Multimedia Tools and Applications
https://doi.org/10.1007/s11042-024-18274-0

# KernelFlexSR: a self-adaptive super-resolution algorithm with multi-path convolution and residual network for dynamic kernel enhancement

**Haotian Zhang[1] · Long Teng[1] · Youyi Wang[2] · Hang Qu[3] · Chak-yin Tang[1]**

Received: 3 August 2023 / Revised: 4 January 2024 / Accepted: 11 January 2024


**Abstract**
Machine learning-based image super-resolution (SR) has garnered increasing research interest in recent years. However, there are two issues that have not been adequately addressed. The first issue is that existing SR methods often overlook the importance of improving the quality of the training dataset, which is a crucial factor in determining SR performance, regardless of the training method employed. The second issue is that while some studies report high numerical metrics, the visual results remain unsatisfactory. To address the first problem, we propose a new image down-sampling method to obtain higher-quality training datasets. To tackle the second problem, we present a new image super-resolution model based on a large-size convolution kernel and a multi-path algorithm. Specifically, we use an adaptive large-size convolutional kernel to extract features from the image based on the size of the input image, and a residual network to generate a deeper model to retain more details of the original input image. Experimental results demonstrate that the proposed multilayer downsampling method (MDM) can significantly improve the visual quality compared to traditional downsampling methods. Moreover, our proposed method achieves the best peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) values compared to several typical SR algorithms. Furthermore, subjective evaluation by human observers reveals that our method retains more details of the original image and produces smoother high-resolution images. Our proposed method effectively addresses the two aforementioned issues, which leads to improved SR performance in terms of both quantitative and qualitative measures.

**Keywords** Single-image-super-resolution(SISR) · Resnet · Big-size convolution kernel · Multi-path structure · Deep learning · Computer vision

## 1 Introduction

Image super-resolution (SR) technology is rapidly gaining traction within the research community, particularly due to advancements in machine learning and deep learning. These

✉ Haotian Zhang
21119479r@connect.polyu.hk

Extended author information available on the last page of the article

Published online: 26 January 2024

technologies have been effectively employed in SR, delivering remarkable improvements. SR aims to transform a low-resolution (LR) image into a high-resolution (HR) image, thereby enhancing visual quality, as illustrated in Fig. 1.

HR images offer enhanced detail and clarity, crucial for a variety of devices including computer monitors, high-definition televisions, smartphones, tablets, and cameras. Moreover, SR has diverse applications across several fields, such as object detection in scenes [1, 2], facial recognition in security footage [3], medical imaging [4], remote sensing [5], astronomical imagery [6], and forensic analysis [7].

SR is primarily categorized into two types: single image super-resolution (SISR) and multiple images super-resolution (MISR). SISR focuses on upscaling a single image to a higher resolution with improved texture and detail, while MISR involves using multiple images to generate a high-resolution image.

However, two critical issues persist within this domain. Firstly, many current SR methods overlook the impact of training dataset quality, raising the question: Can a high-quality training dataset enhance model performance? Secondly, some methods may achieve high PSNR/SSIM scores, yet the resulting images do not necessarily appear more visually appealing to human observers. To address these challenges, this paper makes the following contributions:

1. A Multilayer Downsampling Method (MDM): We introduce a hierarchical downscaling approach to construct a higher quality training dataset. This method is designed to generate sharper LR images with enhanced detail, thereby improving the model's learning capability and performance.

2. Adaptive Multi-Path Structure Model: We propose an architectural framework that incorporates convolutional kernels of varying sizes: large, medium, and small. These kernels correspond to perceptual fields and extract features of different scales. Additionally, we integrate a residual network to deepen the model, expanding its learning capacity and effectiveness.

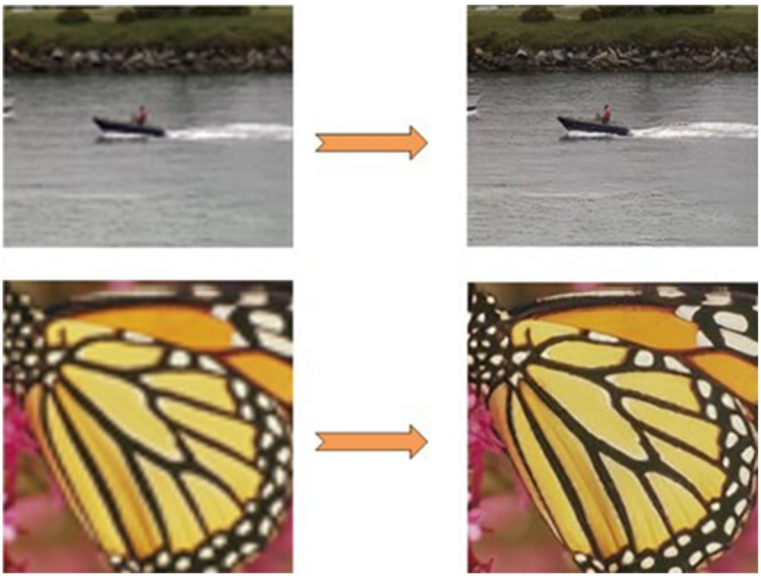

**Fig. 1** The effect of SR

Our method addresses the aforementioned challenges by introducing a more efficient approach for training dataset construction, a structure that incorporates convolutional kernels of various sizes, and perceptual losses into the loss function. The contributions of this work offer a sharper final output image and better visual performance compared to conventional CNN approaches.

The organization of this paper is structured as follows: Section 1 presents the background and context of this research. Section 2 provides a comprehensive review of related works in the field of super-resolution (SR). Subsequently, Section 3 delineates the methodology proposed in this study. Detailed information about the experimental setup and procedures is outlined in Section 4. The findings and outcomes of these experiments are discussed in Section 5. Finally, Section 6 offers conclusions drawn from this research and potential directions for future work.

## 2 Related works

Super-resolution (SR) technology, particularly learning-based approaches, is increasingly becoming a focal point in the SR research community. These methods extensively utilize training data to understand the relationship between low-resolution (LR) and high-resolution (HR) images, enabling the prediction of HR images based on these learned mappings. The rise of machine learning and, more significantly, deep learning, has shifted the paradigm towards learning-based SR methods, which are recognized for their superior performance over traditional techniques. Deep learning, in particular, is celebrated for its robust fitting capabilities, making it a cornerstone algorithm in machine learning [12].

