Journal of Combinatorial Mathematics
and Combinatorial Computing
www.combinatorialpress.com/jcmcc

Combinatorial Press

Multiscale processing of image gradient domain-based convolutional neural networks for visual feature enhancement

Lanyue Pi[1,✉], Yangzi Mu[1], Lanyu Pi[2]

[1] *Zhengzhou Vocational College of Finance and Taxation, Zhengzhou, Henan, 450048, China*

[2] *China International Telecommunication Corporation HE NAN Communication Service Co., Ltd., Zhengzhou, Henan, 450016, China*

ABSTRACT

In recent years, with the development of science and technology, image enhancement has become a very important topic in scientific research, become an indispensable part of machine vision, and has a wide range of applications in various fields of computer vision. In this paper, the image gradient enhancement algorithm is first improved based on the image gradient field, and its enhancement effect on low quality (low resolution) images is found to be poor through experiments. For this reason, the study constructs a multi-scale feature image enhancement model (LIEN-MFC) by convolutional neural network to further optimize the image enhancement effect. By comparing with different algorithms, the average PSNR of the model is 21.80 and the average SSIM is 0.8767, and it outperforms other compared algorithms in both PSNR and SSIM. In addition, the ablation experiments demonstrate that the enhancement effect of the LIEN-MFC model is further improved on the basis of the improved image gradient enhancement algorithm. The results show that the image enhancement model algorithm with multi-scale features proposed in this paper has a significant image enhancement effect and the improved image gradient enhancement in image enhancement of convolutional neural networks improves the model performance to some extent.

Keywords: Image gradient enhancement, convolutional neural network, multi-scale features, image enhancement

1. Introduction

Convolutional neural network (CNN) is a deep learning model that is widely used in the field of computer vision. CNNs have achieved very significant results in tasks such as image recognition, target

✉ Corresponding author.
E-mail address: pi_zzcs0901@163.com (L. Pi).
Received 15 January 2024; Revised 10 February 2024; Accepted 30 November 2024; Published Online 15 April 2025.
DOI: 10.61091/jcmcc127a-269

detection and semantic segmentation. However, traditional CNNs have certain limitations when dealing with multi-scale features [1-4]. Therefore, convolutional neural networks based on the image gradient domain have become one of the hotspots of research, especially for multi-scale processing methods in visual feature enhancement [5-7].

In the field of image processing, multi-scale feature fusion technique is a method to effectively integrate feature information at different scales. The multi-scale feature fusion technique can make full use of the information of different scales in the image, so as to improve the recognition and localization ability of the model to the target [8-11]. In convolutional neural networks (CNNs), multi-scale feature fusion technique is also of great importance. First of all, the problems of traditional CNN models in dealing with multi-scale features mainly include two aspects, one is the loss of feature information and the other is the incompleteness of scale information [12-15]. At a single scale, the CNN model may be limited in its ability to recognize and localize the target, because the size and shape of the target may exhibit different features at different scales. Therefore, a method is needed to effectively integrate feature information at different scales [16-19]. In CNN modeling, multi-scale feature fusion techniques mainly include the following approaches: pyramid network, multi-scale convolution, and feature pyramid [20-22].

For the research of visual feature enhancement in the field of computer vision, this paper focuses on the enhancement algorithms in the field of image gradient and improves the image gradient enhancement algorithm by comprehensively considering the visual features perceived by the human eye. In the experiment, it is found that the effect of image gradient enhancement algorithm alone is not ideal on low-quality images. So a multi-scale feature enhancement model is proposed based on convolutional neural network. Finally, the effect of the improved image gradient enhancement algorithm and LIEN-MFC model in this paper is proved by comparison with other algorithmic models and ablation experiments.

2. Enhancement algorithms based on image gradient fields

2.1. Image Enhancement Algorithm

Image enhancement technology is one of the key techniques of image processing technology, classical image enhancement algorithms have many shortcomings and shortcomings, so the classical image enhancement algorithms often can not better improve the quality of the image. With the in-depth study of image processing technology, many new algorithms have appeared in image enhancement technology.

The brightness of image pixels is different from each other and the difference in brightness can be reflected by the gradient of the image, while the difference in the overall brightness of the image is reflected by the gradient field of the image. In most cases, the contrast of an image can be reflected in terms of the gradient of the image, and a high contrast of an image means that the gradient of the image is strong and the details are clear, i.e., the difference in the brightness of the image pixels is large. Thus, the gradient field of an image can be lifted from the perspective of the gradient field of the image thereby realizing the contrast enhancement of the image.

The image gradient field is directly enhanced using the number of enhancement nets and the resultant image is obtained from the gradient field reconstruction. The wan method simultaneously takes into account the relationship between the image contrast and the image gradient size, to make the gradient field in the low-contrast region of the image to be enhanced, which can be realized by compressing the range of variation of the image gradient. From the enhanced gradient field can not be directly obtained by integrating the reconstruction results, this paper uses the principle of least squares to solve this problem.

In the case of a univariate real-valued function, the gradient is just the derivative. And for a linear function, the gradient is the slope of the line. In vector calculus, the gradient of a scalar field is a vector

field. The gradient at a point in the scalar field points in the direction of the fastest growing scalar field, and the value of the gradient is the rate of change of this maximum. An image is a scalar field of random variables.

In digital image processing, an image is usually viewed as a two-dimensional discrete function, so calculating the gradient of an image is the derivative of this two-dimensional discrete function. Unlike the case of continuous functions, in practical use it is approximated by the difference of the image pixels. Window templates are usually utilized to perform the convolution operation. Commonly used template operators for approximating the gradient of discrete digital images are Laplacian operator, Robert operator, Prewitt operator, and Sobel operator. Usually in image processing, one of the most commonly used Laplacian operators is denoted as:

$$\Delta I = I(x+1,y)+I(x-1,y)+I(x,y+1)+I(x,y-1)-4I(x,y) \tag{1}$$

The simplest forward difference is taken here to approximate the gradient of image $I(x,y)$:

$$\nabla I(x,y) \approx \left(I(x+1,y)-I(x,y) \quad I(x,y+1)-I(x,y)\right) \tag{2}$$

The purpose of gradient field processing of an image is to enhance the overall pixel-to-pixel difference of the image so as to achieve the effect that the contrast of the image after gradient field reconstruction is improved. To achieve the desired enhancement effect, constructing an appropriate enhancement function plays a very important role.

