Shuffle block SRGAN for face image super-resolution reconstruction

Ziwei Zhang¹, Yangjing Shi¹, Xiaoshi Zhou¹, Hongfei Kan² and Juan Wen¹

Abstract
When low-resolution face images are used for face recognition, the model accuracy is substantially decreased. How to recover high-resolution face features from low-resolution images precisely and efficiently is an essential subtask in face recognition. In this study, we introduce shuffle block SRGAN, a new image super-resolution network inspired by the SRGAN structure. By replacing the residual blocks with shuffle blocks, we can achieve efficient super-resolution reconstruction. Furthermore, by considering the generated image quality in the loss function, we can obtain more realistic super-resolution images. We train and test SB-SRGAN in three public face image datasets and use transfer learning strategy during the training process. The experimental results show that shuffle block SRGAN can achieve desirable image super-resolution performance with respect to visual effect as well as the peak signal-to-noise ratio and structure similarity index method metrics, compared with the performance attained by the other chosen deep-leaning models.

Keywords
GAN, image super-resolution reconstruction, Shuffle block, face image

Date received: 12 May 2020; accepted: 17 June 2020

Introduction
With the development of face recognition techniques, face image collection has become an important task. However, in an application system, some factors may have remarkable impacts on image quality, such as the image capture device, the illumination level of image, the distance between target and lens, the motion of the target, etc. Face image recognition results can be significantly influenced by using low resolution (LR) with unclear face details. Image super-resolution (SR) reconstruction technology, which aims to convert an LR image to a high-resolution (HR) image, is a useful approach to enhance the quality of face images.

There are three main types of image SR methods: interpolation schemes, reconstruction schemes, and learning-based schemes. Early image SR methods are mainly based on interpolation. The most common ones are Nearest Neighbor Interpolation, Bilinear Interpolation, Bicubic Interpolation, nonuniform sampling interpolation, etc.¹ Interpolation-based methods have been widely used because they are simple to implement. However, the linear characteristics of these models inhibit their ability to reconstruct high-frequency details. Especially when the scaling factor becomes large, the reconstruction effect decreases dramatically. The reconstruction-based method is inspired

¹College of Information and Electrical Engineering, China Agricultural University, Beijing, China
²School of Information Science and Technology, University of International Relations, Beijing, China

Corresponding author:
Juan Wen, College of Information and Electrical Engineering, China Agricultural University, Beijing 100083, China.
Email: wenjuan@cau.edu.cn

Creative Commons CC BY: This article is distributed under the terms of the Creative Commons Attribution 4.0 License (https://creativecommons.org/licenses/by/4.0/) which permits any use, reproduction and distribution of the work without further permission provided the original work is attributed as specified on the SAGE and Open Access pages (https://us.sagepub.com/en-us/nam/open-access-at-sage).
raising interest among the researchers recently. Yang et al. establish a sparse dictionary between LR and HR through training data, which can be used to generate the SR image patch. The disadvantage of the algorithm is that there is no sparse correlation between HR and LR image blocks. It is difficult to select the appropriate samples, so the running time is relatively long. Wang et al. employ weighted sparse regularization to face reconstruction. They introduce neighborhood embedding (NE) from LR and HR image manifolds to retrieve very LR face images. Rasti et al. introduce an SR scheme for single images. The illumination between adjacent patches can be consistent by taking the minimum distance between the LR patches. The learning-based method breaks the limitation of Nyquist sampling frequency. Even if the sampling rate is less than or equal to 2 times of Nyquist frequency, the original image can still be recovered. However, the computational complexity is high, which is not conducive to practical application. Moreover, when the input image structure does not match the test image structure, the reconstruction quality can be reduced.

As we know, the convolutional neural network (CNN) shows a compelling capability in many computer vision tasks lately. Simultaneously, CNN-based image SR models have gradually demonstrated outstanding performance. Zhang et al. set up a residual dense network, extracting hierarchical features of LR image to learn deep features and reconstruct the SR image. To enhance the SR effect and reduce the training time, Mao et al. introduce an encoder-decoder network with deep residual structure for image SR reconstruction by adding symmetric skip connections between the convolution layers. Luo et al. propose a robust CNN model for the satellite image SR task. To improve the visual effect of SR image, Johnson et al. use perceptual loss function and a pretrained 16-layer Visual Geometry Group model to train a feed-forward network for image transfer and SR tasks. Dong et al. trained an SRCNN to establish an end-to-end mapping between LR and HR images. This method combines the traditional sparse coding with the SR method. Although it shows a better reconstruction effect, SRCNN does not get a better effect while deepening the network layers. Also, it is not suitable for multiscale reconstruction.

