

# Research on Image Defogging Algorithm based on FPGA

Zhibo Chen, Shuai Guo, Guangshao Zhou, Xiaodong Zhang

Southwest University for Nationalities, Chengdu Sichuan, 610041, China

**Abstract:** In recent years, the rapid development of China's industry has brought about environmental pollution problems, especially the frequent occurrence of haze weather across the country, which has had a significant impact on the quality of image shooting. In haze weather, the scattering effect of the atmosphere seriously hinders the visibility of images and the transmission of detailed information. In order to solve this problem, image defogging technology came into being, which plays an important role in the field of image enhancement and restoration. However, as the demand for image clarity increases, software algorithms alone are no longer sufficient to meet current needs. Therefore, image defogging methods that combine hardware and software have become the focus of research. This study uses Xilinx 's Zynq-7020 series FPGA development board, uses Verilog language for program design, and uses Vivado and ModelSim software for program setting and simulation verification. Experimental results show that the image dehazing effect can be effectively improved by combining software and hardware.

**Keywords:** FPGA; Image Dehazing; Dark Channel Prior; Image Processing.

## 1. Preface

Fog is a common bad weather in autumn and winter, which will seriously reduce the visual quality of images. It not only affects human visual perception, but also has adverse effects on some advanced visual tasks such as target detection, distance measurement, image classification, video surveillance, etc., and seriously damages the performance of outdoor visual systems. Fog and haze usually appear together. Fog usually refers to mist, which means small water droplets suspended in the air, which reduces visibility on the ground, but the concentration of fog will gradually decrease with the appearance of the sun; haze is an aerosol system formed by the aggregation of small particles in the air, and it will not dissipate with the appearance of sunlight. Under haze conditions, due to the large amount of particles trapped in the air that will affect the light, the images taken outdoors are usually affected by the weather, which will affect the acquisition and analysis of normal image information, and then lead to inaccurate feature extraction of the image. In addition, haze days will also bring inconvenience to people's travel, and affect road pedestrian detection, road sign recognition, and vehicle automatic driving.

Haze reduces visibility, making the pictures we get unclear and the details in the pictures seriously affected. Therefore, it is very important to restore the clarity of the images obtained from haze. Most defogging algorithms are implemented on personal computers using programming languages such as C++ and Matlab. These methods have the advantage of low cost in defogging operations. However, during the processing, since the CPU runs the program in a serial manner, the defogging efficiency is affected and the execution speed cannot meet the requirements for high-speed computing in image and video processing. Limited by the clock frequency, this type of processor cannot meet the standard of fast image processing.

## 2. Related Theories and Methods

Images obtained from haze days are affected by atmospheric light value and projection rate. Image defogging methods can be divided into two types: image enhancement

and image degradation. The most commonly used methods for the former are based on wavelet transformation, partial differential equations, Retinex, etc. This method mainly changes the clarity of the image by adjusting the contrast and brightness of the image. In order to meet the relevant requirements, this method usually requires a combination of multiple image processing algorithms. The most commonly used methods for the latter are defogging based on atmospheric scattering models, neural network defogging, dark channel prior defogging, etc. This method is to establish a corresponding mathematical model based on the degree of image degradation, so as to restore the detailed information of the image and extract the required feature information from the image, so that the image affected by haze is restored. Image enhancement technology does not consider the specific process of image degradation, but restores a clear original image by using a variety of image enhancement methods. Image restoration technology requires a deep understanding of the image degradation process, and then simulates the inverse process of the image degradation process, and obtains a clear defogging image by implementing this inverse process on the image. Although the ultimate goal of both methods is to restore the original image and improve its quality, the method adopted in this paper is mainly based on the simulation of the image degradation process.

