

<https://doi.org/10.70517/ijhssa463234>

Convolutional neural network based on image gradient information for multi-scale visual feature enhancement

Lanyue Pi^{1,*}, Yangzi Mu¹ and Lanyu Pi²

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

Corresponding authors: (e-mail: pi_zzcs0901@163.com).

Abstract This paper proposes an image gradient extraction method that fuses luminance information with chromaticity information. The method introduces the CIE-L*a*b* color model based on the human eye vision model to obtain the chromaticity gradient information on the basis of retaining the luminance gradient information, while the normalized luminance gradient and chromaticity gradient are later fused with the gradient. Afterwards, to balance the local contrast and color fidelity image enhancement effects, the Retinex algorithm with multiple scales was selected to better process the image. An end-to-end small-scale CNN model is constructed for color correction, and then a multi-scale Retinex model is used for texture enhancement, which integrates the color restoration and texture enhancement of the image, and the texture blurring and color deviation problems of the underwater image. The results show that the entropy value of the image increases from the original 6.5847 to 7.6014 after the processing of the method in this paper, which shows that the entropy value of the image can reflect the difference of the image contrast. After qualitative and quantitative analysis, it is found that the color bias of the underwater image enhanced by the method of this paper is effectively corrected, and the problem of color distortion is significantly improved. Meanwhile, the contrast, sharpness and saturation of the image are improved significantly, which proves the superiority of the proposed method in this paper compared with other methods in underwater image enhancement.

Index Terms Retinex algorithm, CNN model, image gradient, multi-scale visual features

I. Introduction

Convolutional neural network is a deep learning model that is widely used in computer vision [1], [2]. In image processing, multi-scale visual feature enhancement is an important research direction in convolutional neural networks [3]. In image processing tasks, the information contained in an image is usually multi-scale, and features of different scales are valuable for different tasks [4]-[6].

The role of multiscale feature enhancement is mainly reflected in the following aspects. First, multiscale feature enhancement can extract rich information of an image, including local details and global structure [7], [8]. By enhancing features at different scales, a more comprehensive and accurate image description can be obtained, thus improving the performance of image recognition and analysis [9], [10]. Second, multi-scale feature enhancement can improve the robustness of image recognition and analysis [11]. Features at different scales can compensate for each other's shortcomings, thus reducing errors caused by noise, illumination changes, and other factors [12], [13]. Finally, multi-scale feature enhancement can improve the real-time performance of the system [14]. By augmenting features at different scales, the feature dimension can be reduced, thus reducing the computational complexity and improving the real-time performance of the system [15], [16]. However, in the process of extracting feature information, convolutional neural network, due to the existence of step-by-step convolution and pooling layer, often lead to many fine-grained information is lost in the process of extracting features, and get the features with low learning rate, and the performance of convolutional neural network decreases sharply.

This paper first introduces the definition of image gradient, and then analyzes how to obtain image gradient in luminance space and chrominance space, and fuses luminance gradient and chrominance gradient in a unified gradient representation framework. For the foggy blurred image enhancement problem, a multi-scale Retinex-based algorithm is proposed to balance the local contrast and color fidelity image enhancement effect. After that, an end-to-end small-scale network structure based on CNN is constructed, and a simulated image dataset is used to train the network model. The model is then used to color correct the image to get the color restored image. Finally, the feature enhancement effect of convolutional neural network in multi-scale visual feature enhancement with image gradient information is analyzed.

II. Multi-scale visual feature image gradient information enhancement method by fusing CNNs

II. A. Image gradient extraction based on luminance and chromaticity

II. A. 1) Definition of image gradient

The image gradient [17], [18] is an effective feature used to characterize the drasticness of the change between image pixels. This variation includes both luminance and chrominance differences, and it corresponds to the first-order derivative of the image. When considering an image as a two-dimensional discrete function $I(x, y)$, the image gradient is the derivative of this two-dimensional discrete function:

$$G(x, y) = \frac{\partial I(x, y)}{\partial x} d_x + \frac{\partial I(x, y)}{\partial y} d_y \quad (1)$$

where: (x, y) is the two-dimensional coordinates of the image pixels; d_x and d_y represent the unit vectors along the horizontal and vertical directions of the image. For digital images, $\frac{\partial I(x, y)}{\partial x}$ and $\frac{\partial I(x, y)}{\partial y}$ can be used instead of first-order differentiation by first-order difference, i.e.:

$$\begin{cases} \frac{\partial I(x, y)}{\partial x} \leftarrow I(x+1, y) - I(x, y) \\ \frac{\partial I(x, y)}{\partial y} \leftarrow I(x, y+1) - I(x, y) \end{cases} \quad (2)$$

II. A. 2) Luminance-based image gradient extraction

In describing the brightness of an image, the YUV color space is used. The YUV color space was invented during the transition between color and black-and-white TVs. Black-and-white TVs utilize only the luminance component Y to represent the picture, and its luminance difference well matches the human eye's visual perception of luminance intensity.