The field of SR underwent a significant revolution in 2014 when Dong et al. introduced the Super-Resolution Convolutional Neural Network (SRCNN). This three-layer CNN utilizes Mean Square Error (MSE) as its objective function and set a new benchmark for SR technology [13]. Building on this, Shi et al. proposed the Efficient Sub-Pixel Convolutional Neural Network (ESPCN), a method that notably enhances real-time SR reconstruction [14]. Another milestone was achieved by Ledig et al. with the introduction of SRGAN, a Generative Adversarial Network (GAN) designed for image reconstruction that introduced a new perceptual objective function [15].

Further developments in the field have been characterized by innovative architectures and algorithms. The Deep Recursive Convolutional Network (DRCN) and Laplacian Pyramid Super-Resolution Network (LapSRN) are two such examples, utilizing recursive convolutional networks and integrating Laplace's pyramid with deep learning, respectively, to provide multi-level SR models [16, 18]. Enhanced Deep Super-Resolution network (EDSR) leverages a residual-based learning mechanism in CNN [23], while the Residual Dense Network (RDN) merges ResNet and DenseNet structures to enhance the loss function, incorporating pre-activation RaGAN and VGG features [29, 31].

Real-ESRGAN has introduced a novel approach by augmenting the complexity of reduced-order images through the introduction of the sinc filter and U-Net discriminator [32]. SR3 has provided a fresh perspective on conditional image generation by employing a denoising diffusion probability model and denoising score matching [33]. Moreover, cross-scale residual networks and MADNet are designed to exploit scale-dependent features and improve correlation learning [36, 37].

In specialized domains like medical imaging, significant strides have been made. DeepVolume offers a tailored two-step deep-learning architecture for accurate thin-section MR

image reconstruction [20], while MFSR introduces a multi-frame SR architecture with an attention-based fusion module, enhancing the detail and quality of medical images [39]. The Texture Transformer Network (TTSR) utilizes attention mechanisms to transfer high-resolution textures [40], and an innovative end-to-end trainable unfolding network has been proposed to address SISR problems with various scale factors, blur kernels, and noise levels [41].

These advancements collectively underscore the dynamic and continually evolving nature of SR research. Ongoing innovation and the integration of new techniques are steadily pushing the boundaries, contributing to the progressive enhancement and refinement of image super-resolution methodologies.

## 3 Methodology

### 3.1 Dataset generation and pre-processing

The initial phase of our study involves constructing a robust training dataset, as the quality of the dataset significantly influences the performance of the super-resolution (SR) model. Typically, existing downsampling methods utilize a singular approach, such as Bicubic, K-nearest, or Pooling methods, to transform high-resolution images into their low-resolution counterparts [35]. This conventional process often results in a substantial loss of detail, a drawback evident in the low-resolution images derived from the Div2k dataset, as depicted in Fig. 2.

From Fig. 2, it is clear that compared to the original image Fig. 2(a), the reshaped image Fig. 2(b) loses many details and has distinctive jaggedness at the edges of objects. How to keep more details of the original image and avoid the distinctive jaggedness at the edge of the object have not been well addressed. We proposed a novel downsampling method called MDM to solve this problem. According to Fig. 3, the downsampling is not completed in a single step, instead, it is divided into multiple steps, and in each step the MDM downsamples the image by half, the process repeats until the targeted size is reached. It should be mentioned that in each step the downsampling method still uses the Bicubic, K-nearest, or Pooling method.

The effects of MDM are shown in Fig. 2(c). It is clear that the MDM method achieves better image quality and keeps more details of the original HR image. Compared to the traditional method, our MDM method has less distinctive jaggedness at the edge of the item. The traditional downsampling methods usually use only a single-layer downsampling. In constrast, the proposed MDM method adopts a multi-layer structure and downsamples the length and width of the image by half each time. Through our method, the output image keeps more details and has less distinctive jaggedness.

In addition, the Div2k [15] dataset is selected to train the SR model in this research, which has been frequently used by existing works [26–31]. It is noted that we only use the Div2k dataset's HR images part, while the LR images are generated by the proposed MDM method for downsampling. The Div2k dataset has 900 images. Each image in this dataset has different sizes. Among all images, 800 images are selected as the training dataset and the remaining 100 images are used as the validation dataset. Specifically, we use the proposed MDM method for downsampling to convert these images into 3*100*100 as the LR images and 3*200*200 as the HR images. Then, we use Pytorch's toTensor function to convert all images to tensor vectors.