### 2.1. Atmospheric Scattering Model

The atmospheric scattering model mainly involves the scattering and absorption effects of light when it propagates in the atmosphere. These effects will cause the energy attenuation and direction change of light. In order to describe these processes, scholars have proposed multiple mathematical models. The mathematical expression of the most commonly used atmospheric scattering model is as follows:

$$I(X) = J(X) \bullet t(X) + A[1 - t(X)] \quad (1)$$

Among them,  $I(X)$  is the observed fog image,  $J(X)$  is the fog-free image to be restored,  $A$  represents the atmospheric illumination value,  $t(X)$  and is the transmittance, which describes the part of light that is not

scattered during the process from the scene to the camera. It is usually related to the depth of field and the scattering coefficient. The mathematical expression of the transmittance is as follows:

$$t(X) = e^{-\beta d(X)} \quad (2)$$

Among them,  $\beta$  represents the scattering coefficient of the medium to light of different wavelengths (atmospheric scattering coefficient);  $d(X)$  represents the distance the light propagates. This formula reflects the process in which the light intensity gradually attenuates as the propagation distance increases.

According to formula (1), the parameters  $I(X)$  are known. To obtain the parameters  $J(X)$ , we need to calculate  $A$  the values of the parameters  $I(X)$  and  $t(X)$ . When the parameters are known  $t(X)$ , in order to obtain the desired fog-free image, we need to estimate the parameters  $A$  and in the atmospheric scattering model  $t(X)$ . After the estimation is completed, these parameter values are substituted into the model for calculation to obtain a clear fog-free image. In this process, the estimation of  $A$  and  $t(X)$  needs to rely on the relevant principles of the dark channel. This model shows that the observed image is composed of two parts: the attenuated reflected light of the target and the scattered ambient light.

## 2.2. Basic Knowledge of Dark Channels

Channel Prior (DCP) is an algorithm for image dehazing, which was proposed by He Kaiming et al. in the paper Single Image Haze Removal Dark Channel Prior in 2009. The core idea of the algorithm is to use the principle of atmospheric scattering to restore a haze-free image by estimating the transmittance of the scene.

### 2.2.1. Dark Channel Estimation

The dark channel we often refer to is that in a fog-free and clear image, except for the sky area or the pure white area, there must be a color with the lowest value among the three primary colors R, G, and B. Based on this principle, the model is proposed, that is, for a fog-free and clear image, its dark channel is defined as:

$$J^{\text{dark}} \rightarrow 0 \quad (3)$$

the pixel values of the three color channels R, G, and B in an image are all high, the image will appear white. In the image, except for those areas with high values of the three color channels and the sky, other areas can generate a dark channel image by selecting the lowest channel value among the three R, G, and B and combining them in a specific way. These values are usually determined by the atmospheric light intensity, so they can be used to make a more accurate estimate of the atmospheric transmittance. The transmittance estimated by this method can help us restore clear and fog-free images.

The formula for defining the dark channel is as follows:

$$J^{\text{dark}}(X) = \min_{y \in \Omega(X)} \left( \min_{c \in \{r, g, b\}} J^c(y) \right) \quad (4)$$

Among them,  $J^c$  represents the three-color channels

$\Omega(X)$  of R, G, and B, and represents a local window centered on X. Calculating the dark channel of the image is equivalent to calculating the minimum value of the RGB channels of all pixels in the local window.

### 2.2.2. Transmittance Calculation

The dark channel algorithm is used to estimate the transmittance image, whose transmittance represents the visibility of objects in the scene. The calculation formula is as follows:

$$\hat{t}(X) = 1 - \min_{y \in \Omega(X)} \left( \min_c \frac{I^c(y)}{A^c} \right) \quad (5)$$

In the actual operation process, a certain degree of fog is generally retained for the image that needs to be defogged, so that the defogging effect can be obtained. Therefore, we will add a parameter to formula (5)  $\omega$ , as shown in formula (6):

$$\hat{t}(X) = 1 - \omega \times \min_{y \in \Omega(X)} \left( \min_c \frac{I^c(y)}{A^c} \right) \quad (6)$$