Given an RGB color image tradition, its luminance Y component is obtained by the following formula:

$$Y = 0.299 \cdot R + 0.587 \cdot G + 0.114 \cdot B \quad (3)$$

The corresponding luminance-based image gradient is:

$$\Delta Y = \begin{bmatrix} \Delta Y_x \\ \Delta Y_y \end{bmatrix} = \begin{bmatrix} Y(x+1, y) - Y(x, y) \\ Y(x, y+1) - Y(x, y) \end{bmatrix} \quad (4)$$

The modulus of the luminance gradient is:

$$|\Delta Y| = \sqrt{\Delta Y_x^2 + \Delta Y_y^2} \quad (5)$$

II. A. 3) Chromaticity-based image gradient extraction

In describing the image chromaticity, CIE-L*a*b* color space is used. CIE-L*a*b* color space is an improved model based on CIEXYZ color space, which uses the degree of color difference described by MacAdam ellipse and builds a perceptual model of linearized color difference. The transformation process from RGB space to CIE-L*a*b* space is as follows:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.490 & 0.310 & 0.200 \\ 0.177 & 0.812 & 0.011 \\ 0.000 & 0.010 & 0.990 \end{bmatrix} \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (6)$$

$$\begin{cases} L^* = 116 \cdot f(Y/Y_0) - 16 \\ a^* = 500 \cdot [f(X/X_0) - f(Y/Y_0)] \\ b^* = 200 \cdot [(Y/Y_0) - (Z/Z_0)] \end{cases} \quad (7)$$

where: (X_0, Y_0, Z_0) is the value of the white reference point in CIE-XYZ color space. Using this white reference point, regularization in CIE-L*a*b* color space can be achieved:

$$f(t) = \begin{cases} t^+ & t > \left(\frac{6}{29}\right)^3 \\ \frac{1}{3}\left(\frac{29}{6}\right)^2 t + \frac{16}{116} & \text{other} \end{cases} \quad (8)$$

The nonlinear relationship between L^* , a^* and b^* intends to mimic the nonlinear response of the human eye to color, such that uniform numerical changes in the CIE- $L^*a^*b^*$ model correspond to uniform changes in color in human perception.

Representing the color of a pixel as a L^* , a^* and b^* ternary vector, the gradient of the chromaticity-based image is:

$$\Delta C = \begin{bmatrix} \Delta_x^L & \Delta_x^a & \Delta_x^b \\ \Delta_y^L & \Delta_y^a & \Delta_y^b \end{bmatrix} = \Delta C = \begin{bmatrix} L^*(x+1, y) - L^*(x, y), a^*(x+1, y) - a^*(x, y), b^*(x+1, y) - b^*(x, y) \\ L^*(x, y+1) - L^*(x, y), a^*(x, y+1) - a^*(x, y), b^*(x, y+1) - b^*(x, y) \end{bmatrix} \quad (9)$$

The modulus of the chromaticity gradient is:

$$|\Delta C| = \sqrt{\Delta E_x^{*2} + \Delta E_y^{*2}} \quad (10)$$

where: ΔE_x^* and ΔE_y^* are the color Euclidean distances in the CIE- $L^*a^*b^*$ color space, i.e:

$$\begin{cases} \Delta E_x^* = \sqrt{\Delta L_x^{*2} + \Delta a_x^{*2} + \Delta b_x^{*2}} \\ \Delta E_y^* = \sqrt{\Delta L_y^{*2} + \Delta a_y^{*2} + \Delta b_y^{*2}} \end{cases} \quad (11)$$

II. A. 4) Fusion of luminance gradient and chrominance gradient

Since the luminance gradient and the chrominance gradient do not have the same range of values, it is necessary to normalize each of the luminance gradient and the chrominance gradient first in the fusion process:

$$\begin{cases} \Delta Y' = \frac{\Delta Y}{\max |\Delta Y|} \\ \Delta C' = \frac{\Delta C}{\max |\Delta C|} \end{cases} \quad (12)$$

where: $\max |\Delta Y|$ and $\max |\Delta C|$ are the luminance gradient and chrominance gradient with the largest values in the image, respectively.

Since the luminance gradient domain chrominance gradient satisfies the linear transformation at the same time, the gradient representation that fuses the two information is:

$$G(x, y) = \sqrt{\Delta Y'^2 + \Delta C'^2} \quad (13)$$

II. B. Convolutional Neural Networks

Convolutional neural network is a very representative network structure in the field of computer vision, and with the rapid iterative upgrading of computer hardware equipment, convolutional neural networks have made significant progress in tasks such as image classification, instance segmentation and target detection.

II. B. 1) Convolutional layers

The convolutional layer is the core module of a convolutional neural network learning to capture local feature dependencies. Given an input feature map of any $H \times W \times C_{in}$ size, where H is the height of the feature map, W is the width of the feature map, and C_{in} is the number of channels in the feature map. Using C_{out} a $K \times K \times C_{in}$ -sized convolution operation, where K is the convolution kernel size, and the sliding step of the convolution is S , one can obtain $H' \times W' \times C_{out}$ size of the output feature map.

where the resolution size H' and W' of the output feature map are calculated as shown below:

$$H' = \left\lfloor \frac{H - K + 2P}{S} \right\rfloor + 1 \quad (14)$$

$$W' = \left\lfloor \frac{W - K + 2P}{S} \right\rfloor + 1 \quad (15)$$

where P is the edge padding number, which is used to adjust the resolution size of the output feature map in the convolution operation, when $P = \frac{K-1}{2}$, the resolution size of the output feature map is the same as the input feature map. The $\lfloor \cdot \rfloor$ is a downward rounding operation.

The convolutional layer is used as a new feature for the output feature map by calculating the sum of the products of the convolution kernel and the elements in the corresponding region of the input feature map. When the convolution operation is performed on the feature map F by the convolution kernel w , the formula is:

$$F'_{i,j} = \sum_{p=0, q=0}^{K-1, K-1} w_{p,q} F_{i+p-\lfloor \frac{K}{2} \rfloor, j+q-\lfloor \frac{K}{2} \rfloor} \quad (16)$$

II. B. 2) Pooling layer

The pooling layer is usually placed between two convolutional layers and plays an important role in the convolutional neural network structure. Through pooling operations, the resolution size of an image is effectively reduced without introducing additional parameters. Pooling layers are usually divided into two forms: maximum pooling and average pooling. By taking the maximum or average value of the elemental features in the sliding window region, the main information of the features is extracted or the noise information is reduced, which makes the model invariant to small positional variations and thus improves the generalization ability of the model.