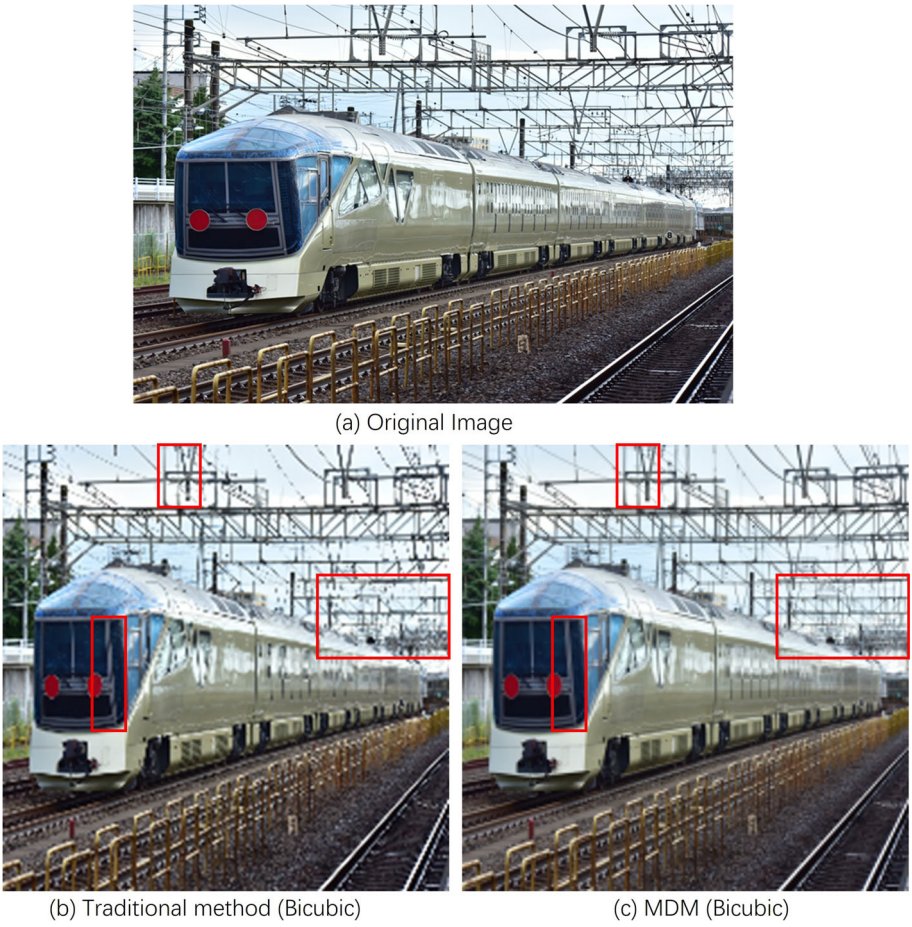


**Fig. 2** Comparison of the Bicubic method and the proposed MDM Bicubic method

### 3.2 Model of SR

In this research, we propose a novel SR algorithm using multi-path convolution kernels and residual networks with different sizes. According to [24], the big-size convolution kernel may result in very good performance. However, it has not been investigated in SR. Moreover, convolution kernels correspond to receptive fields depending on the sizes, which means that convolution kernels with different sizes can extract features with different sizes in one image. Using multiple convolutional kernels of different sizes increases the feature representation capability of the convolutional layer, as each convolutional kernel can extract features of different scales, which helps to improve the model's ability to understand and represent the input image. It can also increase the perceptual field of the convolutional layer because each convolutional kernel can capture the information in the input image at different scales. This helps to enhance the model's global understanding of the input image. It can increase the stability of the model because each convolutional kernel can capture different features in the image at different scales, thus making the model more stable. It can increase the flexibility

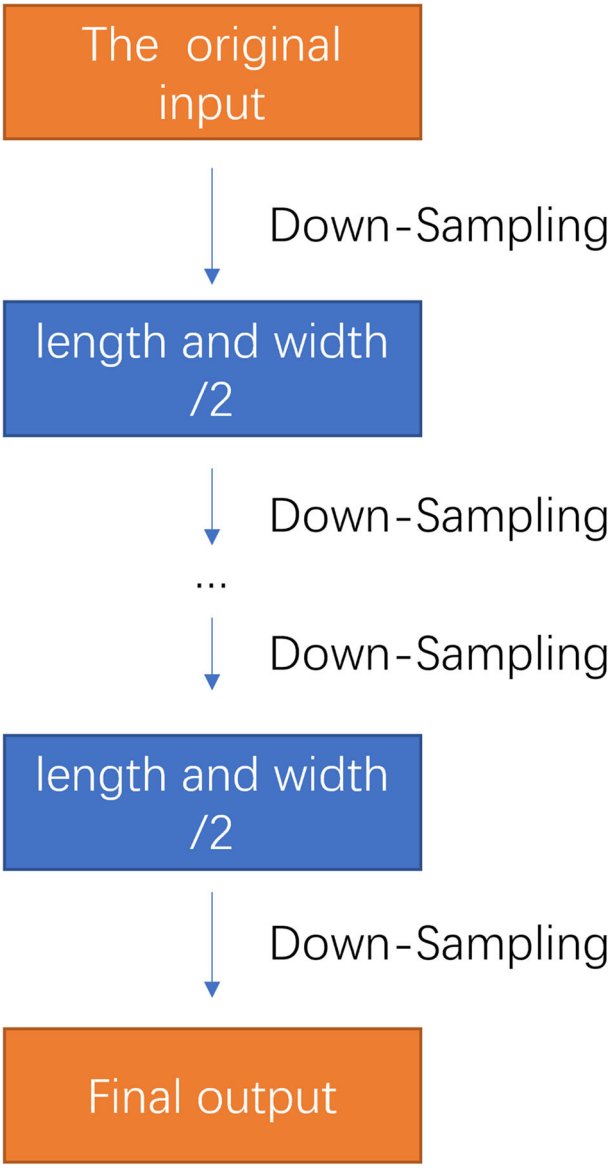


**Fig. 3** The Overall framework of the MDM method

of the model because the model can use convolutional kernels with different sizes to adapt to different input images. This helps to improve the generalization ability and adaptability of the model. In addition, we adopt a residual network in this model as it can be used to build a deeper neural network. It is well known that the deeper the neural network, the better fitting the function. The overall framework of the SR model is shown in Fig. 4.

As shown in Fig. 4, the proposed model has three different paths. And each path is divided into three different parts: the image pre-processing part, the upsampling part, and the

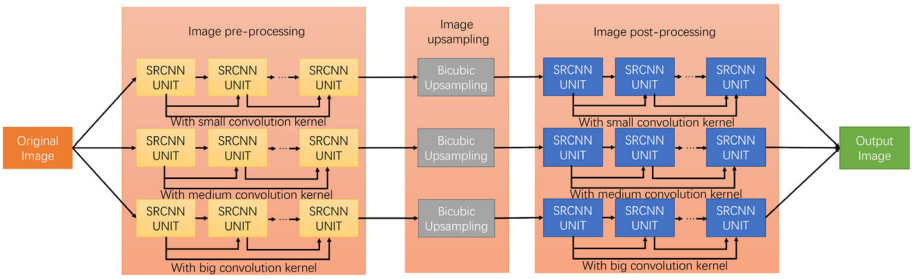


**Fig. 4** The overall framework of the SR model

post-processing part. The first (image pre-processing) part is a set of the SRCNN units, which is used to pre-process the image. The second part is the upsampling unit. In this research, the Bicubic method is adopted as the upsampling method. The third (post-processing) part is a set of SRCNN models too, which is used to make the model fit better.

From Fig. 4, each path has the same number of the SRCNN units. The number of the SRCNN unit depends on the size of the original image. We consider that a bigger image need deeper network to process, which is determined by the following equation,

$$N_{\max} \leq \frac{S_p}{20} + 1 \tag{1}$$

where $N_{max}$ represents the maximum number of SRCNN units of each part in each path, and $S_p$ represents the smaller value between the length and width of input image vectors. Because image sizes are often defined by hundreds to thousands of pixels, the appropriate number of SRCNN units is selected as 1/20 of the image size.

From Fig. 4, it can be seen that the difference between each path is that, convolution kernels with different sizes are used to process the image. Finally, the results of different paths are added on and used as the input of a final SRCNN unit to generate the final output image.

Different paths correspond to the convolution of the image with different degrees. Especially, according to Ref. [24], it is proved that the big convolution kernel achieves very good performance. In this research, the first path adopts the small-size convolution kernel, the second path adopts the medium-size convolution kernel, and the third path adopts the big-size convolution kernel. By using convolution kernels with different sizes the proposed SR model can extract features of images in three different sensory fields: small, medium, and large.