CNN-based schemes have been victorious in image SR tasks. However, most of these models are optimized by reducing the mean square error (MSE), which will cause some problems, such as the unclear details and unreal visual perception of the reconstructed image. Generative adversary networks (GANs), on the other hand, is a new framework of unsupervised learning. It uses an adversarial strategy to learn high-level image features, which is proved to be beneficial for high-quality image feature extraction. The SRGAN introduced by Ledig et al. is shown to be the state-of-the-art framework to realize the photorealistic SR image reconstruction performance. In this study, a face image SR reconstruction network is proposed based on adversarial learning. To further reduce the time complexity of SRGAN, our model optimizes the SRGAN architecture by replacing the residual block with shuffle block that consists of multiple shuffle units, each of which contains channel shuffle, pointwise group convolution, and depthwise separable convolution layers. We name this SR model as shuffle block SRGAN (SB-SRGAN). The efficient convolution design can significantly reduce the computational complexity while maintaining extraordinary model ability. Furthermore, the loss function is renewed to improve the SR image quality. We not only consider the content loss and the adversarial loss of the generated image by comparison to the natural HR image but also analyze the clarity of the generated image itself. In addition, to make full use of the comprehended knowledge of images features, we utilize transfer learning during the training process. Weights and bias are initialized from a pretrained model on a small image set and then fine-tuned by other datasets throughout the training process.

Our contribution in this paper is threefold:

- First, SB-SRGAN is proposed for face image SR tasks based on the improved SRGAN structure. Instead of using the residual connection, SB-SRGAN uses shuffle unit connection with group convolution layer and depthwise convolution (DWConv) layer to extract more reliable image features and train network effectively.
- Second, the content loss in the loss function of SB-SRGAN is upgraded. Content loss focuses on the generated image quality as well as the similarity between the original image and the generated image. In contrast, the adversarial loss focuses on the authenticity of the generated face image.
- Third, transfer learning is implemented during the training process. The initial weights and bias trained in a small image set are used for model fine-tuning on a large image set to accelerate model convergence.

This study is arranged as follows. Section “Related work” briefly reviews the model structure and loss function of the traditional GAN. The details of SB-SRGAN for face image SR are specified in section “Proposed method.” Experimental results and evaluations are given and analyzed in section “Experimental results.” Finally, we draw conclusions based on our work in section “Conclusion.”

Related work
The application of GAN
Since Good Fellow and others put forward GAN, researchers have done a lot of work based on it. GAN
is proved to be a superior network for learning image feature representations adversarially to generate visually attractive samples. However, GAN is usually challenging to train, which often leads to mode collapse or meaningless output. Since then, how to stably train a GAN model becomes an important issue when implementing GAN. Radford et al. introduced a deep convolution GAN and proposed some guidelines to improve the training stability of GAN. Salimans et al. introduced feature matching, small batch discrimination, and other methods to enhance classification performance through semisupervised learning strategy. Gulrajani et al. proposed a training strategy on the basis of the Wasserstein GAN (WGAN). By penalizing the gradient norm of discriminator, the network can generate higher-quality image samples and achieve faster model convergence. Karras et al. proposed a progressive training method, which gradually increased layers in the generator and discriminator as the training progressed, to generate HR images starting from a low resolution. Schlegl et al. conducted unsupervised learning to distinguish irregularities as samples to be marked. Dong et al. proposed an unsupervised method to change images from one domain into another without referring to the original image. Zhong et al. proposed a high-quality SR for face image based on generative adversarial networks. Yuan et al. carried out an image SR reconstruction experiment by using the GAN structure without LR/HR pairs. In this structure, noise and blur input is first mapped to clean LR image space, and then mapped to the intermediate image, which is sampled by the fine-tuning EDSR model. The reconstructed HR image output is obtained. Ledig et al. proposed SRGAN, which can generate natural SR images by using perceptual loss.

Each training iteration can be separated into two steps. The first one is to train discriminator $D$. The previous image generated from $G$ and a real image are paired up as input data. The expected output of the fake image is 0, and the expected output of the real image is 1. The ultimate goal of model optimization is to minimize the deviation between model output and the expected value.

