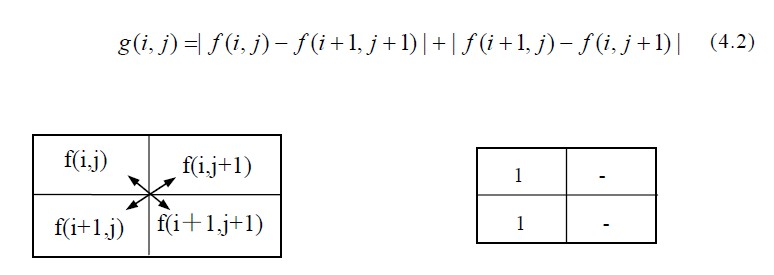
**Class Project 9**

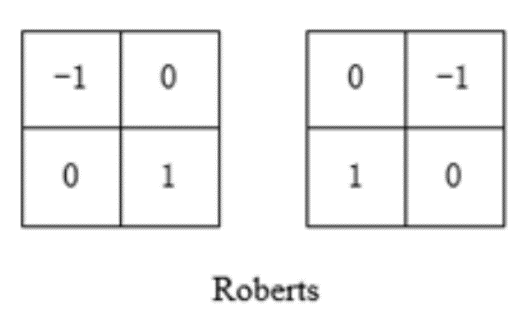
Use two images for each operation to do the following operations and write down their advantages and disadvantages and explain your results:

1. **Obtain gradient images (headCT-Vandy, building\_original, noisy\_fingerprint):**

* Roberts:

**Algorithm:**





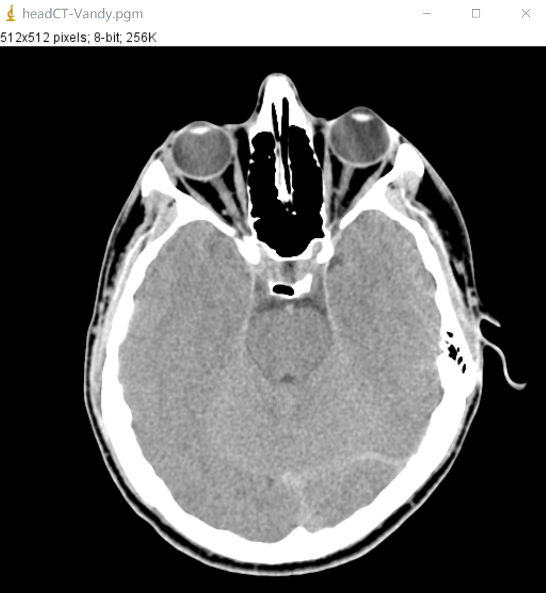
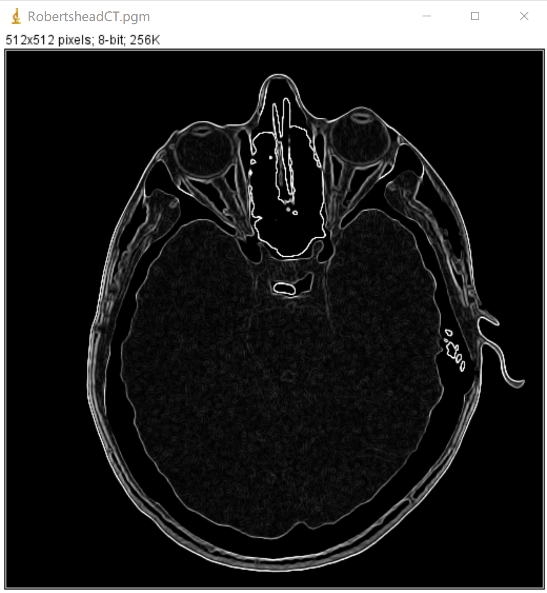
If g(i, j) is bigger than 255, then the pixel value is set to 255,

Else, the pixel value is the result.