Besides, the size of the convolution kernels of each path depends on the size of the input image. The relationship between the size of the input image and the maximum size of the convolution kernel is shown in (2),

$$S_{k-\max} \leq \frac{S_p}{4} + 1 \tag{2}$$

where $S_{k-\max}$ represents the maximum size of the convolution kernel, and $S_p$ represents the smaller value between the length and width of input image vectors. According to (2), the size of the convolution kernel cannot be larger than a quarter of the size of the input vector, to achieve the best performance. Due to the size of the input image being 3*100*100, the size of the big convolution kernel is selected as 25*25. Accordingly, the sizes of the medium convolution kernel and the small convolution kernel are selected as 13*13 and 3*3, respectively.

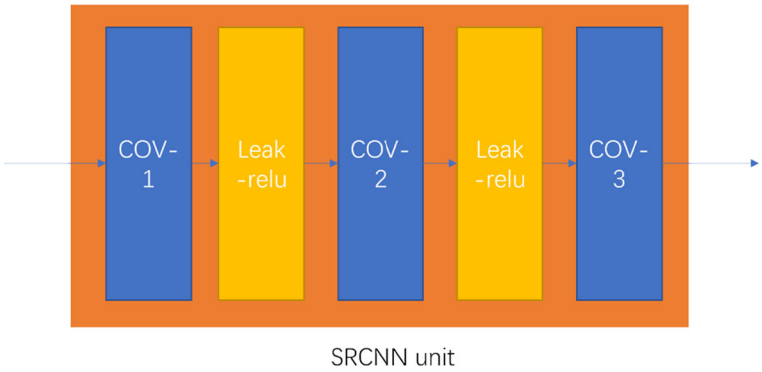


**Fig. 5** Structure of the SRCNN unit

The structure of the SRCNN unit used for image pre-processing and image post-processing is shown in Fig. 5. The SRCNN is proposed by Dong et al. [13]. In this research, our SRCNN model has 5 layers: 3 convolution layers and 2 activation function layers. The first convolution layer is used for patch extraction, which means extracting an image patch and then convolving it to get the features. The second convolution layer is used for nonlinear-linear mapping, which means mapping low-resolution features to high-resolution features. The third convolution layer is used for image reconstruction based on high-resolution features.

The upsampling method applied in this research is Bicubic interpolation [8]. Bicubic interpolation is the most commonly used interpolation method in two-dimensional (2D) space, such as 2D images. In this method, the value of the function $f$ at the point $(x, y)$ can be obtained by a weighted average of the sixteen nearest sampled points in the rectangular grid, where two polynomial interpolation cubic functions are used, one in each direction.

To avoid gradient disappearance or gradient explosion, we adopt the residual network. The residual unit can be implemented as a skip-layer connection for the SR model to get rid of the problem of gradient disappearance and gradient explosion and enable the model to fit better. Although the residual network may consume more time during training, however, in order to get a more accurate SR model, it is worth doing it.

Finally, we add up the output of each path to get the output of the SR model, which means our SR model can achieve different features by using kernels of different sizes. Then, another SRCNN unit is utilized in the final stage.

### 3.3 Loss function

We employ two distinct loss functions in our model: the content loss and the perceived loss. Each of these loss functions serves a specific purpose in improving the quality of super-resolved images.

The first loss function, the content loss, is based on the $L1$ loss, a well-established choice for quantifying content differences between images. It is expressed as:

$$L_1 - |f(x) - Y|$$

Here, $L_1$ represents the $L1$ loss, $f(x)$ is the output value of our model, and $Y$ represents the corresponding real value. The $L1$ loss focuses on minimizing the absolute differences

between the predicted and real values, which encourages the model to generate results that closely match the ground truth.

In contrast, the perceived loss leverages a pre-trained VGG16 network to extract features from both the original high-resolution (HR) image and the output HR image. Specifically, we utilize the activations of the third layer of the VGG16 network, which reflects low-level features of the images. The perceived loss is defined as:

$$L_p = \frac{1}{C_j H_j W_j} \left\| \phi_j(\hat{y}) - \phi_j(y) \right\|_2^2$$

Here, $L_p$ represents the perceived loss, where $C$, $H$, and $W$ are the channel, height, and width of the image, respectively. The subscript $j$ denotes the layer $j$ of the VGG16 network, $\phi_j(\hat{y})$ is the output value of our model, and $\phi_j(y)$ represents the real value. The perceived loss computes the mean squared differences between the feature representations of the predicted and real images at the third layer of VGG16.

The combination of these two loss functions results in our final loss function:

$$\text{Loss } - L_1 + L_p$$

The rationale behind this combination is as follows: The $L1$ loss ensures that the predicted image closely matches the real image in terms of pixel-wise content, while the perceived loss encourages the model to capture and reproduce the low-level features and details that are essential for high perceptual quality. By incorporating both content and perceptual aspects into the loss function, our model is incentivized to produce images that not only exhibit fidelity to the original content but also possess enhanced visual quality.

This combination of loss functions has been found effective in generating SR images that excel both in terms of quantitative metrics and perceived image quality.

The flow of the entire algorithm is shown in Algorithm 1.

Among them, the specific details of each path can be adjusted according to the specific needs and experimental results, such as the size and number of convolution kernels, the number of SRCNN units and Bicubic amplifiers, etc. In the final training, the total loss function can be minimized using appropriate optimization algorithms and hyperparameters to improve the performance and generalization of the model.

## 4 Experiments

### 4.1 Evaluation metrics

To evaluate the performance of our SR model, in this research two evaluation metrics are employed: the Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM). They are commonly used as evaluation metrics in SR.

### 4.2 Validation methods

In this research, we use two validation methods to evaluate the proposed SR model: hand-out validation and 10-fold cross-validation.