The second step is to train generator $G$. During this process, $G$ and $D$ are considered as a whole. The SR image obtained by $G$ is fed to $D$ get a probability. At this step, the expected output of $D$ is 1, which means that $D$ thinks the image is a real one. Then, the difference between the probability obtained by $D$ and the expected output 1 is used for error back propagation. This time, only the parameters of $G$ could be updated, while the parameters of $D$ remain unchanged.

The optimization objectives of GAN can be simply summarized in equation (1)

$$
\min_{G} \max_{D} \mathbb{V}(D, G) = \mathbb{E}_{x \sim \text{Preal}}[\log D(x)] + \mathbb{E}_{y \sim \text{pgen}}[\log (1 - D(G(y)))]
$$

where $\mathbb{V}(D, G)$ is the value to be minimaxed, $\text{Preal}(x)$ is the probability distribution of the real image sample and $\text{pgen}(x)$ is the probability distribution of random noise. $G$ is expected to generate images that cannot be distinguished by $D$, whereas $D$ is expected to effectively identify all fake images.

Equation (1) shows that the optimal model only can be achieved when the condition $P_{\text{real}(x)} = P_{\text{gen}(x)}$ is met, where $P_{\text{gen}(x)}$ is the probability distribution of fake data produced by the generator. The optimal $D^*(x)$ is shown in equation (2)

$$
D^*(x) = \frac{P_{\text{real}(x)}}{P_{\text{real}(x)} + P_{\text{gen}(x)}}
$$

**Proposed method**

**The network structure of SB-SRGAN**

The generator $G$ designed in this paper receives an LR face image as an input and generates an HR image after reconstruction. The purpose of $G$ is to bewilder the discriminator $D$ to consider that it is a natural HR image. Comparatively, the training purpose of the discriminator is to tell the real HR image from the generated SR one precisely. The proposed SB-SRGAN architecture is illustrated in Figure 2.

As can be seen from Figure 2, the generator $G$ contains multiple shuffle blocks that share the same structure. Each shuffle block is composed of two convolution layers: a shuffle unit and a batch normalization (BN) layer. We apply shuffle block connections in this network. Shuffle block connection is a shortcut connection similar to residual connection. By adding shortcut connections, the network is more comfortable
to be optimized. The kernel size of the convolution layer is 3×3. In this network, a deconvolutional layer and an upsampling layer are applied to enhance the image resolution.

The shuffle unit shown is first proposed and applied in Zhang et al.24 The shuffle block in this paper is similar to the SRGAN residual block structure and is improved by adding the shuffle unit. The structure of the shuffle unit is shown in Figure 3. This is a residual unit with four layers: first is 1×1 group convolution, followed by a channel shuffle operation.
Then, there is a $3 \times 3$ DWConv layer, mainly to reduce the amount of calculation, followed by the $1 \times 1$ group convolution, and finally a short-circuit connection, which adds the input directly to the output.

The discriminator $D$ is fed with the HR image as well as the SR image generated by the generator. $D$ also contains several blocks with the same structure. Each block consists of a convolution layer, a BN layer, and a LeakyReLU activation function layer. As the network gets deeper, more and more facial image features can be obtained with a decreased features scale. The discriminator $D$ uses LeakyReLU as the activation function without max pooling. After two FC layers, a LeakyReLU activation function, and a sigmoid activation function layer, the discriminator will end up generating a probability score, which can be used to classify the natural HR image from the generated SR image.

**Loss function**

The traditional SR model chooses MSE between pixels as the optimization object defined as

$$loss = \frac{1}{p^X \cdot p^Y} \sum_{i=1}^{p^X} \sum_{j=1}^{p^Y} (H_{i,j} - G(p^{LR}_{i,j}))^2$$

where $p^{LR}$ denotes the input LR image, which is acquired by downsampling the original HR image $H^{HR}$ using factor $p$. $X$ and $Y$ are the width and height of HR image, respectively, so $pW$ and $pH$ represent the width and height of $p^{LR}$, respectively. This loss function only examines the difference between the input and output images without considering the SR image quality.