Among them,  $\hat{t}(X)$  represents the transmittance,  $\omega$  ( $0 < \omega \leq 1$ ) represents a constant, usually 0.95,  $\Omega(X)$  is a local window centered on x,  $A^c$  represents the ambient atmospheric light intensity, and  $I^c$  represents the observed light intensity. A represents the ambient light intensity value, and the value of A can be obtained with the help of the dark channel map. First, select the pixels with the top 0.1% brightness from the dark channel map, and then find the values of the points corresponding to these highest brightness points from the foggy image as the value of A. When the transmittance  $\hat{t}(X)$  is very small, some parts of the dehazed image will be overexposed. Usually, a threshold is set to limit it. The commonly used threshold is 0.1, which is recorded as  $\hat{t}_0$ .

### 2.2.3. Dehazing Mapping and Image Restoration

After obtaining the transmittance map and the estimated value of the atmospheric light intensity, image restoration can be performed. The final defogging image is obtained by using the defogging mapping formula. The defogging mapping formula is as follows:

$$J(X) = \frac{(I(X) - A)}{\max(t(X), t_0)} + A \quad (7)$$

Among them,  $J(X)$  is the defogged image,  $I(X)$  is the original foggy image,  $t(X)$  is the transmittance, A is the atmospheric light intensity value, and X is the pixel position.

## 3. Algorithm Optimization

### 3.1. Algorithm Improvement Idea

When performing defogging operations on foggy images according to the traditional dark channel prior defogging algorithm proposed by He Kaiming et al.,  $\omega$  the value of each image is unchanged by default. However, for any foggy image, the density of fog in different areas is different. If the same  $\omega$  value is used to process the entire foggy image, the

processed image will be severely distorted. Therefore, in order to obtain a fog-free image that is closest to the actual situation after defogging, it is necessary to select the most suitable  $w$  value for calculation in different areas of the foggy image.

### 3.2. Dehazing Process

According to the above description, the original traditional dark channel prior defogging algorithm was modified and implemented through FPGA. The main idea is to first crop the foggy image; then perform traditional dark channel prior defogging on the cropped image; then perform RGB to HSL color processing on the obtained defogging image, adjust L (brightness) to change the brightness of the image color; finally, convert the adjusted defogging image back to RGB color to obtain the final defogging image. The modified algorithm flow chart is as follows:

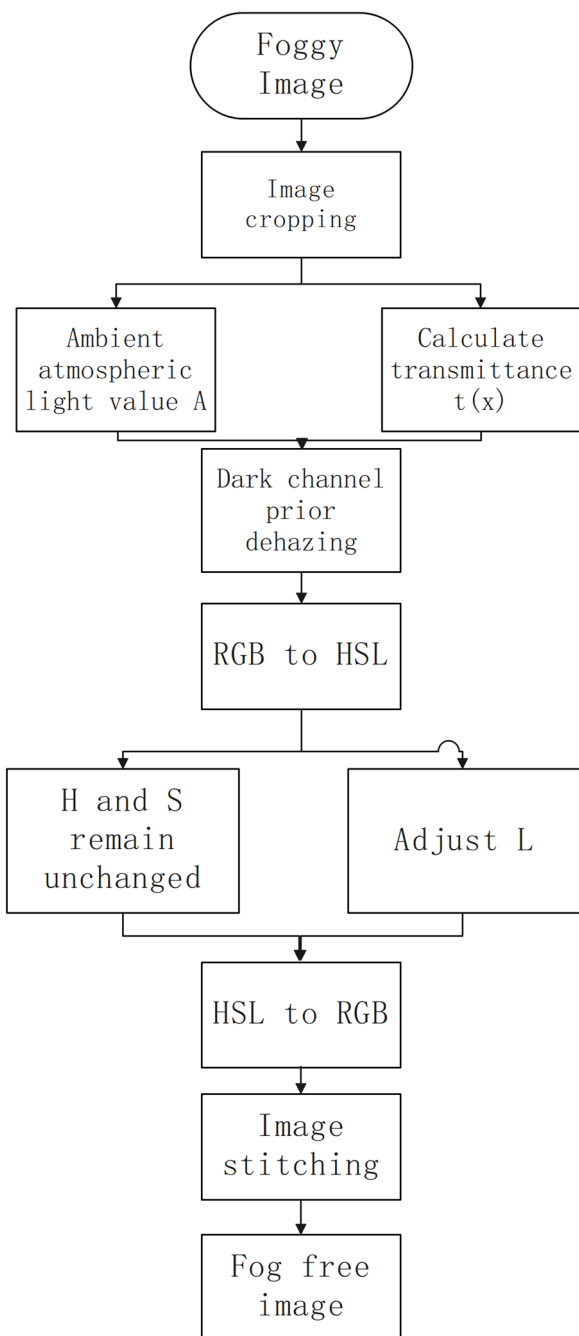


Figure 1. Modified algorithm flow chart

## 4. FPGA Implementation

### 4.1. Image Cropping

Image cropping is a commonly used technique in image processing. Image cropping can crop the part we need from the original image. Here we can use image cropping technology to divide the foggy image into several parts according to the concentration of fog. Each part is assigned a different  $w$  value according to the concentration of fog.

When using FPGA for image cropping, we need to calculate the coordinates of each pixel point, and intercept the specified rectangular range image according to the starting point, height and width of the image to be cropped. Rectangular window is the most commonly used method for image cropping.

The simulation image of image cropping using FPGA is as follows:

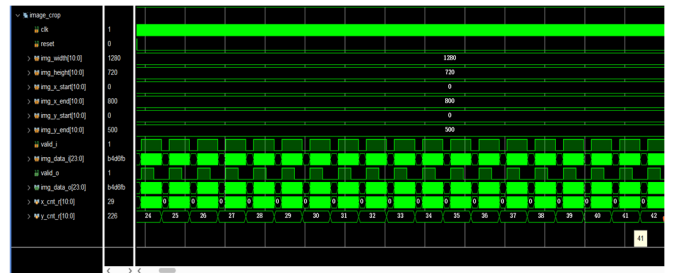


Figure 2. Simulation of image cropping

The cropped foggy image is as follows:



Figure 3. Cropped foggy image

### 4.2. Dark Channel Prior Dehazing Algorithm (Revised)

The system design flow chart of the traditional dark channel prior dehazing algorithm based on FPGA is as follows:

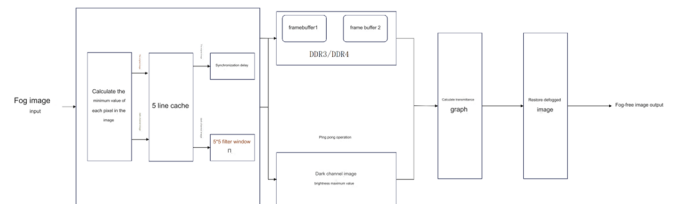


Figure 4. Flowchart of dark channel prior dehazing algorithm

First, the minimum value of the three pigment values of R, G, and B in each pixel must be calculated; second, a  $5 \times 5$  filter window is used to select the minimum brightness in the window. Here, the original image and the image result calculated in the previous step need to be cached, so a 5-row

line cache memory module is required; the maximum brightness statistics of the frame cache and dark channel image are performed; finally, the transmittance image is calculated and the defogging image is restored.

