Project 2-1 Histogram

1、Objective

Histogram method usually increases the global contrast of many images, especially when the usable data of the image is represented by close contrast values. Through this adjustment, the intensities can be better distributed on the histogram. This allows for areas of lower local contrast to gain a higher contrast. Histogram equalization accomplishes this by effectively spreading out the most frequent intensity values. The method is useful in images with backgrounds and foregrounds that are both bright or both dark, and the method can lead to better detail in photographs that are over or under-exposed. A key advantage of the method is that it is a fairly straightforward technique and an invertible operator.

In order to make the dark part of the image visible, there remain three methods of histogram we used in this project, including histogram equalization (HE), adaptive histogram equalization (AHE) and contrast limited adaptive histogram equalization (CLAHE). There are two picture to verify the method if it works well.

Figure 1 Astronaut

Figure 2 Rock

In two pictures, some dark parts can't see it clearly. So in this project, histogram method will make them visible.

2、Method and Tricks

2.1 HE

HE is a method in image processing of contrast adjustment using the image's histogram. Histogram Equalization, also known as histogram flattening, essentially stretches the image nonlinearly and redistributes the image pixel values so that the number of pixel values in a certain grayscale range is approximately equal. Thus, the contrast of the peak portion in the middle of the original histogram is enhanced, and the contrast of the valley portion on both sides is lowered, and the histogram of the output image is a flat segmented histogram: if the output data segmentation value is small, it will be generated that a rough classification of visual effects will result.

Figure 3 shows the histogram equalization, which is to modify the randomly distributed image histogram into a uniformly distributed histogram. The basic idea is to do some mapping transformation on the pixel gray of the original image, so that the transformed image is gray.

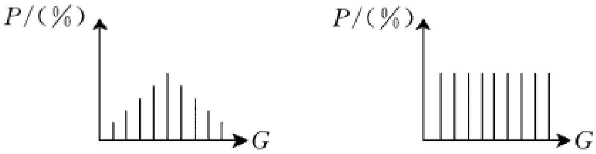

Figure 3 Histogram Equalization

Through this technique, the brightness of the image can be clearly seen on the histogram, and the brightness of the image can be adjusted as needed.

When using HE method, first, we should calculate the number of every intensity, and save in an array. In this array, the first number of the array holds the number of pixels with a value of 0 and so on. Second, we should convert the number into probability of intensity. Final, we redistribute each pixel value based on probability according to Formula 1.

$$s = (L-1)\sum_{j=0}^{k} p_r(r_j) = (L-1)\sum_{j=0}^{k} \frac{n_j}{MN} = \frac{L-1}{MN}\sum_{j=0}^{k} n_j \qquad \text{Formula 1}$$

In Formula 1, s is new pixel value, M, N is height and weight of image, (L-1) is the maximum value that the pixel can reach, n_j is the number of j-th pixel.

2.2 AHE

AHE is a computer image processing technique used to improve contrast in images. It differs from ordinary HE in the respect that the adaptive method computes several histograms, each corresponding to a distinct section of the image, and uses them to redistribute the lightness values of the image. It is therefore suitable for improving the local contrast and enhancing the definitions of edges in each region of an image. However, AHE has a tendency to overamplify noise in relatively homogeneous regions of an image.

AHE performs histogram enhancement on each pixel by calculating a transformation function for each pixel neighborhood. The simplest form is to equalize each pixel based on the histogram of the square neighborhood of the pixel. As shown in the following figure, the idea of enhancing based on the histogram is exactly the same as the ordinary histogram enhancement because of this transformation. The function is commensurate with the cumulative distribution function (CDF) of the pixel neighborhood. But in order to speed up the calculation, we adopt Faster AHE. Using the window of W * W to calculate the CDF of the histogram, and then not only transforming one pixel of the image, but transforming a series of pixels.

The boundary pixels of the image should be specially processed because the neighborhood of these pixels is not completely contained in the image, such as the blue box in the above figure, which can be solved by mirroring the pixel rows or columns; copying the boundary pixel rows directly It is not advisable because it causes a high peak in the pixel neighborhood histogram.

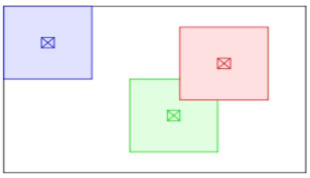

Figure 4 Affect a pixel value

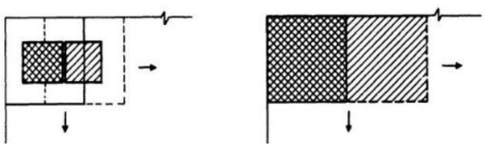

Figure 5 Transforme a series of pixels.

When using AHE method, we should pay attention to the boundary condition. For example, when the window is in the upper left corner of the image, the affected area is not (A, A), but the whole window (W,W). The same applies to the upper right, lower right, lower left and surrounding. So when I was using this method, first, I am calculating the circle around the image. And on the boundary, I'm going to do a mirror padding. So when I'm dealing with the middle part, there's not a lot to think about.