Therefore, in this paper, we combine MSE loss between pixels and image quality measurement as the loss function. To summarize, the loss function is defined as the weighted sum of content loss and adversarial loss, as shown below

$$l_{SB-SRGAN} = l_c + \alpha l_{adv}$$

where $l_c$ is the content loss, $l_{adv}$ denotes the adversarial loss, and $\alpha$ is the balancing parameter. The content loss is related to the quality of SR image and the similarity of SR and HR images, shown in equation (5)

$$l_c = \frac{1}{p^X \cdot p^Y} \sum_{i=1}^{p^X} \sum_{j=1}^{p^Y} (H^{HR}_{i,j} - G(u(p^{LR}_{i,j})))^2 + \lambda \frac{1}{C(I^{SR})}$$

where

$$C(I^{SR}) = \sum_{i=1}^{p^X} \sum_{j=1}^{p^Y} |G(u(p^{LR}_{i,j}))(i + 2, j) - G(u(p^{LR}_{i,j}))(i, j)|$$

The first part of content loss is based on MSE, while the second part considers the clarity of the SR image $C(I^{SR})$. Using the Brenner gradient function, the gray-level difference between two adjacent pixels in $I^{SR}$ is calculated, which can be used to evaluate the clarity of the generated image $I^{SR}$. $\lambda$ is the balancing parameter of the two parts.

The adversarial loss $l_{adv}$, on the other hand, depends on the parameters of the discriminator $D_u$. After transferring the LR image to the SR image, the discriminator tries to classify the HR image and the original HR one. To achieve a better visual perception effect, it is essential to generate a data distribution that cannot be distinguished by the discriminator. The adversarial loss function is presented in equation (7)

$$l_{adv} = \sum_{K=1}^{K} - \log D_u(G(u(p^{LR})))$$

where $K$ denotes the number of training images.

**Adversarial training**

In this paper, we use content loss and adversarial loss to restore image resolution. The discriminator $D_u$ is trained along with the generator $G_u$ in an alternate way, as shown in equation (1).

The purpose of the adversarial training is, on the one hand, to maximize the discriminator’s ability to identify whether the training samples are real or fake during the training process and, on the other hand, to improve the
capacity of the generator by generating indistinguishable images. As the training steps go on, the generator enhances the ability to create realistic SR images, while the discriminator develops the strength to identify the generated image and HR image simultaneously.

**Experimental results**

**Face image dataset**

The size and type of training set will directly affect the parameters of a model, and then ultimately have some influence on the quality of the reconstructed face image. To investigate the experimental results of SB-SRGAN on different datasets, we use three different face image datasets for training and testing, Wider Face, CelebA, and LFW (Labeled Faces in the Wild (LFW)). All of them are public data sets. Models are trained separately for each face database.

Wider face, including 32,203 images and 393,703 labeled faces. Forty percent of the images are train images with HR, all zoomed to 1024 in width. The minimum annotation face image is $10 \times 10$. Test images account for 50% of the total, with an average of 12.2 faces per image. Validation images account for 10%.

CelebA is a large face attribute dataset provided by the Chinese University of Hong Kong. The dataset contains more than 200,000 celebrity images; each of them has 40 attribute labels. Most of the images have a messy background and carry a variety of human poses. To sum up, CelebA includes 10,177 characters, 202,599 faces, and five human face positions.

The LFW face image dataset is a public benchmark for face verification. It contains 13,233 $250 \times 250$ images with unconstrained natural scenes. These images are not collected by some photographers but are composed of random images obtained from the Internet. There are 5749 people in the dataset, where 1680 of them have multiple distinct images with different scenes.

The number of samples we used in our experiment in the three face image datasets is shown in Table 1. The number of validation images is one-quarter of the number of train images. After a certain number of face images are extracted to form the three different datasets, the images selected should be normalized to make the pixel value change the range from 0–255 to $2^{11}$.

**Table 1.** Number of samples used in different datasets.

|               | # of train samples | # of val samples | # of test samples |
|---------------|--------------------|------------------|-------------------|
| Wider Face    | 12,000             | 3000             | 15,000            |
| CelebA        | 30,000             | 7500             | 20,000            |
| LFW           | 5000               | 1250             | 6250              |

LFW: labeled faces in the wild.

**Experiment setup**

The platform used in these experiments is based on Windows 10 operating system, with GTX 1060 GPU to accelerate. The algorithm is implemented by Python 3.5 under Tensorflow framework.

To make sure the network can extract better face features, we use a face detection network to locate human face so that the faces can stay in the fixed position of the image and hasten model convergence. MTCNN is a robust face detection algorithm, which can not only detect the face of the image but also automatically achieve face truncation, face alignment, and face size specification.

After preprocessing by MTCNN, we transfer the original images in the face image datasets into HR face images that focus only on the human face. Then, we apply the bicubic algorithm to preprocess the HR face images with a downsample scaling factor of $\times 4$ to form a corresponding LR face image dataset.

An example of image preprocessing is displayed in Figure 4. Figure 4(a) is an original HR image from the face image dataset, and Figure 4(a) is the preprocessed HR face images generated by MTCNN. Doing so allows the network to extract facial features better and get a better reconstruction result.