The constructed enhancement function should have a large enhancement effect on very small gradient values, and a relatively small enhancement effect on originally large gradient values, and should not make the enhanced gradient value larger than 255. function $\phi(x,y)$ meets the above requirements:

$$\phi(x,y) = \frac{1}{(\beta+10^{-6})^{\alpha}} \tag{3}$$

Among them:

$$\beta = \frac{\| \nabla I(x,y) \|}{255} \tag{4}$$

The reason for adding 10^{-6} after β is to prevent the case where β is zero, which makes the calculation meaningless. And α is an adjustable parameter, which can be adjusted to α for different images to get a better enhancement effect. The value of $\alpha > 0$ is from 0.3 to 0.45. Analyzing this constructor, we can find that when the mode $\| \nabla I(x,y) \|$ of the gradient obtains the theoretical maximum value of 255, the value of the enhancement function is equal to 1, which maintains the original value of the gradient, and then the local contrast in the original image is also high enough, and there is no need for enhancement. When the mode $\| \nabla I(x,y) \|$ of the gradient is smaller than the theoretical maximum value of 255, the enhancement function is greater than 1, and the gradient value is enhanced. And $\| \nabla I(x,y) \|$ the smaller the enhancement effect is more obvious.

Then the gradient image after the enhancement function $\phi(x,y)$ enhancement is:

$$G(x,y) = \nabla I(x,y)\phi(x,y) \tag{5}$$

After obtaining the enhanced gradient image, the output image needs to be reconstructed from the gradient image. Unlike the case of one-dimensional signals, for the case of two-dimensional images, the output image $\tilde{I}$ after gradient enhancement cannot be obtained by directly integrating over $G(x,y)$. There generally does not exist an image $\tilde{I}$ that satisfies $\nabla^2 \tilde{I} = G$. For a potential function, such as a two-dimensional discrete image signal, its gradient has to be a conserved field, i.e., the image gradient $V\tilde{I}$ has to be satisfied:

$$\frac{\partial^2 \tilde{I}}{\partial x \partial y} = \frac{\partial^2 \tilde{I}}{\partial y \partial x} \tag{6}$$

A gradient approximation to $G(x, y)$ is found using a closest solution from all possible luminance images, i.e., by least squares, for a luminance image $\tilde{I}(x, y)$. Thus the output image $\tilde{I}(x, y)$ should make the integral:

$$\iint F(\nabla \tilde{I}, G) dx dy \tag{7}$$

Obtain the minimum value:

$$F(\nabla \tilde{I}, G) = \| \nabla \tilde{I} - G \|^2 = \left(\frac{\partial \tilde{I}}{\partial x} - G_x \right)^2 + \left(\frac{\partial \tilde{I}}{\partial y} - G_y \right)^2 \tag{8}$$

By the variational principle, the $I(x, y)$ that minimizes the above integral will satisfy the Euler I Lagrange equation:

$$\frac{\partial F}{\partial \tilde{I}} - \frac{d}{dx} \frac{\partial F}{\partial \tilde{I}_x} - \frac{d}{dy} \frac{\partial F}{\partial \tilde{I}_y} = 0 \tag{9}$$

This is a partial differential equation for I, which is obtained by bringing F into the above equation:

$$\frac{\partial^2 \tilde{I}}{\partial x^2} + \frac{\partial^2 \tilde{I}}{\partial y^2} = \frac{\partial G_x}{\partial x} + \frac{\partial G_y}{\partial y} \tag{10}$$

The above equation is the well-known Poisson equation:

$$\nabla^2 \tilde{I} = div(G) \tag{11}$$

By solving this equation, we can obtain the contrast-enhanced luminance image. The pixel brightness and gradient at the image boundaries are 0. Where:

$$\nabla^2 \tilde{I}(x, y) \approx \tilde{I}(x+1, y) + \tilde{I}(x-1, y) + \tilde{I}(x, y+1) + \tilde{I}(x, y-1) - 4\tilde{I}(x, y) \tag{12}$$

The scatter of the enhanced gradient field is approximated by a backward first-order difference pass:

$$div(G) \approx G_x(x, y) - G_x(x-1, y) + G_y(x, y) - G_y(x, y-1) \tag{13}$$

So far there are many methods to solve Poisson's equation, and in this paper, a fast algorithm of discrete sinusoidal transform is used to solve it. The algorithm is characterized by combining the global and local information of the image, which solves the disadvantage that the histogram equalization algorithm, SSR algorithm and MSR algorithm fail to enhance the local information of the image. Compared with the enhancement effect of histogram equalization algorithm, SSR algorithm and MSR algorithm, the advantage of this algorithm is that it does not produce halo phenomenon. However, the algorithm also has some disadvantages, firstly, the computational complexity and the solution of Poisson's equation make the algorithm complexity relatively high. Secondly, the construction of the enhancement function has a large impact on the processing effect of the algorithm, and the same gradient enhancement function is adopted for all images, so the enhancement effect is affected by the enhancement function, and it is not able to produce robust enhancement effect for all kinds of images.

2.2. Improved image gradient enhancement algorithm

Since the perceptual ability of the human eye varies with the background brightness, the gradient field based enhancement algorithm focuses on the domain information of the pixels of the image, and does not take into account the average brightness of the image, which is a gray level statistical information.

The processing effect of this algorithm is closely related to the construction of the enhancement function, and the poor enhancement effect for high brightness regions leads to the loss of details, which is attributed to the excessive suppression of the gradient in these regions. Due to the existence of the critical visible deviation of the human eye, the background brightness needs to be considered while enhancing the gradient of the image.

The gradient field based enhancement algorithm takes into account the grayscale information within each specific pixel field and makes full use of the spatial information of the image, regions with small gradient modes correspond to regions in the original image with close gray levels and low contrast. Regions with large gradient modes correspond to parts of the original image with large gray level differences and high contrast. Based on this feature, we improve the algorithm by combining the background brightness and the gradient field-based enhancement algorithm.

This paper proposes the concept of visually perceived image contrast, i.e., image contrast that combines the visual characteristics of the human eye. Combining the visual characteristics for the gradient value and enhancement function improvement, such as improving the contrast based on the original image after convolution with a Gaussian kernel, in this algorithm, mainly from the perspective of the critical visible deviation, the physical contrast of the image is used in the first-order gradient of the input image. The visually perceived image contrast $Vis_{ed}(x,y,t)$ is denoted as:

$$Vis_{cd}(x,y,t)=\frac{R_{cd}(x,y,t)}{JND(x,y,t)} \tag{14}$$

where $Vis_{cd}(x,y,t)$ is the visually perceived image contrast, $R_{cd}(x,y,t)$ is the physical image contrast, i.e., the first-order gradient of the input image, and $JND(x,y,t)$ is the critical visible deviation.