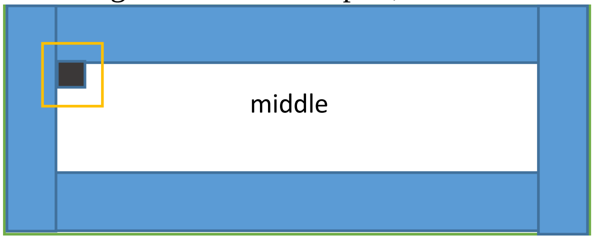

Figure 6 Blue part is boundary condition with W windows. White part is middle part, and black part is stride and affect area. A square with yellow lines is W windows.

2.3 CLAHE

Ordinary AHE tends to overamplify the contrast in near-constant regions of the image, since the histogram in such regions is highly concentrated. As a result, AHE may cause noise to be amplified in near-constant regions. CLAHE is a variant of adaptive histogram equalization in which the contrast amplification is limited, so as to reduce this problem of noise amplification. In CLAHE, the contrast amplification in the vicinity of a given pixel value is given by the slope of the transformation function. This is proportional to the slope of the neighborhood CDF and therefore to the value of the histogram at that pixel value. CLAHE limits the amplification by clipping the histogram at a predefined value before computing the CDF. This limits the slope of the CDF and therefore of the transformation function. The value at which the histogram is clipped, the so-called clip limit, depends on the normalization of the histogram and thereby on the size of the neighborhood region.

In general, it is not good to simply ignore the parts that exceed the clipping limit of the histogram. Instead, distribute the cropped parts evenly to other parts of the histogram. This redistribution process may cause those times that have been clipped to be re-clipped to exceed the clipping value. It can be ignored. If this is not desired, the excess can be repeated until the requirements meeting.

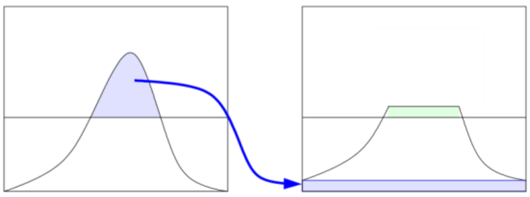

Figure 7 Redundant redistribution

In order to speed up the calculation, we use interpolation method to finish CLAHE method. Interpolation greatly improves the efficiency of the above algorithm, and does not decrease the quality. First, divide the image evenly into equal rectangular sizes. Then calculate the histogram, CDF and corresponding transformation function of each block. This transformation function is fully defined for the center pixel of the block (the small black square on the left side of the image below). The other pixels are interpolated by the transform function of the four adjacent blocks. The pixels located in the blue shaded part of the figure are bilinear lookup interpolation, while the pixels located in the convenient edge (shaded green) are linear interpolation, and the corner points (shaded red) are directly using the transformation function of the block. Such a process greatly reduces the number of times the transform function needs to be calculated, but only increases the amount of computation for bilinear interpolation.

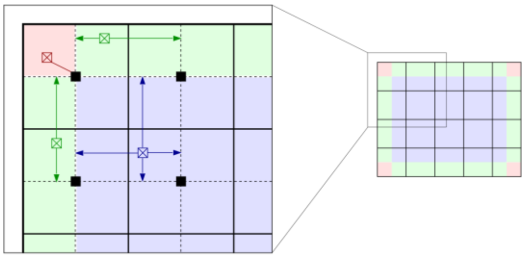

Figure 8 Illustration of tile-based interpolation for CLAHE.

When using CLAHE method, first, we should need to pad the image for better processing using mirror padding. Second, for parts with different colors, we could deal with separately. To prevent double counting, we can save the every CDF of windows in an array. For Blue part, we use Formula 2. And for Green part, we use Formula 3.

$$\mathrm{f(D)} = (1-\Delta\mathrm{y})\big((1-\Delta\mathrm{x})f_{ul}(D) + \Delta x f_{bl}(D)\big) + \Delta y\big((1-\Delta\mathrm{x})f_{ur}(D) + \Delta x f_{br}(D)\big)$$

Formula 2

$$\mathrm{f(D)} = \big((1-\Delta\mathrm{x})f_{l}(D) + \Delta x f_{r}(D)\big) \qquad \text{Formula 3}$$

For Formula 2, Δx and Δy are the ratio of the distance between the blue pixel point and the center of the upper left corner window, that is, the distance between the black pixel point in the upper left corner and the window size. For Formula 3, Δx is the ratio of the green pixel point to the window boundary distance to the window size.

2.4 RGB and HSI

$H=\begin{cases}\theta, & G\geq B\\ 2\pi-\theta, & G<B\end{cases}$, where $\theta=\cos^{-1}\left(\frac{(R-G)+(R-B)}{2\sqrt{(R-G)^2+(R-B)(G-B)}}\right)$	$S=1-\frac{3\min(R,G,B)}{R+G+B}$	$I=\frac{R+G+B}{3}$

HSI to RGB

- $0° \leq H < 120°$

$$B = I(1-S), \qquad R = I\left[1+\frac{S\cos(H)}{\cos(60°-H)}\right], \qquad G = 3I-(R+B)$$

- $120° \leq H < 240°$

$$R = I(1-S), \qquad G = I\left[1+\frac{S\cos(H-120°)}{\cos(180°-H)}\right], \qquad B = 3I-(R+G)$$

- $240° \leq H < 360°$

$$G = I(1-S), \qquad B = I\left[1+\frac{S\cos(H-240°)}{\cos(300°-H)}\right], \qquad R = 3I-(G+B)$$

When getting HSI model, we can process I using histogram method.