The batch size is set to 32. $\alpha$ and $\lambda$ in equations (4) and (5) are set to be 0.001 and 0.5, respectively, according to the experience in SRGAN and the results in preexperiment. During the training process, first, random initialization is used to set the values for the weights and bias with batch size 32. Training is stopped after 500 epochs to get the first-stage weights. All the layers except the last three layers of the generator and discriminator are frozen, and training is continued for 100 epochs. All the layers are unfrozen, and training is continued for the rest of the epochs; the learning rate is dynamically adjusted according to the change rate of loss. We set the maximum number of iterations as 100,000. Use the early-stopping strategy to stop the training process when the loss is stable.
In the process of model training, the adjustment of weight is realized by the gradient descent algorithm. Adam (Adaptive Moment Estimation) optimizer is used with $\beta = 0.9$ in the generator and $\beta = 0.99$ in the discriminator. When the reconstruction of an SR image is completed, the difference between the reconstructed SR image and the original HR image needs to be calculated using the objective evaluation index. The feature extraction network used in this paper consists of several shuffle block layers. The number of shuffle blocks is a variable, which needs to be obtained in later experiments.

**SR image quality evaluation metrics**

The SR image quality evaluation standard needs to be calculated with a specific mathematical formula to numerically quantify the reconstructed SR image quality. A good evaluation standard must first be consistent with the judgment by the vision of the human eye. In addition, whether the SR image is good should be evaluated by calculating the difference between the HR image and the reconstructed SR image. The peak signal-to-noise ratio (PSNR) and structure similarity index method (SSIM) are two commonly used metrics for evaluating the reconstructed SR image quality.

PSNR quantifies the quality of a reconstructed or corrupt image with reference to the ground-truth. The higher the PSNR, the smaller the distortion after reconstruction. PSNR is generally calculated by MSE. The formula of PSNR (in dB) is shown in equation (8)

$$PNSR = 10 \log_{10}\left(\frac{\text{peak}^2}{\text{MSE}}\right)$$

where `peak` is the maximum possible pixel in the image, and its value depends on the sampling interval. If using linear PCM with $k$ bits per sample, the `peak` value is $2^k - 1$, for example, the sampling interval is 3, and the `peak` value is 7. The calculation formula is shown in equation (9)

$$\text{MSE} = \frac{1}{\alpha \beta} \sum_{i=0}^{\alpha-1} \sum_{j=0}^{\beta-1} (HR(i,j) - SR(i,j))^2$$

where $\alpha$ and $\beta$ represent the length and width of the image, respectively, and HR and SR denote the original HR image and the reconstructed SR image, respectively.

PSNR is sensitive to the reconstruction error because its calculation is based on the erroneousness between these corresponding pixels of two images. However, sometimes PSNR is inconsistent with human visual perception. For example, the human eye is more sensitive to the difference in the low-frequency region of the image than in the high-frequency region, and the perception result of the human eye to an area will be influenced by the adjacent region, etc. Accordingly, PSNR often seems inconsistent with human feelings in the image quality aspect.

SSIM is another measure designed to evaluate image quality from a distinct perspective. Because the original HR image has a similar structure with the reconstructed image, the reconstruction result can be evaluated according to the structure similarity. SSIM mainly combines image similarity in the aspect of contrast, brightness, and image structure. SSIM has frequently proved that its performance is much better than MSE and other metrics, and it has gradually become a general standard for evaluating the quality of reconstructed images.

The value of SSIM is between 0 and 1. Larger SSIM indicates better image quality, as well as a higher structural similarity between the target image and the reference image. The calculation formula of SSIM is shown in equation (10)

$$\text{SSIM}(T, R) = \frac{(2\tau_T \tau_R + a)(2\sigma_{TR} + b)}{(\tau_T^2 + \tau_R^2 + a)(\sigma_T^2 + \sigma_R^2 + b)}$$

where $T$ and $R$ are the target image and the reference image, respectively; $\tau_T$ and $\tau_R$ represent the mean values of image $T$ and $R$ with respect to brightness, respectively; $\sigma_{TR}$ is the covariance of the two images; $\sigma_T$ and $\sigma_R$ are the standard deviations of the two images; and $a$ and $b$ are constant coefficients.

