Y.; Lin, Z.; Cohen, S. Fast image super-resolution based on in-place example regression. In Proceedings of the 2013 IEEE Conference on Computer Vision Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 1059–1066.
16. Zhou, S.R.; Ke, M.L.; Luo, P. Multi-Camera Transfer GAN for Person Re-Identification. *J. Visual Commun. Image Represent.* **2019**, *59*, 393–400. [[CrossRef](#)]
17. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. In Proceedings of the 2012 International Conference Neural Information Processing Systems, Lake Tahoe, NV, USA, 3–6 December 2012; pp. 1097–1105.
18. Dong, C.; Loy, C.C.; He, K.M.; Tang, X.O. Image Super-Resolution using Deep Convolutional Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *38*, 295–303. [[CrossRef](#)] [[PubMed](#)]
19. Dong, C.; Loy, C.C.; He, K.M.; Tang, X.O. Learning a Deep Convolutional Network for Image Super-Resolution. In Proceedings of the 2014 International European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 184–199.
20. Wang, Y.F.; Wang, L.J.; Wang, H.Y.; Li, P.H. End-to-End Image Super-Resolution via Deep and Shallow Convolutional Networks. *IEEE Access* **2019**, *7*, 31959–31970. [[CrossRef](#)]

21. Kim, J.; Lee, J.K.; Lee, K.M. Accurate Image Super-Resolution using Very Deep Convolutional Networks. In Proceedings of the 2016 IEEE Conference on Computer Vision Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1646–1654.
22. Yang, J.X.; Zhao, Y.Q.; Chan, J.C.W.; Yi, C. Hyperspectral Image Classification using Two-Channel Deep Convolutional Neural Network. In Proceedings of the 2016 IEEE International Geoscience and Remote Sensing Symposium, Beijing, China, 10–15 July 2016; pp. 5079–5082.
23. Kim, J.; Lee, J.K.; Lee, K.M. Deeply-Recursive Convolutional Network for Image Super-Resolution. In Proceedings of the 2016 IEEE Conference on Computer Vision Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1637–1645.
24. Ke, R.M.; Li, W.; Cui, Z.Y.; Wang, Y.H. Two-Stream Multi-Channel Convolutional Neural Network (TM-CNN) for Multi-Lane Traffic Speed Prediction Considering Traffic Volume Impact. Available online: <https://arxiv.org/abs/1903.01678> (accessed on 4 June 2019).
25. Lim, B.; Son, S.; Kim, H.; Nah, S.; Lee, K.M. Enhanced Deep Residual Networks for Single Image Super-Resolution. In Proceedings of the 2017 IEEE Conference on Computer Vision Pattern Recognition, Workshops, Honolulu, HI, USA, 21–26 July 2017; pp. 1132–1140.
26. Tai, Y.; Yang, J.; Liu, X.M.; Xu, C.Y. MemNet: A Persistent Memory Network for Image Restoration. In Proceedings of the 2017 IEEE International Conference Computer Vision, Venice, Italy, 22–29 October 2017; pp. 4549–4557.
27. Asvija, B.; Eswari, R.; Bijoy, M.B. Security in Hardware Assisted Virtualization for Cloud Computing—State of the Art Issues and Challenges. *Comput. Netw.* **2019**, *151*, 68–92. [[CrossRef](#)]
28. Zhou, S.W.; He, Y.; Xiang, S.Z.; Li, K.Q.; Liu, Y.H. Region-Based Compressive Networked Storage with Lazy Encoding. *IEEE Trans. Parallel Distrib. Syst.* **2019**, *30*, 1390–1402. [[CrossRef](#)]
29. Min, X.; Ma, K.; Gu, K.; Zhai, G.; Wang, Z.; Lin, W. Unified Blind Quality Assessment of Compressed Natural, Graphic, and Screen Content Images. *IEEE Trans. Image Process.* **2017**, *26*, 5462–5474. [[CrossRef](#)] [[PubMed](#)]
30. Gu, K.; Zhai, G.T.; Yang, X.K.; Zhang, W.J. Using Free Energy Principle for Blind Image Quality Assessment. *IEEE Trans. Multimed.* **2015**, *17*, 50–63. [[CrossRef](#)]
31. He, K.M.; Zhang, X.Y.; Ren, S.Q.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE International Conference on Computer Vision Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
32. Nair, V.; Hinton, G.E. Rectified Linear Units Improve Restricted Boltzmann Machines. In Proceedings of the 2010 International Conference on Machine Learning, Haifa, Israel, 21–24 June 2010; pp. 807–814.
33. He, K.M.; Zhang, X.Y.; Ren, S.Q.; Sun, J. Delving Deep into Rectifiers: Surpassing Human-Level Performance on Imagenet Classification. In Proceedings of the 2015 IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1026–1034.
34. Sun, J.; Xu, Z.B.; Shum, H.Y. Image Super-Resolution using Gradient Profile Prior. In Proceedings of the 2008 IEEE International Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA, 24–26 June 2008. [[CrossRef](#)]
35. Nah, S.; Kim, T.H.; Lee, K.M. Deep Multi-Scale Convolutional Neural Network for Dynamic Scene Deblurring. In Proceedings of the 2016 IEEE Conference on Computer Vision Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 257–265.
36. Zeiler, M.D.; Fergus, R. Visualizing and Understanding Convolutional Networks. In Proceedings of the 2014 European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 818–833.
37. Goodfellow, I.; Pouget-Adadie, J.; Mirza, M.; Xu, B.; Farley, D.W.; Ozair, S.; Courville, A.; Bengio, Y. Generative Adversarial Nets. In Proceedings of the 2014 Advances in Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014; pp. 2672–2680.
38. Timofte, R.; Agustsson, E.; Gool, L.V.; Yang, M.H.; Zhang, L.; Lim, B.; Son, S.; Kim, H.; Nah, S.; Lee, K.M.; et al. Ntire 2017 challenge on single image super-resolution: Methods and results. In Proceedings of the 2017 IEEE Conference on Computer Vision Pattern Recognition, Workshops, Honolulu, HI, USA, 21–26 July 2017; pp. 1110–1121. [[CrossRef](#)]
39. Bevilacqua, M.; Roumy, A.; Guillemot, C.; Alberi-Morel, M.L. Low-Complexity Single-Image Super-Resolution Based on Nonnegative Neighbor Embedding. In Proceedings of the 2012 British Machine Vision Conference, Northumbria University, Newcastle, UK, 3–6 September 2018; pp. 135.1–135.10.