3、Result

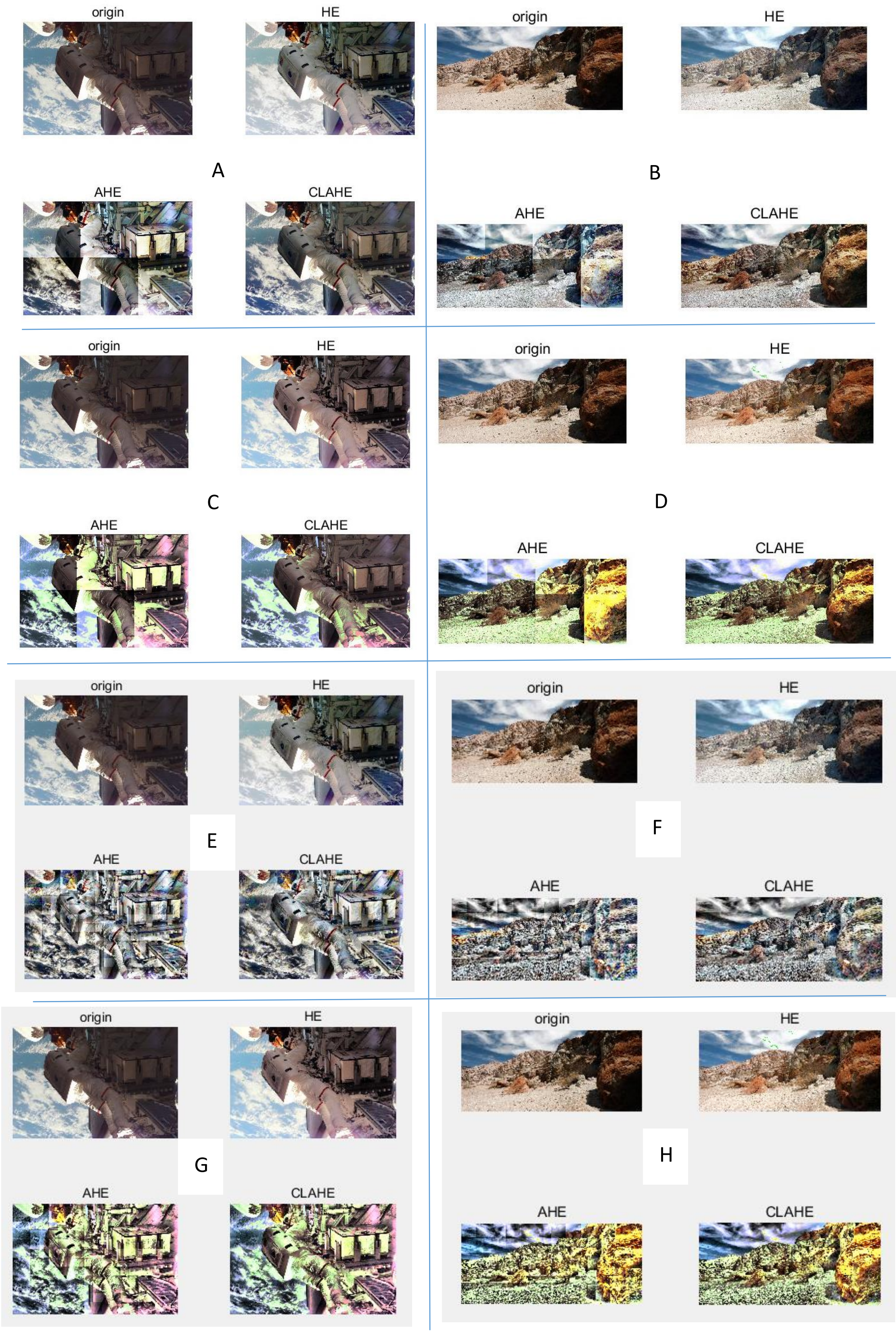

Figure 9 There are two models for testing. A,B,E,F are RGB model, and C,D,G,H are HSI model . A,B,C,D are with same window size, and E,F,G,H are with same window size.

4、Discussion

For HE, it is a histogram equalization of the image globally, so the image will generally increase the brightness. Because after the equalization, the dark places will be weakened. AHE improve the local contrast and edge of the image to get more details. However, the problem of AHE is that while enhancing the contrast, it also amplifies the noise of the image. So it looks like there are a lot of borders. When using CLAHE, because the interpolation method is used, the image will look smoother. But for HE and AHE or CLAHE, HE improves brightness in the original dark part, but weaker than AHE and CLAHE. But in vision, I think HE looks better. For RGB model and HSI model, different window sizes create different styles. In my opinion, HSI is better than RGB in certain aspects, like in vision, contrast. Maybe it has distortion.