**Results (including pictures):**

Result of processing “headCT-Vandy.pgm”:

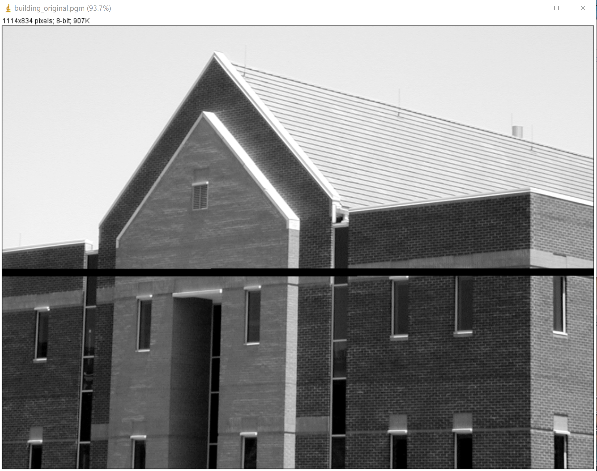
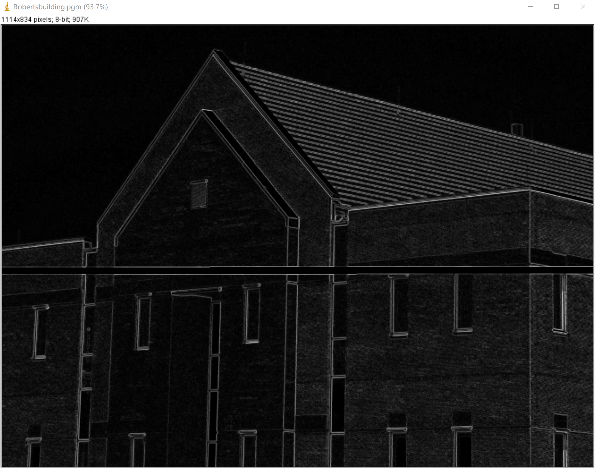
Source Image: Result after Edge detection: (Roberts)



Result of processing “building\_original”:

Source Image: Result after Edge detection:

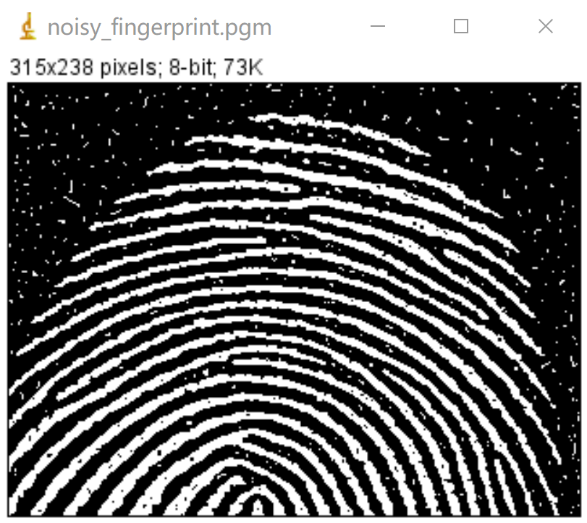
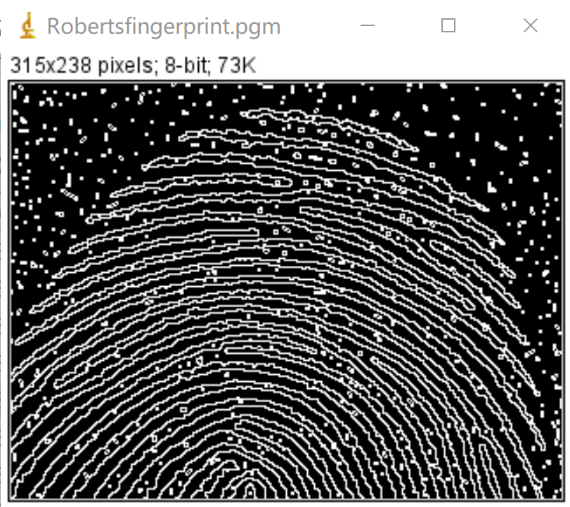
(Roberts)



Result of processing “noisy\_fingerprint”:

Source Image: Result after Edge detection:

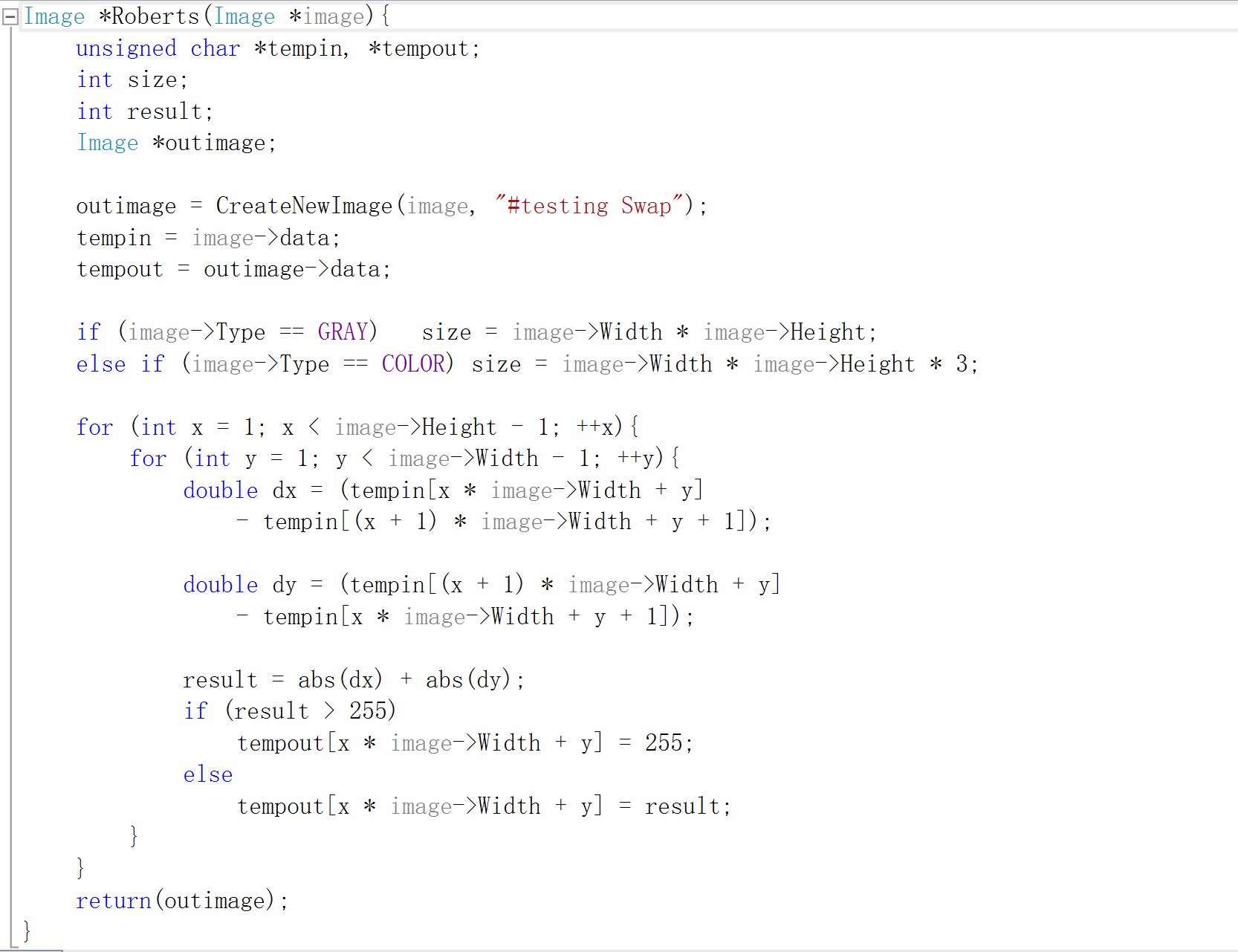
(Roberts)



**Discussion:**

The Roberts operator is also known as the cross-differential algorithm. It is a gradient algorithm based on cross-differential, which detects edge lines through local differential calculation. It is commonly used to process steep and low-noise images. When the edge of the image is close to plus or minus 45 degrees, the processing effect of this algorithm is more ideal. The disadvantage is that the positioning of the edges is not accurate, and the extracted edge lines are thicker.