In the specific algorithm, the visually perceived contrast does not change the direction of the original gradient, but only the modulus of the gradient. The formula is as follows:

$$Vis=\frac{\nabla I(x,y)}{JND(x,y)}\times\frac{\nabla I}{\|\nabla I\|} \tag{15}$$

Where, I is the input image, $\nabla I(x,y)$ is the first order gradient of the input image, $JND(x,y)$ is the visible deviation to the human eye, and $\frac{\nabla I}{\|\nabla I\|}$ represents the orientation information of the gradient field. It can be seen that for the same absolute gradient, the value of the visible deviation JND of the human eye is smaller where the background brightness is larger, and the value of the visible deviation JND of the human eye is larger where the background brightness is smaller, so the same absolute gradient will adaptively change with the difference of the visible deviation of the human eye, thus obtaining the visual perceptual contrast, which is in line with the characteristics of the visual perception of the human eye.

This improvement introduces the visual perceptual contrast to replace the physical contrast of the image, so that the first-order gradient of the image is not a constant value, but through the visible deviation of the human eye to achieve the effect of adaptive adjustment, and the reconstruction of the image after enhancement is the same as the above method. This improved algorithm is more in line with the visual characteristics of the human eye, and has a better enhancement effect for a large number of actual images. In order to quantitatively verify the image enhancement effect of the model, two widely used image quality assessment metrics, namely, peak signal-to-noise ratio (PSNR) and structural similarity (SSIM), are adopted.

2.3. Research on the effect of image gradient enhancement

In order to ensure the fairness as well as comprehensiveness of the comparison, this paper conducts comparative experiments on five public datasets, the test sets include SET5, SET14, BSDS100 and URBAN100.To verify the effectiveness of the image gradient enhancement proposed in this paper, it is

compared with several existing method models including Bilinear Difference Algorithms (Bi), Double-Triple Difference based Convolutional Networks (SRCNN), SelfExSR, a method based on the combination of channel attention and spatial attention (DRCN), ESPCN (Efficient Sub-pixel Convolutional Neural Network), SRGAN (Super-Resolution Convolutional Neural Network), Fast Super-Resolution Reconstructive Convolutional Neural Network (FSRCNN), Deep Joint Convolutional Neural Network (VDSR), Rotational Regression Network (DRRN), LapSRN, Iterative Error Reconstruction Network (SRDenseNet), and Residual Connection and Convolutional Neural Network (SRResNet) based. The effect of these method models is examined on the four validation datasets mentioned above, and comparisons between low quality images (low resolution) and high quality images (high resolution) are performed separately, evaluated in terms of Peak Signal to Noise Ratio (PSNR) and Structural Similarity (SSIM) as quantitative metrics.

By comparing the original HD image with the enhanced image pixel by pixel, what is obtained by this method is not the quality of the image in the strict sense, but the degree of similarity or the degree of fidelity between the reconstructed image and the original HD image, of which the more classical method is the Peak Signal-to-Noise Ratio (PSNR).The Structural Similarity Index (SSIM), which measures the degree of similarity of the image in terms of the image's luminance, contrast, and structure, respectively. The structural similarity, SSIM, measures the similarity of images in terms of brightness, contrast, and structure, respectively.

Comparison of high-quality image enhancement effects is shown in Table 1, for high-quality image enhancement with higher resolution, the PSNR and SSIM based on the improved image gradient enhancement algorithm on the four datasets are effectively improved, with the PSNR ranging from 26.13 to 32.31, and the SSIM above 0.8. From the PSNR and SSIM metrics, it achieved better enhancement results compared to other models. Especially on SET5 the effect is better with PSNR and SSIM of 32.31 and 0.985 respectively.

Table 1. The image enhancement effect of high quality quantity is compared

Dataset	SET5		SET14		BSDS100		URBAN100	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Bi	28.54	0.902	26.12	0.810	25.97	0.750	23.25	0.745
SRCNN	30.63	0.953	27.63	0.850	26.92	0.790	24.65	0.812
SelfExSR	30.44	0.950	27.55	0.850	26.86	0.791	24.93	0.827
DRCN	31.68	0.975	28.06	0.871	27.25	0.803	25.25	0.840
ESPCN	29.35	0.940	26.40	0.845	25.50	0.775	24.13	0.816
SRGAN	29.60	0.930	26.60	0.820	25.75	0.745	24.60	0.826
FSRCNN	30.81	0.954	27.70	0.855	26.98	0.795	24.71	0.815
VDSR	31.65	0.974	28.05	0.867	27.30	0.805	25.30	0.842
DRRN	31.82	0.975	28.25	0.873	27.40	0.808	25.55	0.853
LapSRN	31.67	0.975	28.10	0.870	27.35	0.807	25.34	0.842
SRDenseNet	32.16	0.980	28.51	0.875	27.55	0.815	26.16	0.870
SRResNet	32.20	0.980	28.50	0.880	27.62	0.820	26.20	0.873
Ours	32.31	0.985	28.57	0.890	27.83	0.828	26.13	0.883

Comparison of low-quality image enhancement effect is shown in Table 2, for low-quality image enhancement with lower resolution, the improved gradient enhancement algorithm in this paper is obviously inferior to the high-quality image, although the effect of the algorithm is improved to a certain extent compared with some algorithms. Its performance on SET5 and SET14 datasets is not as good as LapSRN and SRDenseNet, and the SSIM on BSDS100 and UEBAN100 datasets is not more than 0.6, which shows that the improved gradient enhancement algorithm based on image gradient domain has good effect on high-quality image enhancement but weak enhancement effect on low-quality images. Weaker.