II. B. 3) Full connectivity layer

A fully connected layer, also known as a densely connected layer or affine layer, is often located at the end of a neural network and is used to integrate and transform the features extracted earlier in the convolutional neural network in order to realize the network's classification prediction or regression prediction of the data. In the fully connected layer, each node is connected to all nodes of the input, and each connection is accompanied by a weight parameter, which has a very high computational complexity and causes huge memory consumption. With the development of convolutional neural networks, researchers have found that the computational complexity of convolutional operations is relatively low and the input is more flexible, which is more suitable for processing image information. Therefore, in later convolutional neural network structures, most of them use 1×1 convolutional operations instead of fully connected operations to improve the performance of the network model.

II. B. 4) Activation functions

The activation function is a core component of convolutional neural networks. The introduction of activation functions can improve the nonlinear modeling ability of neural networks. Common activation functions mainly include ReLU activation function, Sigmoid activation function and Tanh activation function.

ReLU activation function is the most used activation function in convolutional neural networks, which compares the input eigenvalues directly with 0, and the eigenvalues less than 0 are directly categorized as 0, which makes the ReLU activation function have nonlinear characteristics, and also can alleviate the problem of gradient disappearance; in other cases, the original eigenvalues are retained to make the model converge quickly. However, the ReLU activation function also has the disadvantages of neuron death and output asymmetry, etc. The ReLU activation function is calculated as follows:

$$ReLU(x) = \max(0, x) \quad (17)$$

The Sigmoid activation function compresses the input feature values to between 0 and 1 and is usually used for binary classification problems. However, the Sigmoid activation function creates a gradient vanishing problem when the input is larger or smaller intervals, making the gradient update too slow during the neural network learning process. The Sigmoid activation function is computed as:

$$\text{Sigmoid}(x) = \frac{1}{1 + e^{-x}} \quad (18)$$

The Tanh activation function is similar to the Sigmoid activation function, but the Tanh activation function maps the input eigenvalues to a range of values between $[-1, 1]$, with a wider range of outputs, and with a mean value close to zero. The Tanh activation function is computed as follows:

$$\text{Tanh}(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (19)$$

Depending on the tasks and related data handled by the neural network model, different activation functions are often chosen, so the activation function can be tested and optimized several times during the model training process to achieve better performance.

II. B. 5) Propagation algorithms

Propagation algorithm is one of the core techniques in neural network modeling, which is mainly used to train neural network models to adapt to specific tasks. The propagation algorithm mainly contains two processes: forward propagation and back propagation. First, in the forward propagation process, the neural network calculates the output value of each neuron layer by layer according to the flow direction of the input data. At the same time, the output of the network model is compared with the real label to get the error size of the model. In the process of back propagation, the gradient of each learnable parameter to the loss function is calculated by the chain rule, and the value of each learnable parameter is updated by using the gradient back propagation algorithm, which makes the model prediction error decrease gradually. Through the process of repeated iterative training, the neural network model gradually learns the abstract feature representation of the input data and realizes accurate prediction and classification results. The gradient back propagation algorithm plays a key role in the training process of the neural network, and it is necessary to choose the appropriate algorithm to optimize the model parameters according to the specific task and the characteristics of the data.

II. C. Image enhancement algorithm based on multi-scale Retinex algorithm

II. C. 1) Retinex Algorithm

Combining with the visual characteristics of human eyes, the former proposed the Retinex theory for the first time; and used the center-surround method to estimate the incident component of the image, and obtained a variety of single-scale (SSR) and multi-scale Retinex (MSR) algorithms [19], [20]. Various Retinex algorithms are widely used in foggy blurred image enhancement.

Retinex theory suggests that the shape and color of an object are determined by the object's ability to reflect light, and the surface features of an object are not affected by the non-uniformity of light and have consistency. The surface characteristics of an object are obtained by estimating the incident component and separating the reflected component from the original image. The visual image $S(x, y)$ can be composed of the incident component $L(x, y)$ and the object reflection component $R(x, y)$, and they can be expressed by the following equation:

$$S(x, y) = L(x, y) \times R(x, y) \quad (20)$$

where $L(x, y)$ is the incident component, which is affected by the environmental factors of light and shadow and determines the dynamic range of the image $S(x, y)$; $R(x, y)$ is the reflected component, which reflects the reflective nature of the object itself, and the ability of the object to absorb and reflect light is related to the nature of the object's surface.

In order to better separate the incident and reflected components, Eq. (20) is expressed logarithmically:

$$s(x, y) = l(x, y) + r(x, y) \quad (21)$$

where $s(x, y) = \lg(S(x, y))$, $r(x, y) = \lg(R(x, y))$, $l(x, y) = \lg(L(x, y))$. Since the incident component cannot be obtained directly, it is mostly approximated and estimated using certain mathematical models.

II. C. 2) Multiscale Retinex Algorithm (MSR)

In order to balance the image enhancement effect of local contrast and color fidelity, and to take into account the advantages of multiple scales of large, medium and small, the Retinex algorithm of multiple scales can be selected to achieve better image enhancement effect. To synthesize the image enhancement effects of different scales of Retinex algorithms, different scales of c values are selected, and the results of different scales of Retinex algorithms are weighted and averaged:

$$R_k(x, y) = \exp \left\{ \sum_{n=1}^N w_n [\lg(S_k(x, y)) - \lg(F_n(x, y) * S_k(x, y))] \right\} \quad (22)$$

where $k = 1, 2, \dots, K$ denotes the band number; $K = 1$ when dealing with grayscale images; $K = 3$ when dealing with color images, corresponding to R, G, and B components, respectively. $R_k(x, y)$ denotes the enhanced image in the k th band; $S_k(x, y)$ denotes the original image in the k th band; N denotes the number of scales; and w_n is the weight of the n th scale, which is satisfied:

$$\sum_{n=1}^N w_n = 1 \quad (23)$$

The MSR algorithm generally selects three scale convolutional kernels, small ($c < 20$), medium ($c = 20200$), and large ($c > 200$), and chooses the same set of weights (generally taken as $w_n = 1/3$) for image enhancement.