**Algorithm 1** A novel super-resolution algorithm using multi-path big-size convolution kernel and residual network.

Initialize training dataset $D_{train}$ and testing dataset $D_{test}$
Set the upscaling factor $s$, number of channels $C$, and number of residual blocks $R$
Define the input and output size for the network
Define the network architecture for each pathway
**for** each pathway $p \in [1, 3]$ **do**
  Initialize pathway-specific weights $w^p$
  Define pathway-specific input tensor $X^p$
  Define pathway-specific output tensor $Y^p$
  Define the convolution kernel size for pathway $p$: $S_k^p$
  Define the number of SRCNN units in the first and third sections: $N_1^p$ and $N_3^p$
  Define the number of bicubic upscaling units in the second section: $N_2^p$
  **for** each SRCNN unit $u \in [1, N_1^p]$ in the first section **do**
    Define pathway-specific convolution layers $conv_u^p$ with kernel size $S_k^p$ and activation function $ReLU_u^p$
    Apply the residual operation to $X^p$: $X^p \leftarrow X^p + ReLU_u^p(conv_u^p(X^p))$
  **end for**
  **for** each bicubic upscaling unit $u \in [1, N_2^p]$ in the second section **do**
    Define pathway-specific bicubic upscaling layers $upscale_u^p$
    Apply the bicubic upscaling operation to $X^p$: $X^p \leftarrow upscale_u^p(X^p)$
  **end for**
  **for** each SRCNN unit $u \in [1, N_3^p]$ in the third section **do**
    Define pathway-specific convolution layers $conv_u^p$ with kernel size $S_k^p$ and activation function $ReLU_u^p$
    Apply the residual operation to $X^p$: $X^p \leftarrow X^p + ReLU_u^p(conv_u^p(X^p))$
  **end for**
  Define pathway-specific upscaling layer $upscale^p$ and apply it to $X^p$ to obtain $Y^p$: $Y^p \leftarrow upscale^p(X^p)$
**end for**
Add up the outputs from the three pathways to obtain the final output: $Y = Y^1 + Y^2 + Y^3$
Apply another SRCNN unit to $Y$ to obtain the final output: $Y \leftarrow ReLU(conv(Y))$
Define the perceptual loss function $L_{pe}$
Define the total loss function as a combination of perceptual loss and a suitable content loss: $L_1 + L_p$
Evaluate the performance of the network on $D_{test}$

#### 4.2.1 Hand-out validation

The hand-out validation method statically divides the dataset into a training set, a validation set, and a test set according to a fixed ratio. The validation dataset can help us check the state of the model during training: and whether convergence has been achieved. Totally, the experiment has been conducted 10 times to evaluate the SR model's performance. The best hyperparameters are used and the model is retrained using all training datasets to obtain the final model.

#### 4.2.2 10-Fold cross validation

The second validation method of our research is the 10-fold cross-validation method. The 10-fold cross-validation method is a dynamic validation method that reduces the impact of data segmentation. We divide the training set into 10 divisions and use 1 of the 10 divisions at a time as the validation set and the rest as the training set. After 10 training sessions, we have 10 different models. Then we evaluate the effectiveness of the 10 models and select the

**Table 1** The experimental platform

| Name | Type |
|---|---|
| CPU | Intel i9 10900 |
| GPU | Nvidia RTX3060 |
| RAM | 64GB |

best hyperparameters from them. Then the best hyperparameters are used and the model is retrained using all 10 of the data as the training set to obtain the final model.

### 4.3 Experimental platform and parameter setting

The configuration parameters of the experimental platform are shown in Table 1. The values of hyperparameters of the model are listed in Table 2.

## 5 Results

### 5.1 Experiment result for building the dataset

In this study, we employed the proposed Multilayer Downsampling Method (MDM) to downsample images for constructing the training dataset for the super-resolution (SR) model. To assess the efficacy and performance of our method, we compared it against several established downsampling methods:

1) Bicubic without MDM;
2) Bicubic with MDM;
3) K-nearest without MDM;
4) K-nearest with MDM;
5) Bilinear without MDM; and
6) Bilinear with MDM.

Additionally, various weights were applied to integrate each method with the proposed MDM to determine the optimal downsampling approach for dataset construction.

**Results** The comparative results are illustrated in Figs. 6 and 7. It's evident from the analysis that downsampling methods incorporating MDM outperform those without it. Specifically, methods with MDM exhibited smoother edges and overall better image quality. Among the three downsampling strategies, Bicubic with MDM demonstrated the most visually appealing results, whereas the other methods tended to produce mosaic-like artifacts noticeable to the naked eye. Consequently, Bicubic with MDM has been selected as our primary method for building the training dataset.

**Table 2** The hyperparameters of the model

| Parameters | Value |
|---|---|
| Learning rate | 0.0001 |
| Batch size | 8 |
| Epochs | 10000 |
| Optimizer | ADAM |

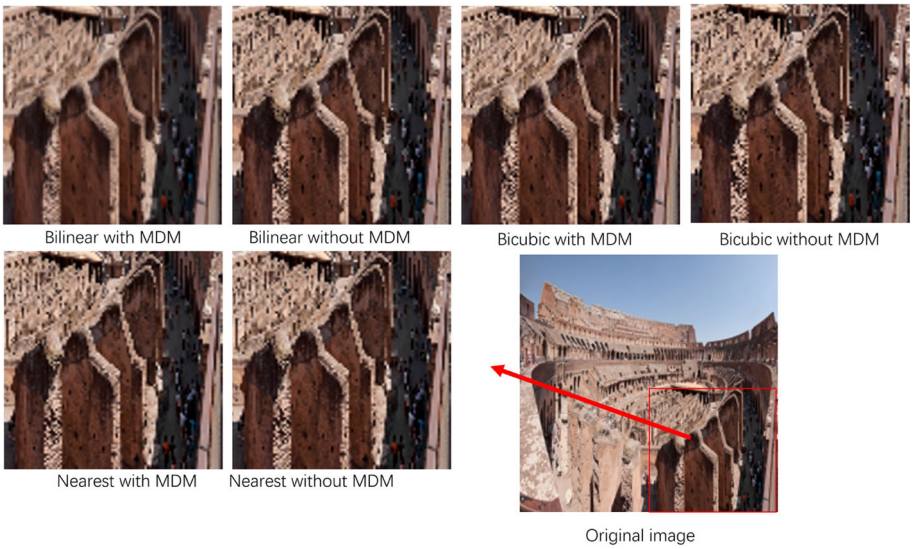


**Fig. 6** Performance of different downsampling methods on the Colosseum image

**Discussion** The superior performance of the Bicubic with MDM method can be attributed to its more sophisticated approach in preserving image details during downsampling. Unlike traditional methods that often lead to significant information loss, MDM ensures a more gradual and detailed reduction process. This is particularly beneficial for SR tasks where the preservation of intricate details is crucial for the accurate reconstruction of HR images. Furthermore, the adaptability of MDM to integrate with various downsampling techniques highlights its versatility and potential for further optimization and application in diverse SR