**The influence of the number of shuffle blocks on the reconstruction performance**

ShuffleNet is proposed by Megvii Inc., a company owns the world’s leading AI algorithm, face recognition software, and hardware. ShuffleNet applies a CNN architecture that extremely computation efficient. As a result, it can even be used in mobile devices. One good advantage of ShuffleNet is the pointwise group convolution layer and channel shuffle. This dramatically reduces the calculation of the model while maintaining sharpness. The basic unit is improved on the basis of a residual unit. It manages to obtain lower top-1 error on ImageNet classification and achieves 13 times actual speedup as many as AlexNet while maintaining comparable accuracy.

In this section, we carry out experiments to decide how many shuffle blocks should be employed in the SB-SRGAN and to see how does it affect the SR image quality. The number of shuffle blocks is set from 1 to 30, with the structure GConv + channel shuffle + DWConv + GConv + ReLU. The convolution layer and the deconvolution layer before and after the shuffle connection structure both contain 64 convolution kernels with a size of $3 \times 3$. The last convolution layer contains three convolution kernels with a size of $3 \times 3$, corresponding to the three RGB channels of the generated image. Adam is selected as the optimization function, which basically adds the momentum term Root Mean Square prop (RMSprop) to deal with the radically diminishing learning rates. Adam adjusts the parameter learning rate by using gradient first-
order and second-order moment estimation. Different from the traditional Stochastic Gradient Descent algorithm (SGD), Adam applies the bias correction strategy, which can make sure that the learning rate of each iteration has a stable change rate to ensure the stability of the updated parameters.

For the convenience of model training, we set the Adam learning rate to 0.01 and the batch size for each training step to 32. We randomly pick 2000 images from each of the three face datasets to form a mixed dataset with 6000 face images. Then, we divide the mixed dataset into the training set, validation set, and testing set according to the proportions of 50%, 10%, and 40%, respectively. The impact of the number of different shuffle blocks on the PSNR and SSIM of SR images is demonstrated in Figures 5 and 6.

As can be seen from Figures 5 and 6 that with the growth of the number of shuffle blocks, the values of PSNR and SSIM also change to different degrees. When the number of shuffle blocks is small, the trained SB-SRGAN model is vulnerable. When the number gets too large, the PSNR and SSIM can also decline. It is because when the number of shuffle blocks is too small, the advantages of the model cannot be reflected. While the number is too big, it will lead to an enormous solution space of the model, which means the model is difficult to converge to its optimal.

As the number of shuffle blocks increases, the PSNR and SSIM of the model begin to increase. When the number of shuffle blocks increases to 18, PSNR reaches the maximum. Then, with the increase in the number, PSNR began to decrease slightly. For SSIM, when the number of shuffle blocks reaches 20, the SR reconstruction of face image achieves the best result. However, SSIM under 18 shuffle blocks and 20 shuffle blocks are close. As a result, the block number we choose in our experiment is 18.

The influence of the number of training samples on the reconstruction performance

In this section, to study how the size of the training set affects the performance of SB-SRGAN, experiments are carried out on different training sizes. As shown in Figure 7, when the amount of the training set changes from small to large, the SR reconstruction effect of face image will be enhanced. When the face image data set size reaches a certain degree, the growth rate becomes flat, and the reconstruction curve based on PSRN becomes smooth. From this experiment, we can infer that the increase in the number of training samples can intensify the model ability to a certain extent. Due to the model training time complexity, it is necessary to consider the appropriate training size.

Face image SR reconstruction results

To test the performance of the SB-SRGAN, we select three deep learning-based image SR reconstruction models for comparative experiments, the traditional GAN model, SRCNN model, and SRGAN. The four models are trained on the preprocessing images sets from the three face image datasets. The number of images in the training set, testing set, and validation set can be found in Table 1. We choose some examples to show the comparison of the SR reconstruction results, which are shown in Figure 8.

As it is demonstrated in Figure 8, we pick different types of face images, including those people with covers (wearing hats), with side face, with different facial
expressions and skin colors, and with distinct clarity of the original image as well. From the left column to the right, it indicates the original HR image, and the down-sampled LR image, SR reconstruction image generated by traditional GAN, SRCNN, and SB-SRGAN. It is apparent to see that the reconstructed face image via SB-SRGAN has a higher image quality with better image resolution and sharp lines and edges in the image. The details of the human face, including the eyebrows, eyes, wrinkles, mouth, hair, and so on, in the face image generated by the SB-SRGAN are more visible than those produced by the other two models. Therefore, SB-SRGAN proposed in this paper is superior to the

|                | Traditional GAN | SRCNN | SRGAN | SB-SRGAN |
|----------------|-----------------|-------|-------|----------|
|                | PSNR (dB)       | SSIM  | PSNR (dB) | SSIM  | PSNR (dB) | SSIM  |
| LFW            | 28.18           | 0.764 | 30.56  | 0.873   | 31.57  | 0.896  |
| Wider Face     | 30.79           | 0.792 | 32.08  | 0.890   | 33.30  | 0.912  |
| CelebA         | 32.68           | 0.813 | 34.59  | 0.914   | 35.72  | 0.942  |
| Average        | 30.55           | 0.790 | 32.41  | 0.892   | 33.53  | 0.911  |