40. Zeyde, R.; Elad, M.; Protter, M. On Single Image Scale-up using Sparse-Representations. In Proceedings of the 2010 International Conference Curves and Surfaces, Avignon, France, 24–30 June 2010; pp. 711–730.
41. Choi, S.Y.; Dowan, C. Unmanned Aerial Vehicles using Machine Learning for Autonomous Flight; State-of-The-Art. *Adv. Rob.* **2019**, *33*, 265–277. [[CrossRef](#)]
42. Gao, G.W.; Zhu, D.; Yang, M.; Lu, H.M.; Yang, W.K.; Gao, H. Face Image Super-Resolution with Pose via Nuclear Norm Regularized Structural Orthogonal Procrustes Regression. *Neural Comput. Appl.* **2018**, 1–11. [[CrossRef](#)]
43. Chen, Y.T.; Wang, J.; Chen, X.; Zhu, M.W.; Yang, K.; Wang, Z.; Xia, R.L. Single-Image Super-Resolution Algorithm Based on Structural Self-Similarity and Deformation Block Features. *IEEE Access* **2019**, *7*, 58791–58801. [[CrossRef](#)]
44. Hong, P.L.; Zhang, G.Q. A Review of Super-Resolution Imaging through Optical High-Order Interference. *Appl. Sci.* **2019**, *9*, 1166. [[CrossRef](#)]
45. Pan, C.; Lu, M.Y.; Xu, B.; Gao, H.L. An Improved CNN Model for Within-Project Software Defect Prediction. *Appl. Sci.* **2019**, *8*, 2138. [[CrossRef](#)]
46. Yin, C.Y.; Ding, S.L.; Wang, J. Mobile Marketing Recommendation method Based on User Location Feedback. *Human-Centric Comput. Inf. Sci.* **2019**, *9*, 1–17. [[CrossRef](#)]



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).