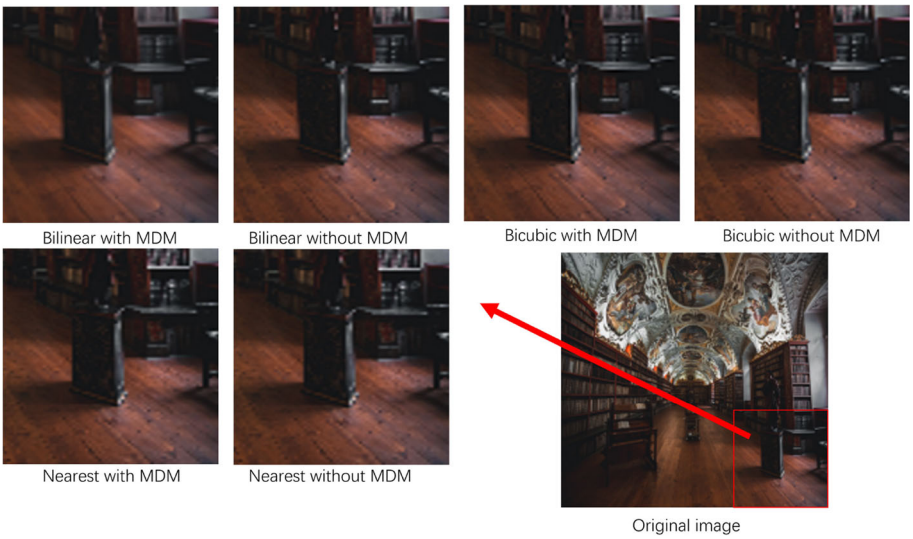


**Fig. 7** Performance of different downsampling methods on the Living room image

**Table 3** The experimental results of loss functions

| | Propoosed loss function | | L1 loss function | |
|---|---|---|---|---|
| | PSNR | SSIM | PSNR | SSIM |
| set5 | 30.32 | 0.940 | 29.89 | 0.921 |
| set14 | 26.99 | 0.892 | 26.32 | 0.872 |
| Urban100 | 23.99 | 0.827 | 23.48 | 0.803 |

scenarios. Moving forward, exploring the intricate balance between downsampling efficiency and detail preservation will be crucial in enhancing the overall performance and applicability of SR models.

### 5.2 Experiment result of loss function

We conducted experiments to demonstrate the effectiveness of our proposed loss function. The same model was tested using our loss function and L1 loss function with the same parameters and training set. The experimental results are presented in figures, which clearly show the superior performance of our loss function. Furthermore, we evaluated the proposed method using three different datasets and compared it with existing SR algorithms. Table 3 presents the experimental results, while sample images are displayed in Figs. 8 and 9. Our proposed method achieved significant improvements in both quantitative and qualitative measures, indicating its effectiveness in enhancing image super-resolution performance.

**Results** The experimental results presented in Table 3 demonstrate that our proposed loss function outperforms the traditional L1 loss function in terms of both SSIM and PSNR. The superiority of our loss function is further confirmed by the sample images shown in Figs. 8 and 9. These images clearly reveal that our proposed loss function generates more detailed and visually rich images compared to the traditional L1 loss. The combination of our proposed model and loss function results in significant improvements in image super-resolution performance, highlighting the importance of designing effective loss functions to achieve superior results.

**Discussion** The observed improvements can be attributed to the adaptive nature of the proposed loss function, which potentially captures a more comprehensive range of image features compared to the L1 loss. Our function might be better attuned to the perceptual differences that are more aligned with human visual interpretation, thereby generating images that are not only high in pixel-accuracy but also in perceived quality.

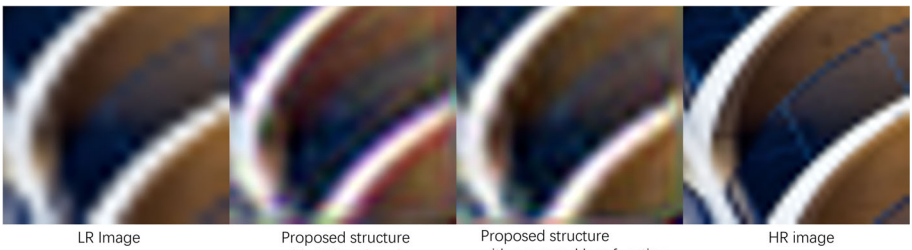


**Fig. 8** The sample 1 of the loss function

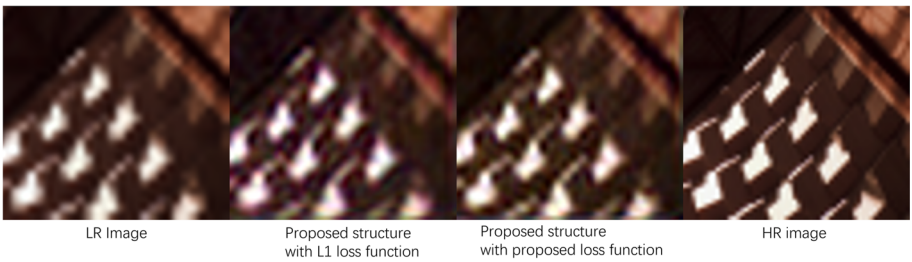


**Fig. 9** The sample 2 of the loss function

### 5.3 Experiment result of SR algorithm

In our study, we utilized two distinct validation methods: a 90%-10% hand-out validation and a 10-fold cross-validation. These methods were employed to validate the robustness and effectiveness of our SR algorithm under different conditions.

To test the transferability of our model across varied scenes - a crucial aspect for real-world applicability - we employed three different datasets: Set5 [26], Set14 [26], and Urban100 [27]. Transferability is vital as it ensures the trained SR model can be effectively applied to a diverse range of image types encountered in real-world scenarios.

For comparative analysis, we also trained several established SR algorithms using the Div2k [25] dataset, which is the same dataset used for our model. These algorithms include Bicubic [8], SRCNN [13], SRResnet [15], ESPCN [14], and Real-ESRGAN [32], serving as benchmarks against which we could measure the performance of our proposed model.

**Results** Our results, illustrated in Figs. 10, 11, and 12 and detailed in Table 4, demonstrate that our model consistently achieves superior PSNR and SSIM scores compared to the other six SR methods across all tested datasets. For instance, in Fig. 10, it's evident that our model retains more detailed features of the mountain behind the penguin, and the edge of the penguin's head appears smoother and more defined. Similarly, Figs. 11 and 12 showcase the model's ability to enhance clarity and detail, surpassing other methods. Notably, while Real-ESRGAN shows a good visual effect, it tends to oversimplify and remove details, whereas our method maintains a better balance between enhancing visual effect and retaining image details. The comparative results strongly validate the effectiveness of our SR model.

**Discussion** The superior performance of our SR algorithm can be attributed to several factors. Firstly, the innovative architecture and loss function designed specifically for this model


**Fig. 10** The results of the Penguin image


**Fig. 11** The result of the Wolve image

allows for more nuanced feature extraction and reconstruction, leading to higher-quality image outputs. Secondly, the model's adaptability to various datasets, as shown in our experiments, indicates its robustness and potential for broader application.

One interesting observation is the consistently better performance of 10-fold cross-validation over hand-out validation. This suggests that our model benefits from a more extensive and varied training regime, which 10-fold cross-validation provides by iterative training on different data subsets. This finding indicates potential areas for further optimization, such as experimenting with even more diverse and extensive training datasets.