Table 2. Comparison of the effect of low quality image enhancement

Dataset	SET5		SET14		BSDS100		URBAN100	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Bicubic	24.52	0.727	23.22	0.640	23.78	0.581	20.85	0.545
SRCNN	25.45	0.760	23.83	0.665	24.25	0.605	21.32	0.575
SelfExSR	25.62	0.775	24.05	0.675	24.30	0.609	21.91	0.606
DRCN	25.15	0.767	23.40	0.668	24.02	0.615	21.32	0.585
ESPCN	23.15	0.696	21.60	0.565	21.90	0.483	19.74	0.500
SRGAN	25.52	0.752	23.95	0.663	24.35	0.607	21.45	0.567
FSRCNN	25.83	0.782	24.23	0.680	24.50	0.616	21.64	0.591
VDSR	25.87	0.791	24.22	0.655	24.55	0.575	21.11	0.562
DRRN	26.23	0.807	24.36	0.692	24.61	0.625	21.90	0.614
LapSRN	26.10	0.778	24.25	0.673	24.56	0.572	21.77	0.595
SRDenseNet	26.73	0.826	24.56	0.695	24.76	0.627	22.15	0.620
SRResNet	24.52	0.727	23.21	0.638	23.78	0.587	20.84	0.545
Ours	25.22	0.803	23.96	0.663	24.51	0.600	21.97	0.590

3. Multiscale visual feature enhancement based on convolutional neural network

Although image enhancement based on image gradient domain can improve the contrast of visual perception to a certain extent, its enhancement effect is not ideal in the face of low-quality images. For this reason, this paper proposes a visual feature enhancement method for low-quality images based on convolutional neural networks to improve the image enhancement quality problem.

3.1. Convolutional Neural Networks

In recent years, deep learning techniques represented by convolutional neural networks (CNNs) have made a big splash in the field of computer vision. The Le Net network for recognizing handwritten math styles was the first time that convolutional neural networks were used in practical applications. The development of convolutional neural networks was once limited by hardware resource constraints and the dependence on training data labels. With the development of computer hardware and the release of large datasets, such as the ImageNet dataset, the research of CNNs in the field of computer vision, such as image classification, image super-resolution, target detection, etc., has entered into a spurt of development.

Convolutional neural network is a kind of feed-forward neural network, which mainly consists of one or more layers of convolution, pooling layer and the top fully connected layer and other structures, and extracts the features of the original input data by stacking them layer by layer, and in this paper, for the task of low-quality image enhancement, we mainly use the following kinds of structures:

1) The convolutional layer is an important feature extractor in the convolutional neural network, different convolutional kernels represent different types of feature extraction, the features extracted by convolution are called feature maps, and the number of convolutional kernels determines the number of feature maps produced. The convolution kernel of size 3×3 and the elements of the input image at the sliding position are multiplied and summed in an operation that maps the information in the receptive field to an element in the feature map. All the elements in a feature map are computed by a single convolution kernel, which means that a feature map shares the same weights and bias terms, and such a computational strategy reduces the memory footprint and the complexity of the network model. The convolution operation of a single layer can only obtain simple features such as edges, for example, and more complex features can be obtained by the stacking of convolutional layers, and the convolution of later layers.

The convolution kernel parameter setting affects the size of the output feature map, where the size of the convolution kernel is denoted as (FH, FW), the step size, which is the distance the convolution kernel slides, is denoted as S, and the padding is P. If at this point the size of the inputs is (H, W), and the size of the outputs is (OH, OW), the size of the outputs is:

$$OH = \frac{H + 2P - FH}{S} + 1 \tag{16}$$

$$OW = \frac{H + 2P - FH}{S} + 1 \tag{17}$$

In practice, the size of the convolution kernel also affects the feeling field and the amount of computation, too large convolution kernel will lead to excessive memory occupation, so the 3 × 3, 5 × 5 small convolution in the actual application of large-scale application, there is also a 1 × 1 convolution kernel is more special, 1 × 1 convolution will not change the size of the input features, is often used to carry out the channel of the upgrading and downgrading operations, and at the same time 1 × 1 convolution with the activation function added to the network can also enhance the nonlinearity of the network. 1 convolution with the activation function added to the network can also enhance the nonlinearity of the network. In addition, the combination of 1×1 convolution and other convolutions can also reduce the amount of operations, which is typical of the depth separable convolution.

Suppose the size of the input feature map is $D_p \times D_p \times M$, the size of the convolution kernel is $D_K \times D_K \times M$, the feature map to be output is $D_F \times D_F \times N$, and the number of parameters for a standard convolution is:

$$P = (D_K \times D_K \times M) \times N \tag{18}$$

Depth separable convolution divides the convolution process into two steps to complete the process, depth convolution deals with the features of only one channel, and then the features on different channels are fused by point-by-point convolution which is also known as 1×1 convolution, where the size of depth convolution is $D_K \times D_K \times 1$, the size of point-by-point convolution is $1 \times 1 \times M$, and the number of parameters for depth separable convolution is:

$$P = D_K \times D_k \times M + M \times N \tag{19}$$

The ratio on the number of visible depth-separable convolutional parameters to the number of standard convolutional parameters is:

$$R = \frac{D_K \times D_k \times M + M \times N}{D_K \times D_K \times M \times N} = \frac{1}{N} + \frac{1}{D_K^2} \tag{20}$$

2) Pooling layer. The information contained in digital images often has a large number of redundant parts, and the introduction of a pooling layer allows for feature dimensionality reduction of the data, reducing the memory footprint and compressing feature information. Pooling operation is not carried out across channels and does not change the number of channels, the main types of pooling are maximum pooling and average pooling.

Pooling operation divides the feature map into regions of the same size by setting the size of the window, and extracts a value within the region as a regional feature point. Introducing a pooling layer reduces the size of the feature map while expanding the receptive field for the subsequent convolution kernel, so that the convolution of the latter layer can extract advanced features.

In addition the pooling layer is different from the convolutional layer, the Kernel size of the pooling layer refers to the size of the local partition, which has no parameter that needs to be learned. In addition the pooling layer has the feature of local translation invariance, when the target feature is translated, rotated, scaled, etc. in the region, the pooling result does not change. Using this feature, pooling layers are widely used in face recognition tasks. For example, when the face is located in a specific position in the image, the size and direction of the occupied image is not the main object of concern, only need to pay attention to the image contains features related to the five senses, through the introduction of the pooling layer, when the face occurs in the movement and other operations, the network can still identify the features, so as to improve the accuracy of the recognition. Of course, the translation distance is related to the pooling parameters, too large a translation distance will lead to changes in the features extracted by the pooling layer.

3) Activation function. Activation function is an important part of the convolutional neural network, purely convolutional operation of the linear model is difficult to meet the needs of reality, through the introduction of activation function can increase the nonlinear ability, so as to meet the demand for nonlinear expression in the actual scene. This paper mainly introduces the commonly used Sigmoid, ReLU, PReLU activation functions.