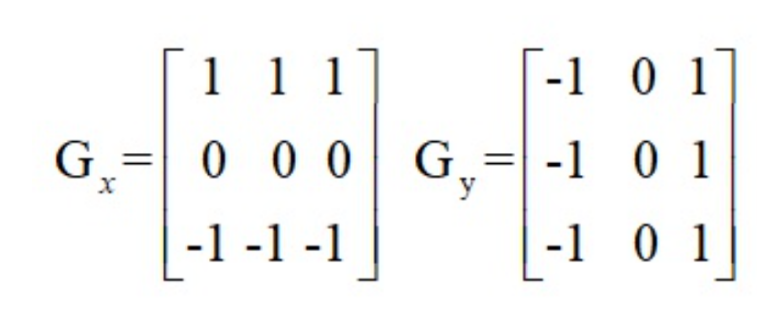
**Codes:**



* Prewitt

**Algorithm:**





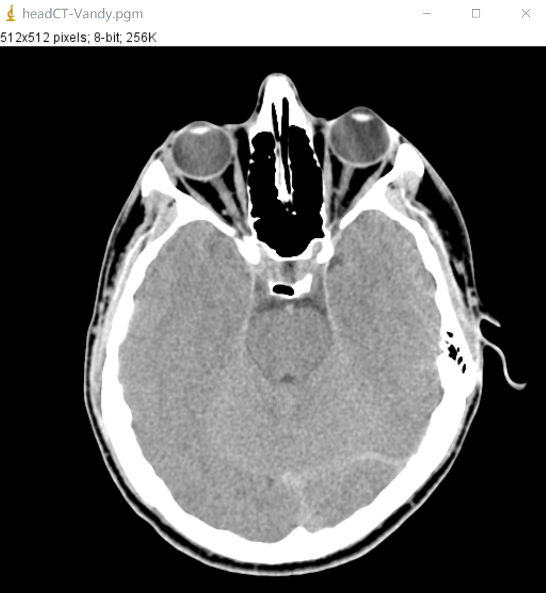
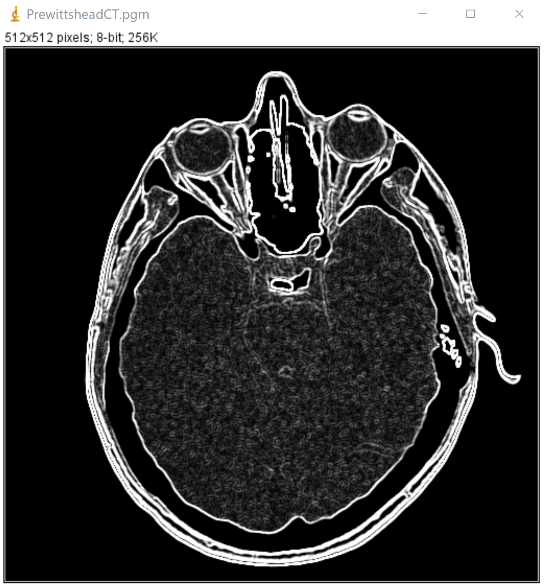
If |Gx| + |Gy| is bigger than 255, then the pixel value is set to 255,

Else, the pixel value is the result.