GAN: Generative adversary networks; SRCNN: super-resolution convolution neural network; SRGAN: Super Resolution Generative Adversarial Network; SB-SRGAN: Shuffle block Super Resolution Generative Adversarial Network; PSNR: peak signal-to-noise ratio; SSIM: structure similarity index method; LFW: Labeled Faces in the Wild.

Figure 8. Examples of original HR image, LR image, the generated SR images by traditional GAN, SRCNN, and SB-SRGAN (from left to right).
traditional GAN in terms of visual perception. Also, it is comparable to SRCNN, with more noticeable details.

In addition to the subjective perception evaluation, this paper compares the performance of three models in diverse datasets in regard to PSNR and SSIM. To accelerate the convergence speed of the model, we implement the transfer learning strategy during the training processes. First, an initial model is trained using random initialization on LFW data. Then, the initial model is utilized to initialize the parameters of other models trained on Wider Face and CelebA. All the layers except the last three layers are frozen, and training is continued for 100 epochs. Then, all layers are unfrozen and the training process is continued until the model converges. The performance of the four models on different datasets is presented in Table 2.

As shown in Table 2, compared with the traditional GAN, SRCNN improves PSNR by 1.86 dB on average, with an improvement rate of 6%, and improves SSIM by 0.038, with an improvement rate of 12%. Compared with SRCNN, SB-SRGAN improves PSNR by 1.43 dB on average, with a rate of about 4.41%. And it improves SSIM by 0.027 at a rate of about 3.03%. And compared with SRGAN, SB-SRGAN slightly improves PSNR by 0.93% and SSIM by 0.088%. It can be seen from the results that SB-SRGAN proposed in this paper is much better than the traditional GAN and SRCNN in the SR reconstruction of face image and slightly better than SRGAN. From the perspective of different datasets, the LFW database-based model has the worst performance because it has the least training samples and only uses the method of random initialization. From the experimental results on different datasets, it can be seen that transfer learning improves the performance of the model to some extent. For example, when using LFW as a training set, the model is trained without the transfer learning strategy. Therefore, the performance in LFW is the lowest among the three datasets. When applying transfer learning on Wider Face and CelebA datasets, the model converges speed has increased, and the model performance has also been improved. This is because transfer learning makes the features learned in other datasets reused by fine-tuning in the new datasets.

Conclusion

We have introduced shuffle block GAN, namely SB-SRGAN, which makes use of the shuffle blocks to accomplish efficient image SR reconstruction. We changed the content loss function by considering the generated image clarity. We tested our model on three public image face datasets and evaluated it with SSIM and PSNR metrics. Besides, transfer learning has been applied during the training process. We compared our model to three deep-learning-based SR networks. The experimental results have confirmed the advantage of SB-SRGAN.

Declaration of conflicting interests
The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding
The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by the National Natural Science Foundation of China (No. 61802410), and the Chinese Universities Scientific Fund (2018XD002 & 2018QC024).