II. D. Image Enhancement Methods Fusing Deep Learning and Multiscale Retinex

II. D. 1) CNN-based image color restoration

CNN is suitable for areas such as processing underwater images with large amount of data and complex environmental changes. The neuron layer model of this network is as follows:

$$N(m, n) = f(W^T X + b) = f \left(\sum_{i=1}^m \sum_{j=1}^n [W(i, j) X(m+i, n+j)] + b(i, j) \right) \quad (24)$$

where m and n denote the pixel coordinates of the output image N after color restoration, X is the input of the neuron, $N(m, n)$ is the output of the neuron, W^T is the transpose of the weights matrix, b is the bias vector, and f is the transfer function acting between the inputs and the outputs.

To ensure that the inputs and outputs of each layer are nonlinear, this paper adopts the modified linear unit (ReLU) as the activation function for each convolutional layer in the network, with Eq:

$$\text{ReLU}(x) = \begin{cases} x, & x > 0 \\ 0, & x \leq 0 \end{cases} \quad (25)$$

where x denotes the value of a single pixel in each feature map.

During the network training process, a loss function is introduced to calculate the error between the trained and labeled images, and the loss value is continuously updated to optimize the loss value during the forward propagation process, and in this paper, we use the mean square error (MSE) as the loss function L_{MSE} :

$$L_{MSE} = \frac{1}{K} \sum_{i=1}^K \|X_i - N_i\|^2 \quad (26)$$

where K is the number of data in the training set, X_i is the exact value of the land-based image in the dataset, and N_i is the output image of the recovery network.

II. D. 2) Color space transformations

The color recovery image N is mapped from the RGB color space to the LAB color space, and its luminance channel (L component) is extracted for contrast and brightness adjustment, and then processed by multi-scale guided filter Retinex to obtain the texture enhancement image M , and then fused with the color channels (A and B components) of the image N for the CIELAB color space transformation [21] to obtain the final processed underwater enhanced image.

III. Analysis of the effect of feature enhancement based on image gradient information

III. A. Measurement of Aesthetic Information Based on Image Contrast

Contrast refers to the different brightness levels between the brightest white and darkest black in the light and dark areas of an image, i.e., the size of the gray-scale contrast of an image. A larger range of differences represents a greater contrast. Gray scale histogram is a function of the distribution of gray levels in an image. The histogram

equalization method `histeq` in MATLAB is used to enhance the image contrast, the test image is shown in Fig. 1, (a) and (b) represent the original image-contrast weak and equalized respectively.

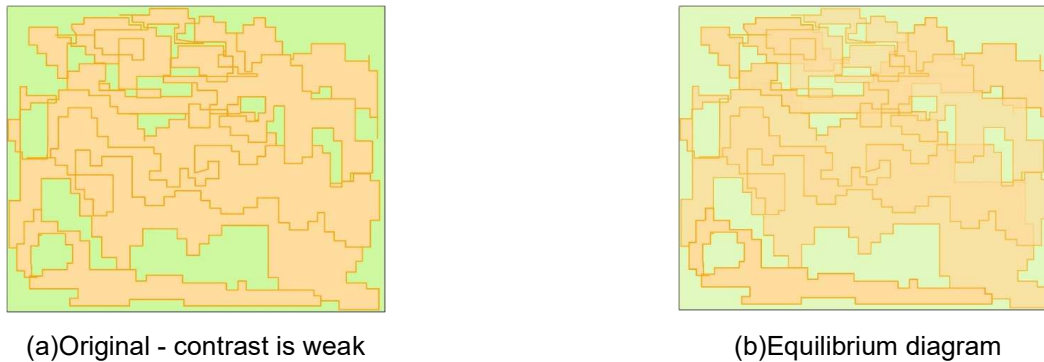


Figure 1: Test image

The MATLAB custom function `imgentropy.m` is further utilized to calculate the image entropy for comparison of entropy before and after image contrast enhancement. Histogram equalization for image contrast enhancement is shown in Fig. 2, where (a) and (b) represent the original grayscale histogram and the grayscale histogram after equalization, respectively.

From the experimental results, after histogram equalization, the contrast of the image is obviously enhanced, the visual effect is significantly improved, and the image quality is enhanced. From the analysis of the experimental results, the range of the gray value of the image histogram after `histeq` equalization is obviously enlarged; the histogram of the image after `histeq` equalization tends to be flat, and the gray level is reduced, which indicates that the gray level is merged; the entropy value of the image before contrast enhancement is 6.5847, and the entropy value of the image after contrast enhancement is 7.6014, which indicates that the entropy value of the image is able to reflect the difference in image contrast. In the calculation of entropy value of images with cross-reference, the entropy value shows the consistent rule of change with the aesthetic evaluation of contrast, i.e., the entropy value of contrast with better aesthetic evaluation is also higher. Therefore, image entropy can be used as a measure of universal aesthetic judgment of image contrast.