The simulation diagram of traditional dark channel defogging using FPGA is as follows:

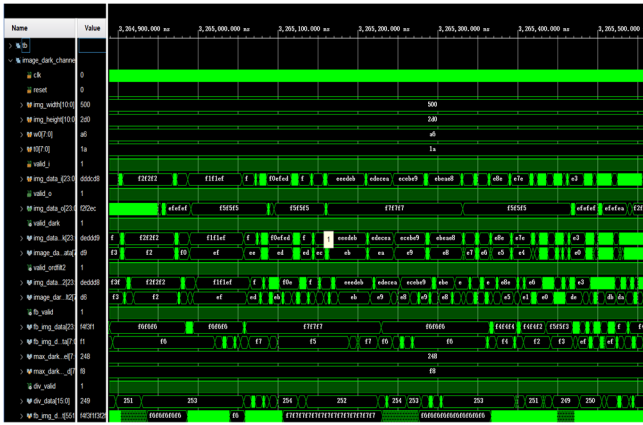


Figure 5. Simulation of traditional dark channel dehazing

The fog-free image produced by the traditional dark channel dehazing algorithm is as follows:

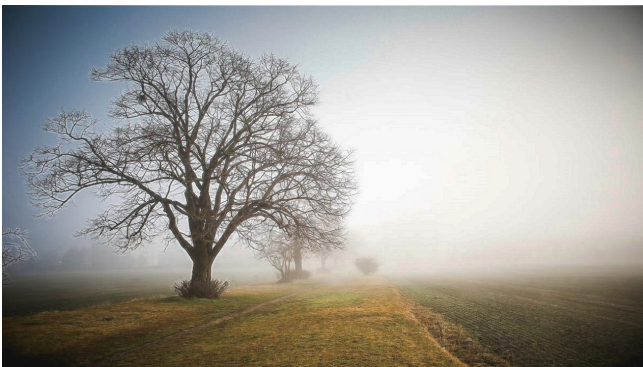


Figure 6. Traditional dark channel dehazing effect diagram

### 4.3. Conversion between RGB and HSL

#### 4.3.1. HSL Color Space

The HSL color space is a way of human perception of color, which decomposes color into three independent components: hue, saturation, and brightness. Among them, hue indicates the type of color, such as red, yellow, green, etc.; saturation indicates the purity of the color, that is, the proportion of gray components in the color. The higher the saturation, the purer the color; brightness indicates the brightness of the color, that is, the proportion of white components in the color. The higher the brightness, the brighter the color. The advantage of the HSL color space is that it can intuitively adjust the various attributes of the color, making color selection and editing simpler and more intuitive.

#### 4.3.2. Advantages of HSL over RGB

The advantage of HSL color space over RGB color space is that it is more in line with human perception of color and human intuition, making the color selection and adjustment process simpler, more direct and more efficient. HSL color space can express the properties of color more intuitively and has a very broad application prospect in image design, color adjustment and image editing.

#### 4.3.3. RGB to HSL Conversion

The formula for converting RGB to HSL has the following

steps:

1. Convert the RGB value to a value between 0 and 1. This can be done by dividing the RGB value (which ranges from 0 to 255) by 255.

2. Find the maximum (Cmax) and minimum (Cmin) values in RGB.

3. Calculate the hue (H). The formula for calculating the hue depends on the color channel corresponding to the maximum value. If the maximum value is R, then  $H = (G' - B') / (Cmax - Cmin)$ ; if the maximum value is G, then  $H = 2 + (B' - R') / (Cmax - Cmin)$ ; if the maximum value is B, then  $H = 4 + (R' - G') / (Cmax - Cmin)$ . It should be noted that if Cmax is equal to Cmin, the hue is 0. In addition, the H value obtained by the above calculation needs to be between 0 and 6, and then it is multiplied by 60 and the remainder is taken to obtain a hue value between 0 and 360.

4. Calculate the saturation (S). Saturation indicates the purity of the color and is calculated as:  $S = (Cmax - Cmin) / Cmax$ . If Cmax is 0, then saturation is also 0.

5. Calculate the lightness (L). Lightness indicates how bright the color is and is calculated as:  $L = (Cmax + Cmin) / 2$ .

HSL to RGB is the reverse process of RGB to HSL, and the steps are similar.