ORCID iD
Juan Wen https://orcid.org/0000-0002-4199-2988

References
1. Keys Cubic RG. Convolution interpolation for digital image processing. IEEE T Acoustic Speech Signal Pr 1981; 29: 1153–1160.
2. Li X and Orchard MT. New edge-directed interpolation. IEEE T Image Pr 2001; 10(10): 1521–1527.
3. Stark H and Oskoui P. High resolution image recovery from image-plane arrays. Using Convex Project 1989; 6: 1715–1726.
4. Yang J, Wright J, Huang TS, et al. Image super-resolution via sparse representation. IEEE T Image Process 2010; 19: 2861–2873.
5. Wang Z, Hu R, Wang S, et al. Face hallucination via weighted adaptive sparse regularization. IEEE T Circuit Syst 2014; 24: 802–813.
6. Rasti P, Nasrollahi K, Orlova O, et al. A new low-complexity patch-based image super-resolution. IET Comput Vis 2017; 11: 567–576.
7. Zhang Y, Tian Y, Kong Y, et al. Residual dense network for image superresolution, 2018, https://arxiv.org/abs/1802.08797.
8. Mao X, Shen C and Yang YB. Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections. In: Advances in neural information processing systems, Barcelona, 5–10 December 2016, pp.2802–2810. New York: IEEE.
9. Luo Y, Zhou L, Wang S, et al. Video satellite imagery super resolution via convolutional neural networks. IEEE Geosci Remote Sens Lett 2017; 14: 2398–2402.
10. Johnson J, Alahi A and Li FF. Perceptual losses for real-time style transfer and super-resolution, 2016, https://arxiv.org/abs/1603.08155.
11. Dong C, Loy CC, He K, et al. Image super-resolution using deep convolutional networks. IEEE Trans Pattern Anal Mach Intell 2016; 38: 295–307.
12. Goodfellow I, Pouget-Abadie J, Mirza M, et al. Generative adversarial nets. In: Advances in neural information processing systems NIPS, Montreal, QC, Canada, 8–13 December 2014, pp.2672–2680. New York: ACM.
13. Ledig C, Theis L, Huszar F, et al. Photo-realistic single image super-resolution using a generative adversarial network. In: IEEE computer vision & pattern recognition, Honolulu, HI, 21–26 July 2017, pp.105–114. New York: IEEE.
14. Radford A, Metz L and Chintala S. Unsupervised representation learning with deep convolutional generative adversarial networks, 2016, https://arxiv.org/abs/1511.06434
15. Salimans T, Goodfellow IJ, Zaremba W, et al. Improved techniques for training GANs. In: Advances in Neural Information Processing Systems, Barcelona, 5–10 December 2016, pp.2226–2234. New York: IEEE.
16. Arjovsky M, Chintala S and Bottou L. Wasserstein GAN, 2017, https://arxiv.org/abs/1701.07875
17. Gulrajani I, Ahmed F, Arjovsky M, et al. Improved training of wasserstein GANs. In: Advances in neural information processing systems, Long Beach, CA, 4–9 December 2017, pp.5769–5779. New York: ACM.
18. Karras T, Aila T, Laine S, et al. Progressive growing of GANs for improved quality, stability, and variation, 2018, https://arxiv.org/abs/1710.10196
19. Schlegl T, Seeböck P, Waldstein SM, et al. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In: International conference on information processing in medical imaging, Boone, NC, 25–30 June 2017, pp.146–157. New York: Springer.
20. Dong H, Neekhara P, Wu C, et al. Unsupervised image-to-image translation with generative adversarial networks, 2017, https://arxiv.org/abs/1701.02676
21. Zhong X, Qu X and Chen C. High-quality face image super-resolution based on Generative Adversarial Networks. In: IEEE 4th advanced information technology, electronic and automation control conference, Chengdu, China, 20–22 December 2019, pp.1178-1182. New York: IEEE.
22. Yuan Y, Liu S, Zhang J, et al. Unsupervised image super-resolution using cycle-in-cycle generative adversarial networks. In: The IEEE conference on computer vision and pattern recognition workshops, Salt Lake City, UT, 18–22 June 2018, pp.814–823. New York: IEEE.
23. Lim B, Son S, Kim H, et al. Enhanced deep residual networks for single image super-resolution. In: Computer vision and pattern recognition workshops, Honolulu, HI, 21–26 July 2017, pp.1132–1140. New York: IEEE.
24. Zhang XY, Zhou XY, Lin MX, et al. An extremely efficient convolutional neural network for mobile devices. In: The IEEE conference on computer vision and pattern recognition, Salt Lake City, UT, 18–22 June 2018, pp.6848–6856. IEEE.
25. Yang S, Luo P and Loy CC. From facial parts responses to face detection: a deep learning approach, 2015, https://arxiv.org/abs/1509.06451
26. Large-scale Celeb Faces Attributes Dataset, http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html
27. Labeled faces in the wild, http://vis-www.cs.umass.edu/lfw/
28. Zhang K, Zhang Z, Li Z, et al. Joint face detection and alignment using multitask cascaded convolutional networks. IEEE Signal Pr Lett 2016; 23(10): 1499–1503.
29. Kingma DP and Ba J. Adam: a method for stochastic optimization, 2015, https://arxiv.org/abs/1412.6980
30. Wang Z, Bovik AC and Sheikh HR. Image quality assessment: from error visibility to structural similarity. IEEE Trans Image Process 2004; 13: 600–612.