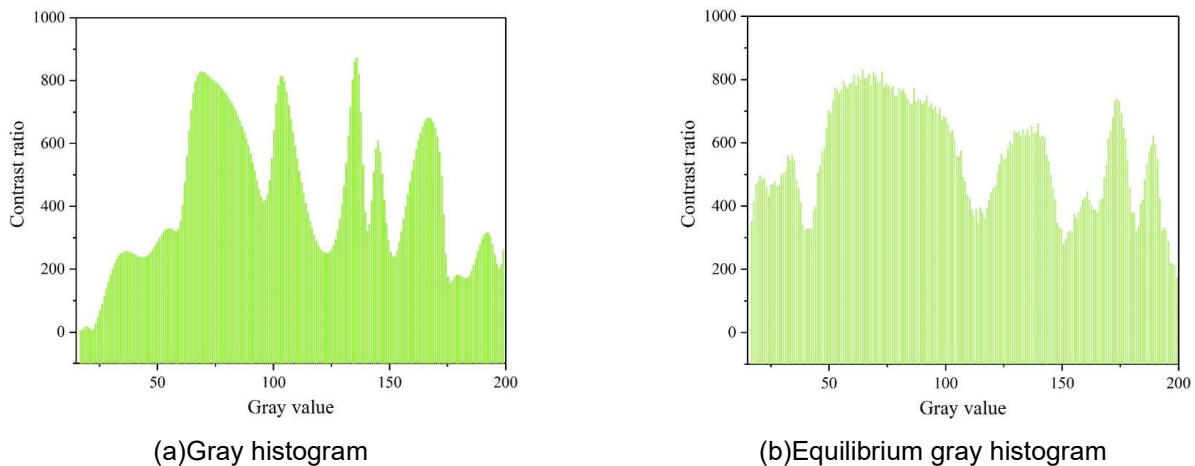
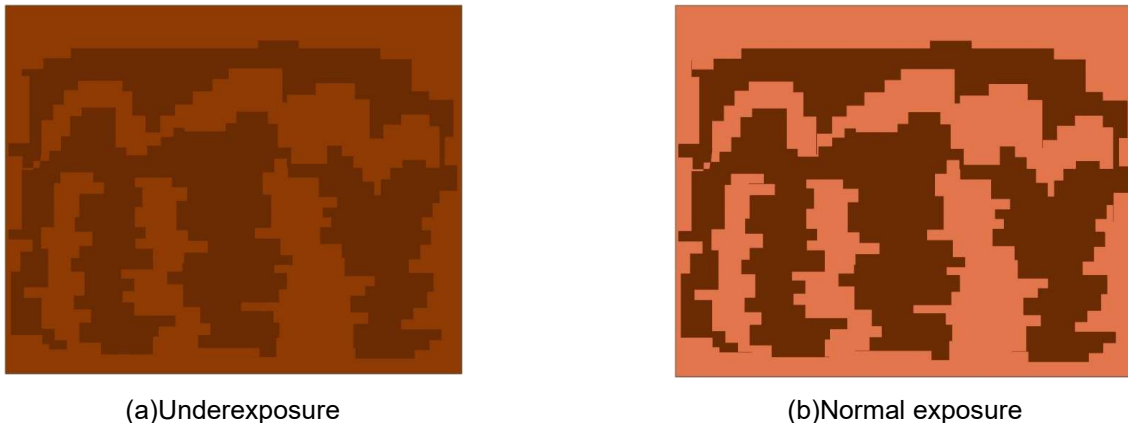


Figure 2: Histogram equalization enhancement image contrast

III. B. Image Gradient Based Measurement of Aesthetic Information

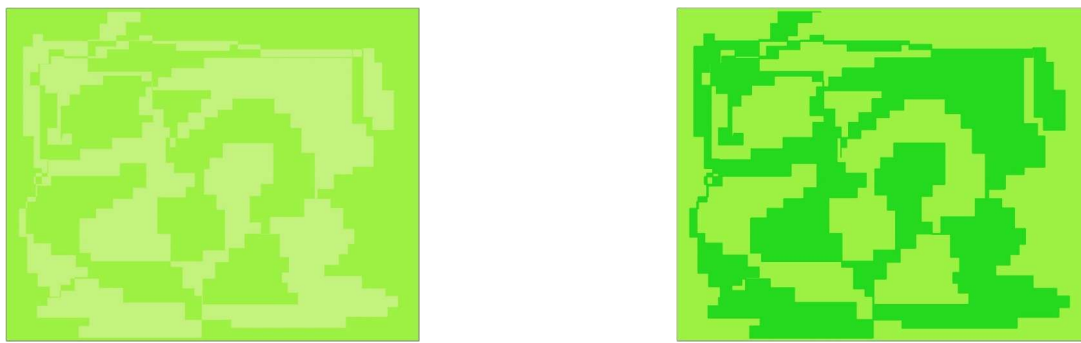
Two sets of images are selected, one with underexposed and normally exposed images as shown in Fig. 3, where (a) and (b) represent underexposed and normally exposed, respectively, and the other with overexposed and normally exposed images as shown in Fig. 4, where (a) and (b) represent overexposed and normally exposed, respectively.



(a) Underexposure

(b) Normal exposure

Figure 3: Exposure and exposure to normal images



(a) Overexposure

(b) Normal exposure

Figure 4: Exposure and exposure to normal images

Table 1: Comparison results of different exposure degree image gradient

Group	Image exposure	Gradient sum	Gradient mean t	Gradient mean sort
First group	Underexposure	6.5938	36.8741	4
	Normal exposure	2.1136	51.5266	3
Second group	Overexposure	1.5931	54.6970	2
	Normal exposure	2.0853	77.6105	1

The MATLAB custom function Ten.m was used to calculate the gradient of the images and the gradient values were compared for 2 groups of images respectively. The results of comparison of gradients of images with different levels of exposure are shown in Table 1. According to the analysis of the experimental results, the average value of the gradient of the underexposed images in group 1 is 36.8741, and the average value of the gradient of the normally exposed images is 51.5266, which corresponds to the quality of the underexposed images being significantly lower than that of the normally exposed images, and the average value of the gradient of the underexposed images is also similarly lower than the average value of the gradient of the normally exposed images; The average value of the gradient of the overexposed image in group 2 is 54.6970, and the average value of the gradient of the normally exposed image is 77.6105, which corresponds to the quality of the overexposed image being significantly lower than that of the normally exposed image, the average value of the gradient of the overexposed image is also lower than the average value of the gradient of the normally exposed image. Comparison of group 1 and group 2 in the exposure of normal images, the difference between the average value of the image gradient is more obvious, the average value of the gradient of group 2 images is higher than that of group 1 images, analyze the reason for this is that the texture of the image is more varied and hierarchical, the structure of the staggered alternation, the aesthetic experience from the group 2 images can be judged to be higher than the quality of the texture of the image of the group 1 images, and thus the value of the gradient of the image can be used as the objectivity and universality

scale of the aesthetic evaluation of the image texture. Therefore, the image gradient value can be used as an objective and universal scale for the aesthetic evaluation of image texture.