Sigmoid function is a graceful curve similar to the shape of S, and its mathematical definition can be expressed as:

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{21}$$

Its derivative is:

$$\sigma'(x) = \frac{1}{(1 + e^{-x})^2} + \frac{1}{1 + e^{-x}} = \sigma(x)(1 - \sigma(x)) \tag{22}$$

The Sigmoid function converts any value of the input between 0 and 1 as a continuous function for easy optimization. Its derivatives on both sides tend to be close to 0, which can easily cause the gradient to disappear, resulting in neurons entering the saturation region.The Sigmoid function will

change the value of the output extremely slowly after the input reaches a certain threshold, according to which it is often applied in such as the Conv LSTM, the Channel Attention Mechanism, and so on.

In order to avoid the gradient explosion phenomenon brought by Sigmoid, ReLU activation function is introduced into the neural network, and its mathematical form is expressed as:

$$f(x) = \max(0, x) \tag{23}$$

Its derivative can be expressed as:

$$f'(x) = \begin{cases} 1, x < 0 \\ 0, x \geq 0 \end{cases} \tag{24}$$

When $x > 0$, ReLU does not exist the state of gradient saturation, only a threshold is needed to obtain the activation value, and it has a simple functional form with lower computational complexity.

The unilateral inhibition of the ReLU function gives the neural network model a certain sparse expression ability, and its saturation can effectively solve the problem of gradient vanishing, providing a relatively wide activation boundary. However, when the input data is less than or equal to 0 the gradient is directly placed at 0, resulting in the parameters of this neuron not being updated. In order to solve this problem, He et al. proposed the PReLU activation function based on ReLU, so that the part of its input less than 0 can still be activated, and its mathematical expression is:

$$f(x) = \begin{cases} x_i, if\ x_i > 0 \\ a_i x_i, if\ x_i \leq 0 \end{cases} \tag{25}$$

4) Fully Connected Layer. For classification and regression tasks, a fully connected layer needs to be introduced to map the 2D features extracted by operations such as convolutional pooling to the label space of the sample. Each neuron in the fully connected layer needs to be connected to all neurons in the previous layer, which makes the number of parameters occupied by the fully connected layer larger. Taking AlexNet as an example, the parameters of the fully connected layer in the network account for 96.19% of all the parameters, and due to the redundancy of the parameters, subsequent researchers have proposed to use a global average pooling layer instead in order to reduce the number of parameters. The core of the fully connected operation is the product of matrix vectors, and the nonlinear capability is improved by introducing activation functions. The fully connected layer discards the two-dimensional spatial features, which is beneficial for classification tasks, which tend to be sensitive to the feature points and often independent of the location of the feature points.

3.2. Image enhancement based on multi-scale features

Due to the powerful feature extraction capability of deep convolutional networks, deep learning-based low-light image enhancement methods are gradually becoming mainstream. Although deep learning-based methods have made great progress, they are not applicable to all low-quality images, and the obtained enhancement results still suffer from noise amplification and color distortion. These problems seriously affect the quality of image restoration and subsequent advanced vision tasks. To address the problems, this paper proposes a low-light image enhancement network based on multi-scale feature complementation (LIEN-MFC) on the basis of image gradient enhancement. The LIEN-MFC model is then utilized on the basis of image gradient enhancement to achieve better visual effects.

The LIEN-MFC framework is shown in Fig. 1, which is a codec network with a U-shaped structure. In which the encoder performs feature extraction on multi-scale low-light images, and the decoder complements and fuses the multi-scale features extracted by the encoder to achieve feature reconstruction at each scale. In order to better recover the color and content of the image, this paper defines a joint loss function to train the network. A saturation loss function is innovatively constructed in the loss function to better recover image contrast and color information and prevent image overexposure.

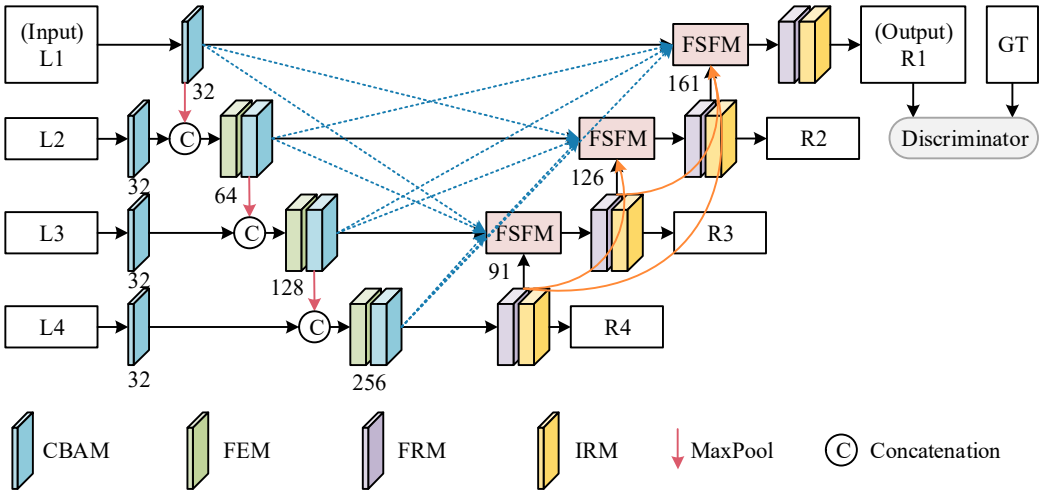

Fig. 1. Overall framework of LIEN-MFC

The proposed LIEN-MFC is designed as a multi-supervised codec network with four branches, each of which extracts and recovers features at one scale. Thus, LIEN-MFC takes four low light images of different scales as input. The input images are obtained by downsampling the original size low light images. Also, to ensure the accuracy of the recovered features at each scale, the corresponding normal light images are downsampled to supervise the output of each branch. The low light images at different scales can be obtained by the following equation:

$$L_{n+1} = Down(L_1, 2^n) \tag{26}$$

Where L_1 is the input image, $L_n(n=1,2,3,4)$ is the low light image at different scales. $Down(\cdot)$ denotes the downsampling operation and $2^n(n=1,2,3)$ denotes the downsampling multiplier.