**Results (including pictures):**

Result of processing “headCT-Vandy.pgm”:

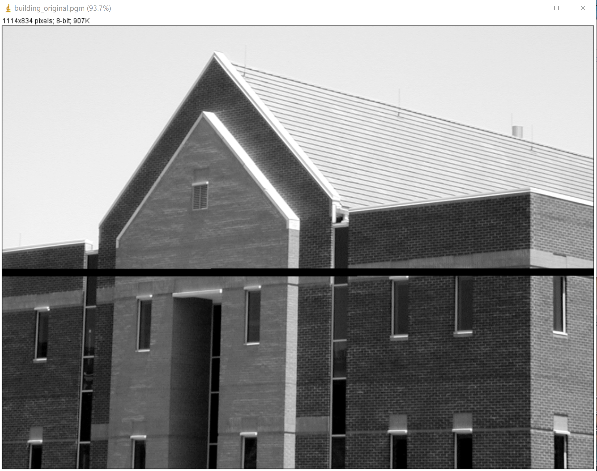
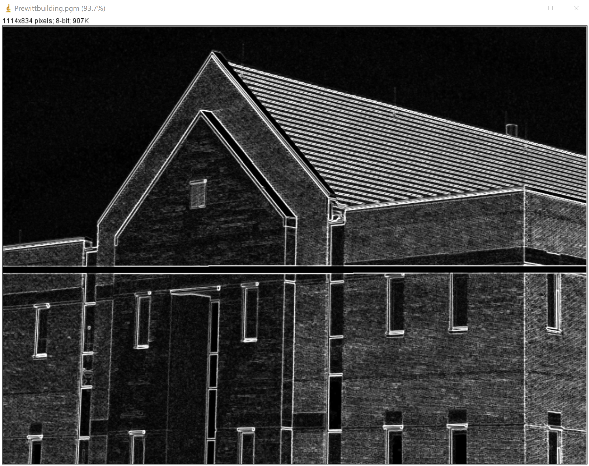
Source Image: Result after Edge detection: (Prewitt)



Result of processing “building\_original”:

Source Image: Result after Edge detection:

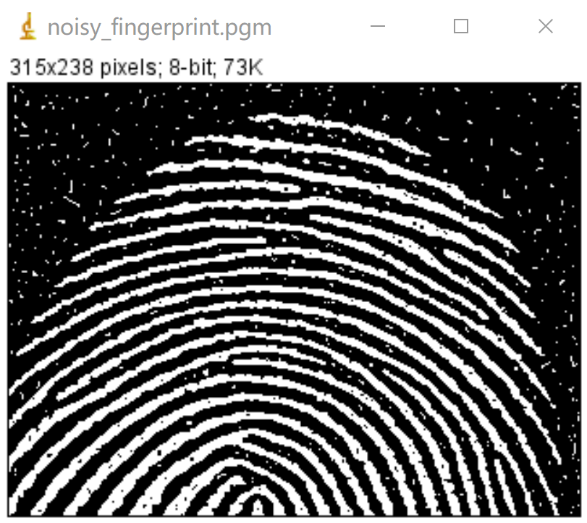
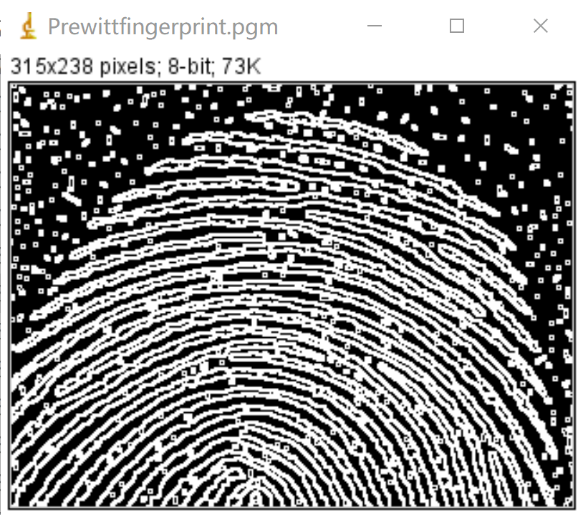
(Prewitt)



Result of processing “noisy\_fingerprint”:

Source Image: Result after Edge detection:

(Prewitt)