III. C. Analysis of image enhancement effect under different models

III. C. 1) Experimental environment preparation and comparative modeling

In order to better reflect the effectiveness of the method in this paper during the model testing phase, two publicly available datasets are used: the UIEB dataset and the EUVP dataset. Figures A and B selected from the datasets represent typical underwater environment images, where Figure A is heavily color-biased and presents a greenish tone, representing a color-distorted underwater image, and Figure B has a low contrast, representing a low-contrast image. This arrangement also better reflects the generalization ability of the model and the effect of image enhancement on different datasets. There are 250 epoch iterations throughout the training process with an initial learning rate of 0.0001, using the Adam solver as the optimizer, Adam's momentum interval is [0.5, 1], and the default value of learning decay iterations is 45.

The proposed method in this paper This method is compared with GDCP, UIBLA, Retinex_based, Two_step, GAN-RS, CycleGAN, FUnIE-GAN and pix2pix, respectively. The first four are traditional physical modeling methods based on MATLAB, and the last four are methods based on convolutional neural networks or conditional generation adversarial network based using PyTorch framework.

In this paper, real underwater images are taken as an example for comparison, and five reference-free quality evaluation methods are used in this paper, including entropy value, fog density based evaluation (DEFADE), Blur, underwater color image quality evaluation (UCIQE), and underwater image quality metrics (UIQM including UICM,UISM, and UIConM). The entropy value indicates the richness of image information, based on the fog density evaluation (DEFADE) can predict the visibility of an underwater scene (similar to a foggy scene) from a single image without reference to the corresponding waterless (fogless) image and without relying on significant objects in the scene, the fuzzy value is used as a non-reference evaluation metric to evaluate the image quality in terms of fuzzy perceptions, and its value ranges from 0 to 1, 0 denotes the worst result and 1 denotes the best result. UIQM, which consists of UICM, UISM and UIConM, represents the combined quality of the enhanced underwater image, and its sub-evaluations quantify the strengths and weaknesses of the underwater image in terms of color, sharpness and contrast. Similarly, UCIQE evaluates the quality of the image through chromaticity, average saturation and brightness.

Table 2: No reference quality evaluation is compared to the two groups

Image	Method	Entropy	UCIQE	Blur	DEFADE
Figure A	Original drawing	7.5904	0.4211	0.2819	0.4934
	UIBLA	7.7612	0.539	0.2302	0.6226
	GDCP	7.2847	0.5843	0.2415	0.2763
	Retinex_based	7.7664	0.5742	0.3508	0.58
	Two_step	7.5705	0.555	0.266	0.4829
	CycleGAN	7.8938	0.5947	0.2906	0.4341
	Pix2pix	7.8683	0.5969	0.2979	0.393
	FUnIE-GAN	7.8017	0.4697	0.332	0.2975
	GAN-RS	7.8006	0.5808	0.2694	0.3988
	Ours	7.9603	0.6279	0.2529	0.2874
Figure B	Original drawing	6.8001	0.4287	0.3838	0.7757
	UIBLA	7.5686	0.6106	0.3779	0.348
	GDCP	7.0203	0.5579	0.3546	0.4822
	Retinex_based	7.7995	0.6231	0.4195	0.2608
	Two_step	7.0177	0.5026	0.3694	0.4225
	CycleGAN	7.7925	0.644	0.4035	0.265
	Pix2pix	7.7173	0.5896	0.304	0.218
	FUnIE-GAN	7.2757	0.4951	0.3847	0.573
	GAN-RS	7.6571	0.5931	0.3712	0.2686
	Ours	7.8718	0.6451	0.3379	0.1842

III. C. 2) Quantitative analysis

The results of the quantitative comparison of the two groups of images with no reference quality evaluation are shown in Table 2, and the results of the quantitative comparison of the two groups of images with UIQM quality

evaluation are shown in Table 3. The quantitative analysis of the four tables shows that the method of this paper has a better improvement for both types of images, especially for the UCIQE and UIQM two comprehensive indexes to improve significantly, indicating that the method of this paper is more superior than other methods in color correction and enhancement of contrast performance.

Table 3: The UIQM quality evaluation is similar to the two groups

Image	Method	UICM	UISM	UIConM	UIQM
Figure A	Original drawing	7.5904	0.4211	0.2819	0.4934
	UIBLA	0.2729	1.2726	0.717	0.7568
	GDCP	6.2455	1.5107	0.9708	1.0492
	Retinex_based	3.8998	3.631	1.2003	1.4034
	Two_step	3.6739	2.1624	1.0897	1.1908
	CycleGAN	4.5765	2.3754	1.2394	1.3491
	Pix2pix	5.0781	2.7568	1.1644	1.3124
	FUnIE-GAN	5.5197	2.5940	1.2541	1.3832
	GAN-RS	1.7627	1.3673	0.8751	0.9126
Ours	4.7129	2.4536	1.1638	1.2866	
Figure B	Original drawing	6.5166	3.0183	1.2996	1.4665
	UIBLA	6.7994	0.4291	0.3845	0.7745
	GDCP	7.5685	0.6065	0.3795	0.3485
	Retinex_based	7.0195	0.5572	0.3543	0.4846
	Two_step	7.8003	0.6223	0.4192	0.2614
	CycleGAN	7.0169	0.5024	0.3694	0.4222
	Pix2pix	7.7923	0.6451	0.4039	0.2658
	FUnIE-GAN	7.7173	0.5891	0.3051	0.2177
	GAN-RS	7.2756	0.4940	0.3861	0.5730
Ours	7.6580	0.5921	0.3711	0.2679	