In the LIEN-MFC encoder, the feature extraction module and the convolutional attention module are constructed. In each branch, the feature extraction module is utilized to extract shallow features from the low-light image of the corresponding scale. In the second, third and fourth branches, the convolutional attention and feature extraction modules were further utilized to extract and enhance deeper features. In order to improve the completeness of the features extracted in each branch, the shallow features in each branch are given complementary information through a maximum pooling operation and then are integrated into the next scale branch. The processing of the encoder can be represented by the following equation:

$$EnO_1 = FE(L_1) \tag{27}$$

$$EnO_n = FE(CBA(Cat(MP(EnO_{n-1}), FE(L_n)))) \tag{28}$$

where $EnO_n(n=1,2,3,4)$ denotes the output feature map of each branch in the encoder, $FE(\cdot)$ denotes the feature extraction module consisting of two convolutional layers, and $CBA(\cdot)$ denotes the convolutional attention module. In addition, $Cat(\cdot)$ and $MP(\cdot)$ denote the splicing operation and the maximum pooling operation, respectively.

In the decoder, all four branches contain a feature recovery module (FRM) and an image reconstruction module (IRM), which can realize feature recovery and image reconstruction at each scale.The FRM consists of five residual modules connected in series, and the IRM contains two convolutions and a ReLU operation. They enable feature mapping with enhanced images. In order to better utilize the features at each scale, FSFM is constructed and used in the first, second, and third branches, which can integrate the features at different scales in the decoder and the features reconstructed by the encoder. The decoder implementation process is represented as:

$$DeO4 = FRM(EnO4), R4 = IRM(DeO4) \tag{29}$$

$$DeO_3 = FRM(FSFM(EnO_1, EnO_2, EnO_3, EnO_4, Up(DeO_4))$$
$$R3 = IRM(DeO3) \tag{30}$$

$$DeO2 = FRM(FSFM(EnO1, EnO2, EnO3, EnO4, \ldots, Up(DeO3), Up(DeO4)))$$
$$R_2 = IRM(DeO2) \tag{31}$$

$$DeO_1 = FRM(FSFM(EnO_1, EnO_2, EnO_3, EnO_4)$$
$$Up(DeO_2), Up(DeO_3), Up(DeO_4)))$$
$$R_1 = IRM(DeO_1) \tag{32}$$

where DeOn denotes the output of the FRM in each branch in the decoder, $FSFM(\cdot)$ denotes the FSFM module, $FRM(\cdot)$ denotes the FRM module, $R_n(n = 1,2,3,4)$ denotes the reconstructed image in each branch, and R_1 is the final enhanced image. In the following we describe the structure of FSFM in detail.

FSFM first integrates features from encoder and decoder at different scales, and then further fuses and enhances the combined features of the two parts through the channel attention module.

In order to better constrain the enhanced image to be closer to the normal light image, a joint loss function is defined in this paper it includes five loss terms, i.e., content loss, structural loss, perceptual loss, saturation loss, and adversarial loss:

$$L_{lotal} = \alpha_1 L_c + \alpha_2 L_s + \alpha_3 L_p + \alpha_4 L_{sa} + \alpha_5 L_a \tag{33}$$

where $L_c, L_s, L_p, L_{sa}, L_a$ denotes content loss, structural loss, perceptual loss, saturation loss, and confrontation loss, respectively.

loss and confrontation loss. The other $\alpha 1, \alpha 2, \alpha 3, \alpha 4, \alpha 5$ is the weight of different loss items.

1) Content loss. The content loss includes the smoothed L1 loss and the cosine similarity loss, which mainly ensure that the content of the recovered image is closer to that of the normal light image at each scale. The content loss is defined as follows.

$$L_c = \sum_{i=1}^{4} \lambda_i (SmoothL1(R_i, GT_i) - Cosine(R_i, GT_i)) \tag{34}$$

where λ_i is the weight of the content loss at the i nd scale, R_i is the enhancement result at the i th scale, and GT_i is the normal light image at the i th scale. $SmoothL1(\cdot)$ denotes the smoothed L1 loss, and $Cosine(\cdot)$ denotes the cosine similarity loss.

2) Structural loss. Structural similarity (SSIM) loss is defined as a measure of the similarity between images in terms of brightness, contrast and structure:

$$L_s = 1 - SSIM(R_1, GT_1) \tag{35}$$

where SSIM(-) denotes the function that calculates the SSIM metrics for both images.

3) Perceptual loss. Perceptual loss is a loss that measures image similarity by comparing the feature differences between two images, which more accurately simulates human perception of images. Therefore, this work constructs a perceptual loss using the feature maps extracted from the first and second layers of the pre-trained VGG-16:

$$L_P = \| \phi(R_1) - \phi(GT_1) \|_2^2 \tag{36}$$

where $\phi(\cdot)$ denotes the feature maps extracted from the first and second convolutional layers of VGG-16.

4) Saturation loss

In order to ensure that the color of the recovered image is closer to the color of the normal light image, we propose a specific loss function to solve this problem. This function is based on the color

moments of the image, and it describes the distribution of colors by calculating the central moments of the image. Since the color information is mainly distributed in the lower order moments, the use of first-order moments, second-order moments and third-order moments is sufficient to express the color distribution of an image. We experimentally demonstrate that second-order moments and third-order moments are difficult to converge during training, and first-order central moments can constrain image recovery. Therefore we choose the first-order color moments of the image as the saturation loss function. The original color moments are obtained by computing the three channels of the RGB image. However, there is no direct connection between the R, G, and B values and the three attributes of color, which cannot reveal the relationship between colors. For this reason, we convert the color modes of the normal light image and the enhanced image to HSV mode to obtain the saturation component of the normal light image and the saturation component of the restored image. The saturation loss is calculated:

$$L_{sa} = \left| \frac{1}{HW} \sum_{i=1}^{H} \sum_{j=1}^{W} S_{i,j}^{R} - \frac{1}{HW} \sum_{i=1}^{H} \sum_{j=1}^{W} S_{i,j}^{T} \right| \tag{37}$$

Where i, j denotes the index of the pixel position. H, W is the height and width of the image. S^T is the saturation component of GT_i and S^R is the saturation component of R_1.

5) Adversarial Loss

Adversarial loss serves as the core of generating an adversarial network model, which helps the network to learn the style of cross-domain images. Therefore, in order to make the distribution of the enhanced image closer to the distribution of the normal light image, we define an adversarial loss to ensure that the enhanced image is more visually realistic. In the network model we use the discriminator in PatchGAN to train with the proposed network to constrain the enhancement results of the network:

$$L_a = -3HWNlog(Sigmod(PD(R_1))) \tag{38}$$

where $Sigmod(\cdot)$ is the Sigmoid function and $PD(\cdot)$ is the discriminator of PatchGAN. N denotes the number of branches in LIEN-MFC.