However, it's also crucial to note some limitations. While our model generally outperforms others, there are instances where the improvements in SSIM are marginal. This highlights the challenge of achieving perceptual quality improvements that align perfectly with quantitative metrics. Future work could focus on exploring more sophisticated perceptual quality assessment methods or integrating additional features into the loss function to better capture human visual perception nuances.

## 6 Conclusion

In this study, we have tackled two problems in SR technology: the undervalued role of training dataset quality and the inconsistency between index scores and human visual perception. Our approach, which includes the introduction of a multilayer dimensionality reduction method (MDM) and an adaptive multi-path structure model with large convolutional kernels, has been shown to enhance the performance and visual quality of SR images. Furthermore, the loss function designed to improve perceptual quality has demonstrated promising results, as

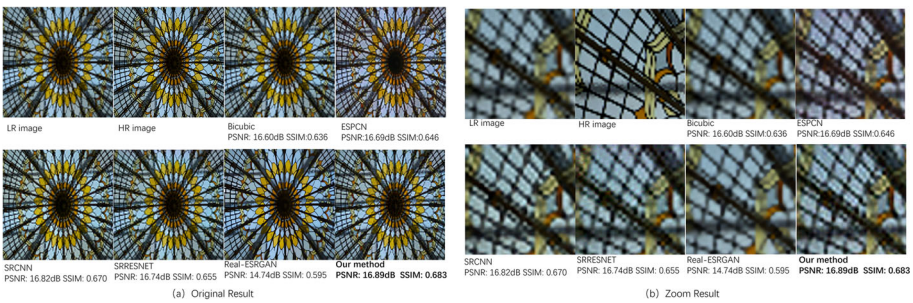


**Fig. 12** The result of the Glass image

**Table 4** The experimental results of our model and several typical methods

| | Our model | SRCNN | Bicubic | SRResnet | ESPCN | Real-ESRGAN |
|---|---|---|---|---|---|---|
| PSNR Scores for 10 Fold Cross-Validation | | | | | | |
| Set5 [26] | **31.09** | 29.41 | 30.55 | 29.86 | 26.08 | 30.83 |
| Set14 [26] | **25.90** | 25.35 | 25.71 | 25.52 | 23.52 | 25.81 |
| Urban100 [27] | **20.85** | 20.63 | 20.52 | 20.62 | 20.18 | 20.83 |
| Div2k[27] | **23.61** | 23.26 | 23.34 | 23.34 | 22.55 | 23.57 |
| PSNR Scores for Hand-out Cross-Validation | | | | | | |
| Set5 | **31.06** | 29.3 | 30.33 | 29.80 | 25.89 | 30.73 |
| Set14 | **25.79** | 25.15 | 25.62 | 25.45 | 23.33 | 25.47 |
| Urban100 | **20.83** | 20.55 | 20.51 | 20.59 | 20.16 | 20.81 |
| Div2k | **23.60** | 22.96 | 23.32 | 23.18 | 22.54 | 23.55 |
| SSIM Scores for 10 Fold Cross-Validation | | | | | | |
| Set5 [26] | **0.938** | 0.936 | **0.938** | 0.922 | 0.909 | 0.932 |
| Set14 [26] | **0.862** | **0.862** | 0.858 | 0.853 | 0.844 | 0.859 |
| Urban100 [27] | **0.750** | 0.740 | 0.734 | 0.742 | 0.734 | 0.744 |
| Div2k [25] | **0.809** | 0.807 | 0.801 | 0.803 | 0.790 | 0.804 |
| SSIM Scores for Hand-out Cross-Validation | | | | | | |
| Set5 | **0.932** | 0.927 | 0.927 | 0.912 | 0.901 | 0.930 |
| Set14 | **0.858** | 0.857 | 0.845 | 0.851 | 0.839 | 0.838 |
| Urban100 | **0.741** | 0.734 | 0.732 | 0.739 | 0.724 | 0.734 |
| Div2k | **0.807** | 0.792 | 0.801 | 0.801 | 0.787 | 0.801 |

The bold emphasis our method has best performance

evidenced by superior SSIM and PSNR values and more visually appealing images. Looking ahead, we plan to explore replacing the traditional loss function with Generative Adversarial Networks (GANs), inspired by literature suggesting that GANs can further smooth and refine the images [15, 25]. We also aim to experiment with different upsampling methods beyond bicubic interpolation to assess potential performance improvements. In summary, our research contributes an effective approach to the SR domain.

**Acknowledgements** The authors would like to express their thanks for the financial support from the Research Committee, Department of Industrial and Systems Engineering, The Research Institute of Advanced Manufacturing of the Hong Kong Polytechnic University, Hong Kong Special Administrative Region (Project code: UAMU, W22B, CD9F) and Fujian Province Education and Research Fund for Young and Middle-Aged Teachers (Science and Technology).

**Funding** Open access funding provided by The Hong Kong Polytechnic University.

**Availability of data and materials** The datasets generated during and/or analysed during the current study are available from the corresponding author on reasonable request.