In this paper, the average of all quality assessment methods is used to reflect the comprehensive performance of each method. The quantitative comparison on the UIEBD dataset is shown in Table 4; the UIQM quantitative comparison on the UIEBD dataset is shown in Table 5. It can be seen that the method in this paper obtains the highest score in terms of underwater color image quality evaluation (UCIQE) and entropy value, which indicates that the proposed method in this paper is superior in maintaining image information and improving chromaticity, average saturation and contrast. Although GDCP performs better in blur removal, its visual perception is very poor for the human eye. As for the index based on fog density evaluation (DEFADE), the score of this paper's method is comparable to the performance of pix2pix (0.1702), and the composite score of UIQM outperforms that of the other compared methods (1.7407). Overall, the method proposed in this paper has good comprehensive performance in quality recovery of underwater images.

Table 4: Quantitative comparison in UIEBD data set

Image	Method	Entropy	UCIQE	Blur	DEFADE
Average	Original drawing	7.2655	0.516	0.251	0.4132
	UIBLA	7.2467	0.5935	0.2399	0.3016
	GDCP	7.2487	0.6101	0.2269	0.2628
	Retinex_based	7.6921	0.6055	0.3029	0.2902
	Two_step	7.4512	0.574	0.2431	0.2857
	CycleGAN	7.7211	0.619	0.2876	0.2909
	Pix2pix	7.7038	0.5889	0.2296	0.1702
	FUnIE-GAN	7.5196	0.5633	0.2797	0.3154
	GAN-RS	7.6669	0.5853	0.285	0.2598
	Ours	7.7615	0.6205	0.2351	0.1702

Table 5: The comparison of UIQM in UIEBD data set

Image	Method	UICM	UISM	UIConM	UIQM
Average	Original drawing	7.2655	0.516	0.251	0.4132
	UIBLA	4.369	4.4272	1.0566	1.3387
	GDCP	7.507	6.0695	1.1158	1.5369
	Retinex_based	8.2891	6.0948	1.1251	1.6125
	Two_step	5.7536	5.1056	1.2808	1.544
	CycleGAN	6.7429	5.5463	1.1875	1.557
	Pix2pix	6.3899	4.5348	1.2101	1.5407
	FUnIE-GAN	5.7272	5.1279	1.2554	1.5815
	GAN-RS	5.7091	3.8277	1.1637	1.3973
Ours	4.9033	5.6331	1.3577	1.7407	

III. C. 3) Edge detection evaluation

In order to evaluate the recovery performance of each method, we use the algorithm of this paper for edge detection. For some low-quality images due to blurring, a reference-free evaluation is used in order to assess their visual recovery. The initial purpose of the evaluation is to determine whether the algorithm can recover some lost edges. First of all, we need to calculate three values to measure the number of new visible edges, the proportion of new visible edges is calculated as follows:

$$e = \frac{n_r - n_0}{n_0} \quad (27)$$

n_r represents the number of visible edges in the enhanced image I_r and n_0 represents the number of visible edges in the original image I_0 . e represents the proportion of new visible edges in the enhanced image I_r . I.e:

$$\bar{r} = \exp \left[\frac{1}{n} \sum_{P_i \in \varphi_i} \log r_i \right] \quad (28)$$

The \bar{r} descriptor considers both visible and invisible edges in the original image I_0 , which denotes the quality of contrast restoration, and the r parameter comes from the fact that φ denotes the number of visible sets of edges in the I_r graph. Namely:

$$\sigma = \frac{n_s}{\dim_x \times \dim_y} \quad (29)$$

 Table 6: The visibility recovery descriptor e and the average of \bar{r}

Method	$e(a)$	$\bar{r}(a)$	$e(c)$	$\bar{r}(c)$	$e(e)$	$\bar{r}(e)$	$e(ave)$	$\bar{r}(ave)$
UIBLA	0.0011	1.0065	0.3168	1.5553	0.1393	1.1697	0.3612	1.5414
GDCP	0.3885	1.5084	0.5193	2.2007	0.3512	1.6766	0.4852	1.9552
Retinex_based	0.1484	1.449	0.6949	3.1322	0.9337	1.9449	1.0045	2.5812
Two_step	0.277	1.759	0.6226	1.9397	0.6131	1.5211	0.6537	2.0158
Cycle_GAN	0.1512	1.5008	0.8168	2.8801	1.0943	2.0288	0.6499	1.9682
Pix2pix	0.4088	2.6283	1.0885	4.0786	1.2435	3.0053	1.4687	2.7387
FUnIE-GAN	0.0394	1.2676	0.4305	1.571	0.3089	1.3117	0.1896	1.4048
GAN-RS	0.327	3.8568	0.9293	4.7008	1.2485	4.0373	0.8479	3.3924
Ours	0.4572	4.1472	0.1797	5.5535	1.2999	3.71	1.4384	3.6015

Finally, after using the contrast recovery method, we calculate the number of saturated black and white pixels and then normalize the size of the image, where \dim_x and \dim_y denote the width and height of the image, respectively.

The color image (a, c, e) is the generated augmented image, which is not compared in this paper due to the presence of many 0 values in σ . A comparison of the mean values of the visibility restoration descriptors e and \bar{r} on the UIEBD dataset is shown in Table 6. Where a, c, e corresponds to the enhanced images generated from

the color images, and the two rightmost columns indicate the average values for the whole dataset. We can see that the \bar{r} value of this paper's method is the highest among all the compared methods (3.6015), indicating that its average recovery quality is better than the other methods. For the e value, pix2pix obtains a higher score (1.4687), but otherwise, this paper's method outperforms all other methods. Thus, our method is overall superior in quality recovery of underwater images.