4. Analysis of image enhancement effects

In order to verify the effectiveness of the proposed algorithm, the enhancement performance of the proposed model image is tested in this subsection and compared with a variety of existing low-light image enhancement algorithms, the selected comparison algorithms contain the traditional low-light image enhancement algorithm CLAHE (Contrast Limited Adaptive Histogram Equalization), BIMEF, LIME, Dong, and deep learning based RetinexNet, MBLLEN algorithms. In this paper, image gradient enhancement is performed on the LOL dataset first and then the obtained images are used as samples to test the different models respectively.The original LOL dataset contains a total of 1000 sets of synthetic low light images with a resolution of 384×384 and their real images.

4.1. Comparison of enhancement effects of different algorithmic models

In this paper, the peak signal-to-noise ratio PSNR and structural similarity SSIM are still used as the evaluation index of image enhancement effect, and the PSNR comparisons of the models of different algorithms are shown in Fig. 2. From the comparison of PSNR, the average value of PSNR of the model in this paper is 21.80, which is improved by 0.44 compared with MBLLEN algorithm.In the quantitative analysis of objective evaluation indexes of PSNR for images of closet, desktop, sofa, bookshelf, vase, window sill, gymnasium, pool, etc., the enhancement of the proposed model in this paper outperforms the other comparison algorithms in terms of PSNR.

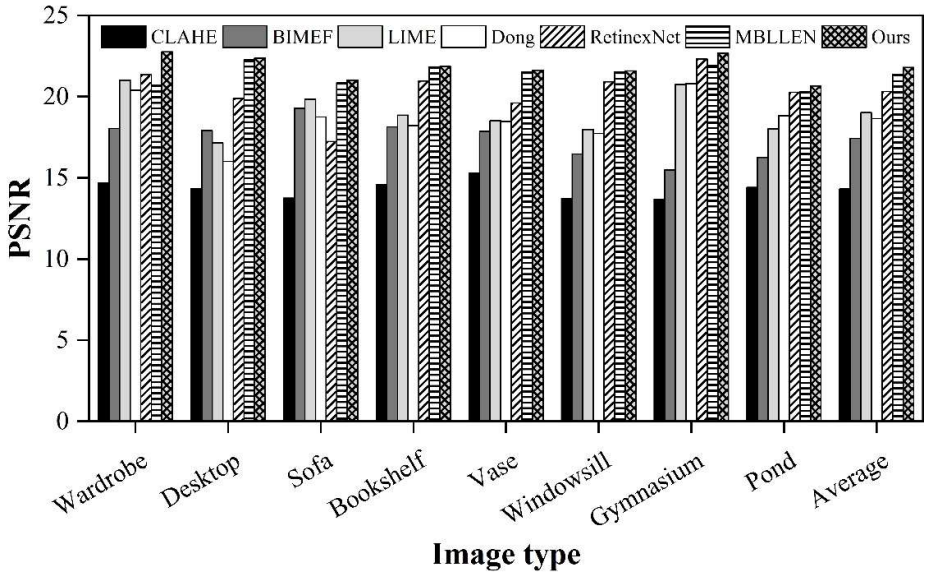

Fig. 2. PSNR comparison of different algorithm models

Comparison of SSIM of different algorithm models is shown in Fig. 3, from the comparison of SSIM index, the average value of SSIM of this paper's method is 0.8767, which is improved by 0.014 and 0.106 compared with the better performance of MBLLEN and RetinexNet, respectively.Taken as a whole, the image enhancement of the model proposed in this chapter is better than the other comparative algorithms in terms of both PSNR and SSIM. The experimental results show that the algorithm proposed in this paper has better ability to improve low-quality images.

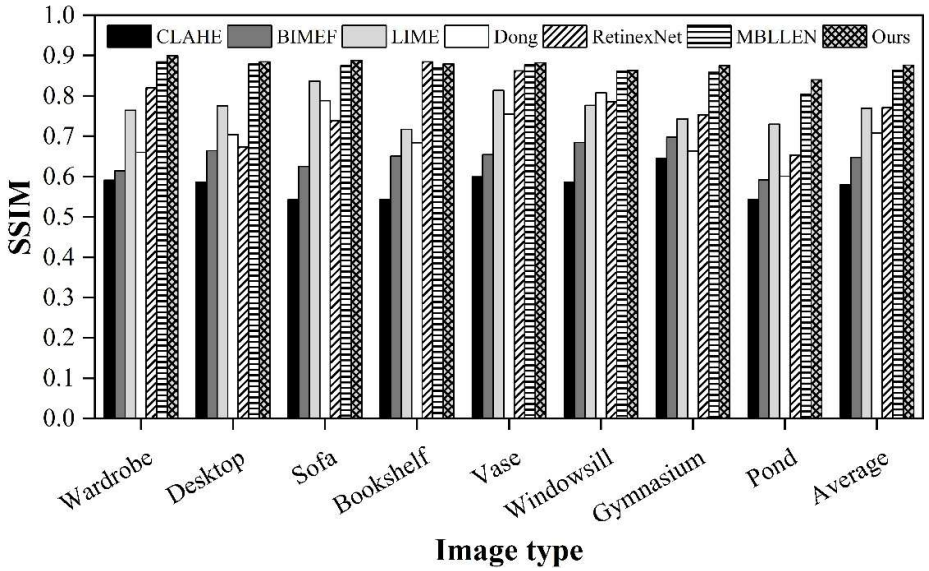

Fig. 3. The SSIM comparison of different algorithm models

4.2. Ablation experiments based on gradient enhancement

The algorithms in this chapter in targeting the image enhancement task first utilize gradient enhancement to process the image and then optimize it using the designed multi-scale convolutional network enhancement model. In order to verify the effectiveness of gradient enhancement, ablation experiments are conducted in this section for further verification. The model without gradient enhancement is compared with the model using gradient enhancement, and the results of the ablation experiment are shown in Table 3. The experimental comparison results show that the unused image

gradient-enhanced model reduces the average value on PSNR and SSIM by 0.7986 and 0.0335, respectively, which illustrates that the model utilizing the image gradient enhancement method effectively improves the image quality and the image enhancement effect is better.