#### 4.3.4. FPGA Implementation and L (Brightness) Adjustment

The simulation diagram of using FPGA to convert RGB to HSL is as follows:

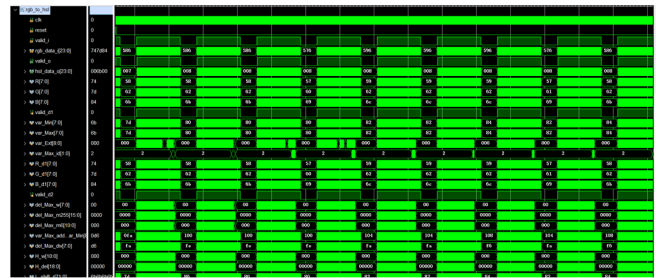


Figure 7. Simulation diagram of RGB to HSL conversion

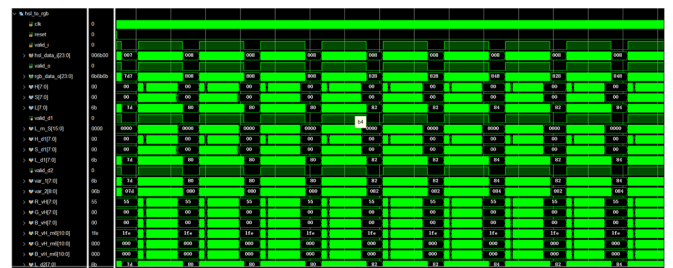


Figure 8. Simulation diagram of HSL to RGB conversion

After applying the dark channel prior dehazing technique to the foggy image, we first convert the image obtained based on the RGB color mode to the HSL color model. In the HSL color space, H stands for hue, S stands for saturation, and L stands for luminance. On the basis of ensuring that other components remain unchanged, we specifically perform adaptive histogram equalization processing to limit contrast for luminance (L). Then, the processed image is converted back to the RGB color space.

### 4.4. Image Stitching

Image stitching refers to the technology of stitching multiple images into one image, which has a wide range of applications in the field of image processing. The cropped

images can be stitched into a complete image.

The system framework of FPGA-based image stitching is as follows:

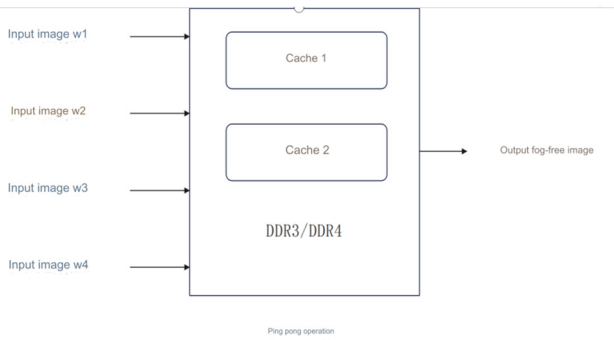


Figure 9. System framework of image stitching

As can be seen from the above figure, multiple images are input into DDR3/DDR4 for storage, cached according to the specified area, and all cached images are output when the image data is output, and uninterrupted data output is achieved through ping-pong operation.

The simulation diagram of image stitching using FPGA is as follows:

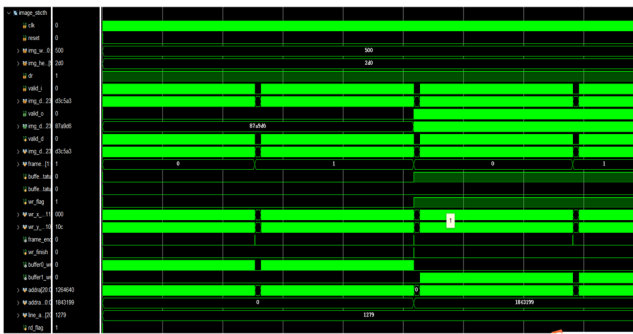


Figure 10. Simulation of image stitching

## 5. Result Analysis

According to the shortcomings of the traditional dark channel defogging, the dark channel defogging algorithm was improved to improve the defogging ability. The parallel computing capability of FPGA was used to further improve the speed and accuracy of the calculation. The comparison between the improved algorithm and the traditional algorithm is as follows:



Figure 11. Original image



Figure 12. Traditional dark channel dehazing image

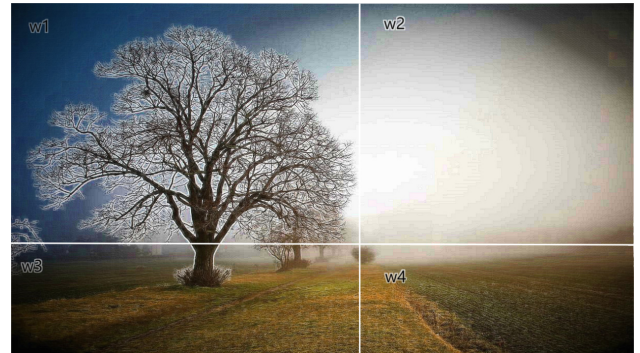


Figure 13. Dehazed image after improved algorithm

## References

- [1] Yang Yong, Qiu Genying, Huang Shuying, et al. Single image dehazing method based on improved atmospheric scattering model [J]. Journal of Beijing University of Aeronautics and Astronautics, 2021, 48(8): 1364-1375.
- [2] Ouyang Nan, Cao Yiping. White blood cell image segmentation method based on color calibration and HSI color space [J]. OPTICS & OPTOELECTRONIC TECHNOLOGY, 2022, 20(3):62-69.
- [3] Wang Lijuan, Chang Xia, Ren Wang. Color image enhancement simulation based on weighted histogram equalization [J]. Computer Simulation, 2021, 38(12): 126-131.
- [4] Huang Jinwei, Yu Luo, Guo Tianyuan, et al. Improved image dehazing algorithm based on dark channel prior[J]. Journal of Jiamusi University (Natural Science Edition), 2023, 41(1): 25-28, 67.
- [5] Zhang Hangying, Wang Xueqi, Wang Huaying, et al. Improved Retinex - Net image enhancement method based on brightness component [J]. Chinese Journal of Physics, 2022, 71(11): 107-115.
- [6] Zhang Xue, Wang Feng, Zhao Jia. Improved single image dehazing algorithm based on dark channel prior [J]. Journal of Fuyang Normal University (Natural Science Edition), 2022, 39(02): 69-75.
- [7] Zhang Hong, Zhang Yulun, Deng Xu, et al. Image dehazing algorithm based on improved dark channel prior [J]. Computer Simulation, 2022, 39(04): 150-155.
- [8] Li Shaoting. Image dehazing algorithm and its implementation based on Zynq [D]. Inner Mongolia University, 2021.
- [9] Cao Yahui. Research and implementation of real-time video defogging system based on FPGA [D]. Nanjing University of Posts and Telecommunications, 2015.
- [10] Fan Di, Guo Xinyun, Lu Xiao, Liu Xiaoxin, Sun Bo. Image Defogging Algorithm Based on Sparse Representation[J]. Complexity, 2020, 2020.

- [11] Jian Peng Chen, Du Yan Bi, Chang Liu, Zhun Lin Fan. Novel Single Image Dehazing Algorithm[J]. Advanced Materials Research, 2014,3326(989-994).
- [12] Xie Shangyou, Zhang Jun. Image dehazing algorithm based on color correction and channel compensation prior[P]. Jiangnan University (China), 2022.
- [13] Wei Linxiao. Research on image dehazing algorithm and FPGA implementation [D]. Xidian University, 2019.
- [14] Zhou Xiaochao. Design of real-time image defogging system based on FPGA [D]. North University of China, 2021.
- [15] He Zhaolan. Design of real-time defogging system based on FPGA [D]. Harbin University of Science and Technology, 2022.