## Declarations

**Conflicts of interest** The authors declare that they have no conflict of interest

## References

1. Girshick R, Donahue J, Darrell T, Malik J (2016) Region-based convolutional networks for accurate object detection and segmentation. IEEE Trans Pattern Anal Mach Intell 38(1):142–158
2. Bai Y, Zhang Y, Ding M, Ghanem B (2018) SOD-MTGAN: Small object detection via multi-task generative Adversarial Network. In: Computer Vision - ECCV 2018 pp 210-226
3. Mudunuri SP, Biswas S (2016) Low resolution face recognition across variations in pose and illumination. IEEE Trans Pattern Anal Mach Intell 38(5):1034–1040
4. Greenspan H (2008) Super-resolution in medical imaging. Comput J 52(1):43–63
5. Lillesand T, Kiefer RW, Chipman J (2014) Remote Sensing and Image Interpretation. John Wiley and Sons
6. Lucy LB (1992) Resolution limits for deconvolved images. Astron J 104:1260
7. Swaminathan A, Wu M, Liu KJR (2008) Digital image forensics via intrinsic fingerprints. IEEE Trans Inf Forensics Secur 3(1):101–117
8. Carlson RE, Fritsch FN (1985) Monotone piecewise bicubic interpolation. SIAM J Numer Anal 22(2):386–400
9. Nikazad T, Davidi R, Herman GT (2012) Accelerated perturbation-resilient block-iterative projection methods with application to image reconstruction. Inverse Prob 28(3):035005
10. Gubin LG, Polyak BT, Raik EV (1967) The method of projections for finding the common point of convex sets. USSR Comput Math Math Phys 7(6):1–24
11. Levitan E, Herman GT (1987) A maximum a posteriori probability expectation maximization algorithm for image reconstruction in Emission Tomography. IEEE Trans Med Imaging 6(3):185–192
12. Hinton GE, Osindero S, Teh Y-W (2006) A fast learning algorithm for deep belief nets. Neural Comput 18(7):1527–1554
13. Dong C, Loy CC, He K, Tang X (2014) Learning a deep convolutional network for Image Super-resolution. In: Computer Vision - ECCV 2014 pp 184-199
14. Shi W, Caballero J, Huszar F, Totz J, Aitken AP, Bishop R, Rueckert D, Wang Z (2016) Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In 2016 IEEE conference on computer vision and pattern recognition (CVPR)
15. Ledig C, Theis L, Huszar F, Caballero J, Cunningham A, Acosta A, Aitken A, Tejani A, Totz J, Wang Z, Shi W (2017) Photo-realistic single image super-resolution using a generative adversarial network. In 2017 IEEE conference on computer vision and pattern recognition (CVPR)
16. Ghifary M, Kleijn WB, Zhang M, Balduzzi D, Li W (2016) Deep reconstruction-classification networks for unsupervised domain adaptation. In: Computer Vision - ECCV 2016 pp 597-613
17. Dong C, Loy CC, Tang X (2016) Accelerating the super-resolution convolutional neural network. In Computer Vision - ECCV 2016 pp 391-407
18. Lai W-S, Huang J-B, Ahuja N, Yang M-H (2017) Deep laplacian pyramid networks for fast and accurate super-resolution. In 2017 IEEE conference on computer vision and pattern recognition (CVPR)
19. Lan R et al (2021) Cascading and Enhanced Residual Networks for Accurate Single-Image Super-Resolution. IEEE Trans Cybern 51(1):115–125
20. Li Z et al (2021) DeepVolume: Brain Structure and Spatial Connection-Aware Network for Brain MRI Super-Resolution. IEEE Trans Cybern 51(7):3441–3454
21. Jiang J et al (2020) Ensemble Super-Resolution With a Reference Dataset. IEEE Trans Cybern 50(11):4694–4708
22. Tai Y, Yang J, Liu X (2017) Image super-resolution via deep recursive residual network. In 2017 IEEE conference on computer vision and pattern recognition (CVPR)
23. Lim B, Son S, Kim H, Nah S, Lee KM (2017) Enhanced deep residual networks for single image Super-Resolution. In 2017 IEEE conference on computer vision and pattern recognition workshops (CVPRW)

24. Xiaohan D, Zhang X, Han J, Ding G (2022) Scaling up your kernels to 31x31: Revisiting large kernel design in cnns. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition pp. 11963-11975
25. Agustsson E, Timofte R (2017) NTIRE 2017 Challenge on Single Image Super-resolution: Dataset and study. In 2017 IEEE conference on computer vision and pattern recognition workshops (CVPRW)
26. Bevilacqua M, Roumy A, Guillemot C, Morel M-A (2012) Low-complexity single-image super-resolution based on nonnegative neighbor embedding. In Procedings of the british machine vision conference 2012
27. Huang J-B, Singh A, Ahuja N (2015) Single Image Super-resolution from transformed self-exemplars. In 2015 IEEE conference on computer vision and pattern recognition (CVPR)
28. Wang X, Xie L, Dong C, Shan Y (2021) Real-ESRGAN: Training real-world blind super-resolution with pure synthetic data. In 2021 IEEE/CVF international conference on computer vision workshops (ICCVW)
29. Zhang Y, Tian Y, Kong Y, Zhong B, Fu Y (2018) Residual dense network for image super-resolution. In 2018 IEEE/CVF conference on computer vision and pattern recognition (CVPR)
30. Sajjadi MS, Scholkopf B, Hirsch M (2017) EnhanceNet: Single image super-resolution through automated texture synthesis. In 2017 IEEE international conference on computer vision (ICCV)
31. Wang X, Yu K, Wu S, Gu J, Liu Y, Dong C, Qiao Y, Loy CC (2019) Esrgan: Enhanced super-resolution generative adversarial networks. In Lecture notes in computer science pp 63-79
32. Chitwan S, Ho J, Chan W, Salimans T, Fleet DJ, Norouzi M (2021) Image super-resolution via iterative refinement. arXiv:2104.07636
33. Hore A, Ziou D (2010) Image quality metrics: PSNR vs. SSIM. In 2010 20th international conference on pattern recognition
34. Karen S, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556
35. Prashanth HS, Shashidhara HL, Murthy KNB (2009) Image scaling comparison using Universal Image Quality index. In 2009 International conference on advances in computing, control, and telecommunication technologies
36. Zhou Y, Du X, Wang M, Huo S, Zhang Y, Kung S-Y (2022) Cross-Scale Residual Network: A General Framework for Image Super-Resolution, Denoising, and Deblocking. IEEE Trans Cybern 52(7):5855–5867
37. Lan R, Sun L, Liu Z, Lu H, Pang C, Luo X (2021) MADNet: A Fast and Lightweight Network for Single-Image Super Resolution. IEEE Trans Cybern 51(3):1443–1453
38. Li Z, Yang J, Liu Z, Yang X, Jeon G, Wu W (2019) Feedback Network for Image Super-Resolution. In 2019 IEEE/CVF conference on computer vision and pattern recognition (CVPR)
39. Bhat G, Danelljan M, Van Gool L, Timofte R (2021) Deep Burst Super-Resolution. In 2021 IEEE/CVF conference on computer vision and pattern recognition (CVPR)
40. Yang F, Yang H, Fu J, Lu H, Guo B (2020) Learning Texture Transformer Network for Image Super-Resolution. In 2020 IEEE/CVF conference on computer vision and pattern recognition (CVPR)
41. Zhang K, Van Gool L, Timofte R (2020) Deep Unfolding Network for Image Super-Resolution. In 2020 IEEE/CVF conference on computer vision and pattern recognition (CVPR)

## Authors and Affiliations

**Haotian Zhang[1] · Long Teng[1] · Youyi Wang[2] · Hang Qu[3] · Chak-yin Tang[1]**

Long Teng
eric-long.teng@polyu.edu.hk

Youyi Wang
eyywang@ntu.edu.sg

Hang Qu
hangqu@foxmail.com

Chak-yin Tang
cy.tang@polyu.edu.hk

1 The Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong, China

2 The School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, Singapore

3 Affiliated Hospital of Yangzhou University, Yangzhou University, Yangzhou, China