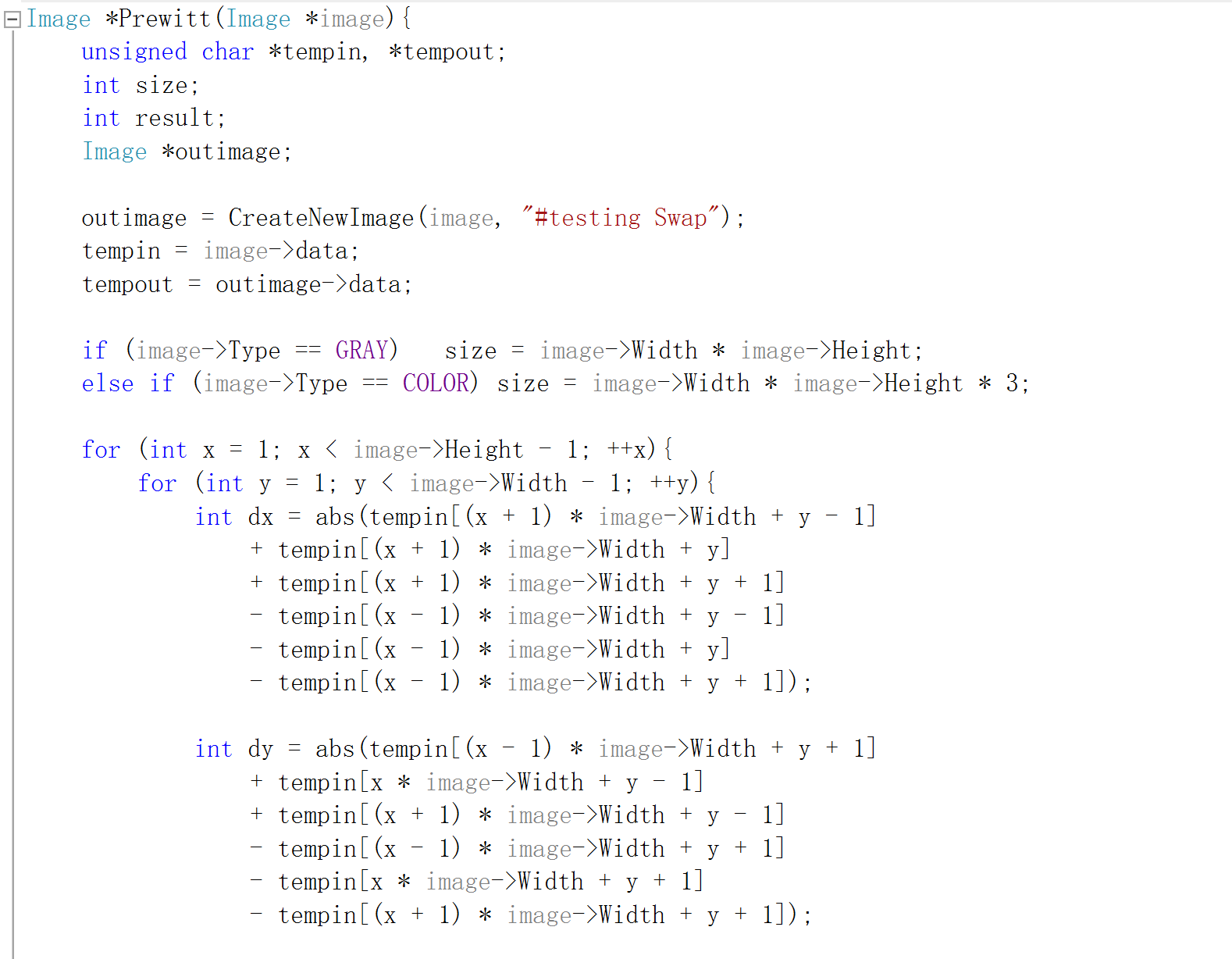
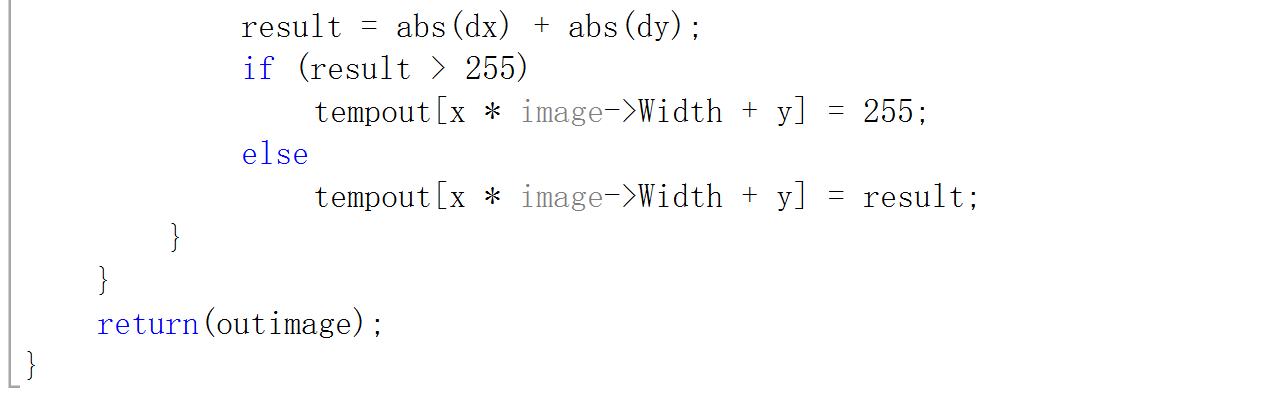