Table 3. Ablation experiment results

Dataset	PSNR		SSIM	
	Unused	Use	Unused	Use
Wardrobe	21.1523	22.7575	0.8012	0.8998
Desktop	22.0213	22.3682	0.8452	0.8849
Sofa	20.9512	21.0146	0.8432	0.8885
Bookshelf	21.1235	21.8548	0.8621	0.8794
Vase	21.0121	21.6309	0.8512	0.8826
Windowsill	20.1532	21.5873	0.8566	0.8632
Gymnasium	21.5421	22.6682	0.8514	0.8751
Pond	20.0023	20.6662	0.8345	0.8401
Average	20.9948	21.7934	0.8432	0.8767

5. Conclusion

This paper focuses on image enhancement for machine vision, which is realized with the help of gradient enhancement and convolutional neural network for low quality images. For the image gradient field, this paper improves the image gradient enhancement algorithm. The PSNR and SSIM of the improved image gradient enhancement algorithm on high-quality images are effectively improved, with the PSNR ranging from 26.13 to 32.31 and the SSIM being higher than 0.8, but its enhancement effect on low-quality images is weak. For this reason, this paper proposes a LIEN-MFC model based on convolutional neural network that fuses multi-scale features. From the PSNR comparison, the average values of PSNR and SSIM of this paper's model are 21.80 and 0.8767, respectively, which achieve optimal results compared with other models. And compared with the better performance of MBLLEN, it improves 0.44 and 0.014 on PSNR and SSIM respectively.It is illustrated by the ablation experiments that the model using image gradient enhancement improves 0.7986 and 0.0335 on PSNR and SSIM on the average, which verifies that image gradient enhancement enhances the performance of LIEN-MFC model.

References

[1] Seifert, C., Aamir, A., Balagopalan, A., Jain, D., Sharma, A., Grottel, S., & Gumhold, S. (2017). Visualizations of deep neural networks in computer vision: A survey. Transparent data mining for big and small data, 123-144.

[2] Luo, H., Xiong, C., Fang, W., Love, P. E., Zhang, B., & Ouyang, X. (2018). Convolutional neural networks: Computer vision-based workforce activity assessment in construction. Automation in Construction, 94, 282-289.

[3] Patel, R., & Patel, S. (2020). A comprehensive study of applying convolutional neural network for computer vision. International Journal of Advanced Science and Technology, 29(6s), 2161-2174.

[4] Huang, W., Cheng, J., Yang, Y., & Guo, G. (2019). An improved deep convolutional neural network with multi-scale information for bearing fault diagnosis. Neurocomputing, 359, 77-92.

[5] Zheng, H., Chen, J., Chen, L., Li, Y., & Yan, Z. (2020). Feature enhancement for multi-scale object detection. Neural Processing Letters, 51, 1907-1919.

[6] Qi, Q., Li, K., Zheng, H., Gao, X., Hou, G., & Sun, K. (2022). SGUIE-Net: Semantic attention guided underwater

image enhancement with multi-scale perception. IEEE Transactions on Image Processing, 31, 6816-6830.

[7] Liang, H., Yang, J., & Shao, M. (2021). FE-RetinaNet: Small target detection with parallel multi-scale feature enhancement. Symmetry, 13(6), 950.

[8] Fan, X., Yang, Y., Deng, C., Xu, J., & Gao, X. (2018). Compressed multi-scale feature fusion network for single image super-resolution. Signal processing, 146, 50-60.

[9] Zhang, C., Chen, Y., Yang, X., Gao, S., Li, F., Kong, A., ... & Sun, L. (2020). Improved remote sensing image classification based on multi-scale feature fusion. Remote Sensing, 12(2), 213.

[10] Niu, A., Zhu, Y., Zhang, C., Sun, J., Wang, P., Kweon, I. S., & Zhang, Y. (2022). MS2Net: Multi-scale and multi-stage feature fusion for blurred image super-resolution. IEEE Transactions on Circuits and Systems for Video Technology, 32(8), 5137-5150.

[11] Huang, L., Chen, C., Yun, J., Sun, Y., Tian, J., Hao, Z., ... & Ma, H. (2022). Multi-scale feature fusion convolutional neural network for indoor small target detection. Frontiers in Neurorobotics, 16, 881021.

[12] He, M., Li, B., & Chen, H. (2017, September). Multi-scale 3D deep convolutional neural network for hyperspectral image classification. In 2017 IEEE International Conference on Image Processing (ICIP) (pp. 3904-3908). IEEE.

[13] Mustafa, H. T., Yang, J., & Zareapoor, M. (2019). Multi-scale convolutional neural network for multi-focus image fusion. Image and Vision Computing, 85, 26-35.

[14] Li, H., Zhao, W., Zhang, Y., & Zio, E. (2020). Remaining useful life prediction using multi-scale deep convolutional neural network. Applied Soft Computing, 89, 106113.

[15] Godinez, W. J., Hossain, I., Lazic, S. E., Davies, J. W., & Zhang, X. (2017). A multi-scale convolutional neural network for phenotyping high-content cellular images. Bioinformatics, 33(13), 2010-2019.

[16] d'Acremont, A., Fablet, R., Baussard, A., & Quin, G. (2019). CNN-based target recognition and identification for infrared imaging in defense systems. Sensors, 19(9), 2040.

[17] Chen, J., Du, L., Guo, G., Yin, L., & Wei, D. (2022). Target-attentional CNN for radar automatic target recognition with HRRP. Signal processing, 196, 108497.

[18] Qu, Z., Shang, X., Xia, S. F., Yi, T. M., & Zhou, D. Y. (2022). A method of single‐shot target detection with multi‐scale feature fusion and feature enhancement. IET Image Processing, 16(6), 1752-1763.

[19] Liu, F., Yang, X., & De Baets, B. (2023). A deep recursive multi-scale feature fusion network for image super-resolution. Journal of Visual Communication and Image Representation, 90, 103730.

[20] Chen, Y., Wang, Z., Peng, Y., Zhang, Z., Yu, G., & Sun, J. (2018). Cascaded pyramid network for multi-person pose estimation. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 7103-7112).

[21] Liu, Y., Zhong, Y., & Qin, Q. (2018). Scene classification based on multiscale convolutional neural network. IEEE Transactions on Geoscience and Remote Sensing, 56(12), 7109-7121.

[22] Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. (2017). Feature pyramid networks for object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2117-2125).