IV. Conclusion

Aiming at the problems of image texture blurring and serious color bias, this paper proposes an image enhancement method that fuses deep learning with multi-scale oriented filtering Retinex, and the performance of this paper's method is tested using underwater image datasets.

(1) The conclusion of aesthetic information measurement has correlation with the conclusion of human aesthetic judgment, therefore, when integrating the aesthetic information measurement of color and texture, the entropy value and gradient average can be used as a comprehensive index of aesthetic judgment and be used to evaluate the aesthetic value. The entropy value of the image after processing by the method of this paper can reflect the difference of image contrast, and the entropy value of the image in this matter increases from 6.5847 to 7.6014.

(2) The measurement of aesthetic information based on image gradient found that the average value of gradient of normal exposure image is lower than the average value of gradient of overexposed image, which shows that there is no absolute relationship between normal exposure and image gradient without reference.

(3) The method in this paper shows superior performance in several aspects such as chromaticity, sharpness, average saturation and contrast, in addition, edge detection is also performed, which further demonstrates the effectiveness of this paper's method in the field of underwater image enhancement.

References

- [1] Li, Z., Liu, F., Yang, W., Peng, S., & Zhou, J. (2021). A survey of convolutional neural networks: analysis, applications, and prospects. *IEEE transactions on neural networks and learning systems*, 33(12), 6999-7019.
- [2] Wu, J. (2017). Introduction to convolutional neural networks. National Key Lab for Novel Software Technology. Nanjing University. China, 5(23), 495.
- [3] He, Y., He, N., Zhang, R., Yan, K., & Yu, H. (2022). Multi-scale feature balance enhancement network for pedestrian detection. *Multimedia Systems*, 28(3), 1135-1145.
- [4] Xia, Y., Yun, H., & Liu, Y. (2023). MFEFNet: Multi-scale feature enhancement and Fusion Network for polyp segmentation. *Computers in Biology and Medicine*, 157, 106735.
- [5] 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.
- [6] Wang, Z., Guo, J., Zhang, C., & Wang, B. (2022). Multiscale feature enhancement network for salient object detection in optical remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 60, 1-19.
- [7] Zheng, H., Chen, J., Chen, L., Li, Y., & Yan, Z. (2020). Feature enhancement for multi-scale object detection. *Neural Processing Letters*, 51, 1907-1919.
- [8] He, K., Qin, Y., Gou, F., & Wu, J. (2023). A novel medical decision-making system based on multi-scale feature enhancement for small samples. *Mathematics*, 11(9), 2116.
- [9] Li, S., Wang, R., Zhou, F., Wang, Y., & Guo, N. (2023). Multi-Scale Feature Enhancement for Saliency Object Detection Algorithm. *IEEE Access*, 11, 103511-103520.
- [10] Li, M., Liu, W., Shao, C., Qin, B., Tian, A., & Yu, H. (2025). Multi-Scale Feature Enhancement Method for Underwater Object Detection. *Symmetry*, 17(1), 63.
- [11] Du, S., Yang, T., Teng, F., Zhang, J., Li, T., & Zheng, Y. (2024). Multi-scale feature enhanced spatio-temporal learning for traffic flow forecasting. *Knowledge-Based Systems*, 294, 111787.
- [12] Tianhao, J. I. A., Li, P. E. N. G., & Feifei, D. A. I. (2023). Object detector with residual learning and multi-scale feature enhancement. *Journal of Frontiers of Computer Science & Technology*, 17(5), 1102.
- [13] Dong, B., Wang, R., Yang, J., & Xue, L. (2021). Multi-scale feature self-enhancement network for few-shot learning. *Multimedia Tools and Applications*, 80(25), 33865-33883.
- [14] Cui, H., Li, J., Hua, Z., & Fan, L. (2022). Attention-guided multi-scale feature fusion network for low-light image enhancement. *Frontiers in neurorobotics*, 16, 837208.
- [15] Xu, X., Wang, Q., Guo, L., Zhang, J., & Ding, S. (2023). FEMRNet: Feature-enhanced multi-scale residual network for image denoising. *Applied Intelligence*, 53(21), 26027-26049.
- [16] Zhang, X., Li, J., & Hua, Z. (2022). MFFE: multi-scale feature fusion enhanced net for image dehazing. *Signal Processing: Image Communication*, 105, 116719.
- [17] Xiangyu Li, Jie Chen, Jianwei Li, Zhentao Yu & Yaxun Zhang. (2025). Adaptive Matching of High-Frequency Infrared Sea Surface Images Using a Phase-Consistency Model. *Sensors*, 25(5), 1607-1607.
- [18] Qing Qi, Jichang Guo & Chongyi Li. (2025). Data-driven gradient priors integrated into blind image deblurring. *Signal Processing: Image Communication*, 135, 117275-117275.
- [19] Yang Yang, Jun Guo, Cheng Liu, Hui Qian & Pinglin Gu. (2025). Brake disc defect detection method based on an improved multi-scale template matching algorithm. *Signal, Image and Video Processing*, 19(6), 484-484.
- [20] Guanping Dong, Rui You, Sai Liu, Nanshou Wu, Xiangyu Kong, Pingnan Huang... & Zixi Wang. (2025). Mosaic ceramic surface defect detection enhancement algorithm based on guided filter and multi-scale fusion. *Materials Testing*, 67(3), 568-578.



- [21] Xin Jin, Dongming Zhou, Shaowen Yao, Rencan Nie, Chuanbo Yu & Tingting Ding. (2016). Remote sensing image fusion method in CIE Lab color space using nonsubsampling shearlet transform and pulse coupled neural networks. *Journal of Applied Remote Sensing*, 10(2), 025023-025023.