**Discussion:**

This operator is like the Sobel operator, except that the weights have changed, but there is still a gap between the functions of the two implementations. According to experience, Sobel can detect image edges more accurately than Prewitt.

The Prewitt operator differentiates in one direction and averages in one direction, so it is relatively insensitive to noise.

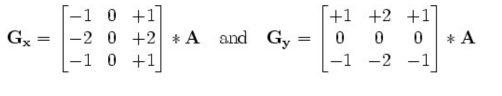
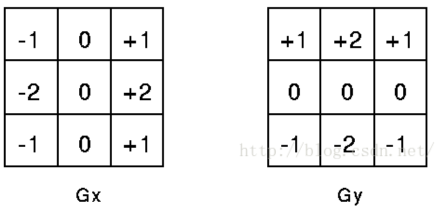
**Codes:**



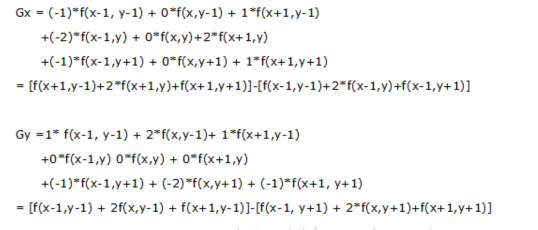
* Sobel

**Algorithm:**

Sobel Convolution factor:



The specific calculation is as follows:



Where f (a, b) represents the gray value of the image (a, b) point

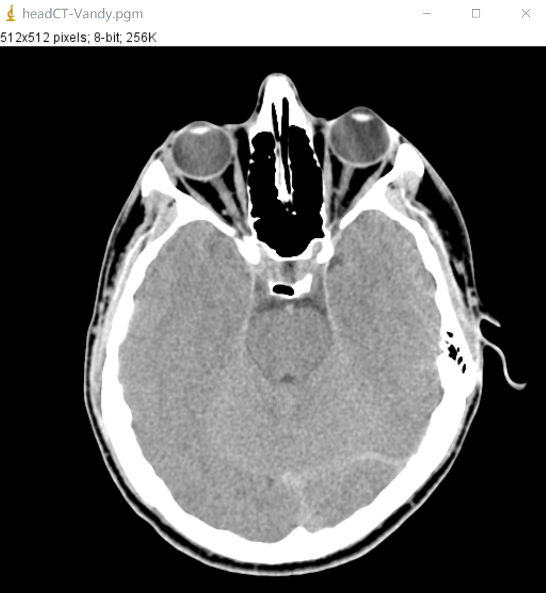
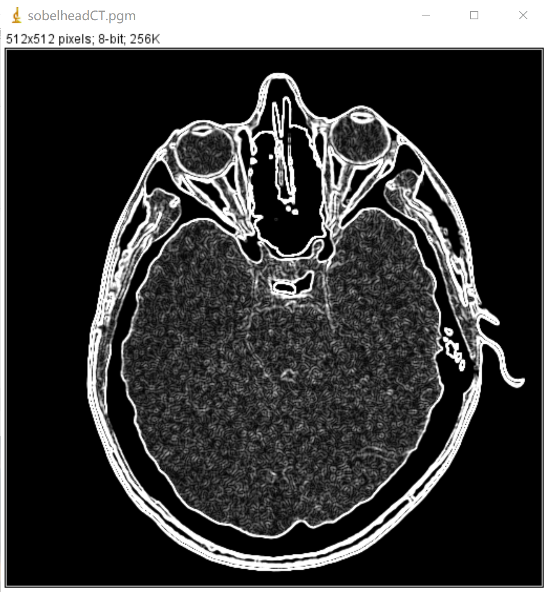
The horizontal and vertical gray values of each pixel of the image are combined by the following formula to calculate the gray scale of the point:



**Results (including pictures):**

Result of processing “headCT-Vandy.pgm”:

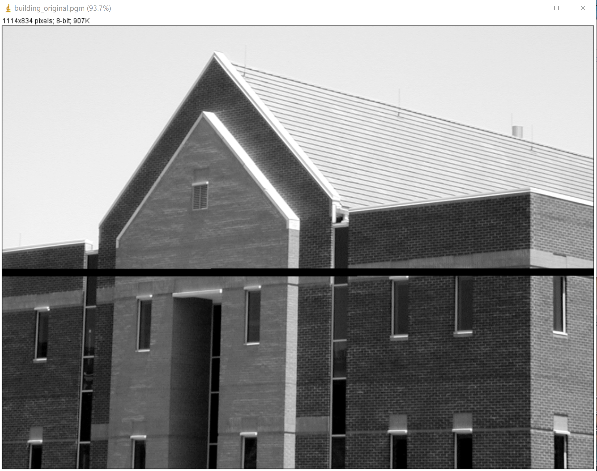
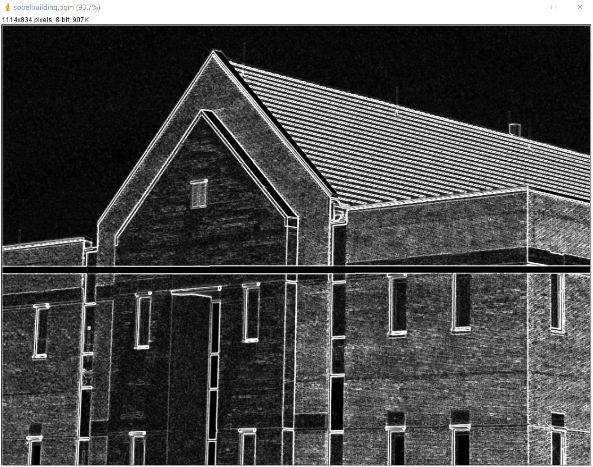
Source Image: Result after Edge detection: (Sobel)



Result of processing “building\_original”:

Source Image: Result after Edge detection:

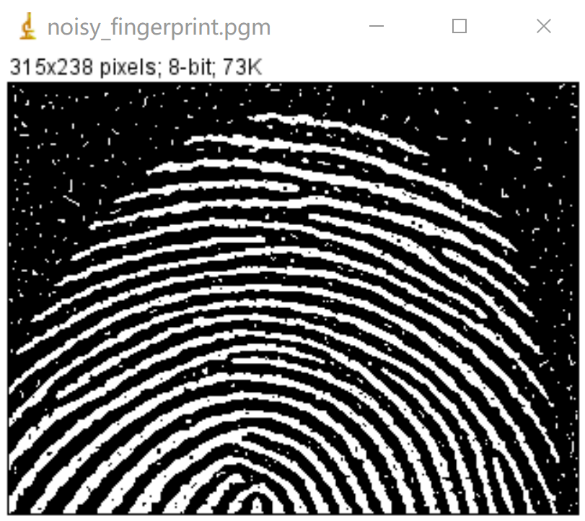
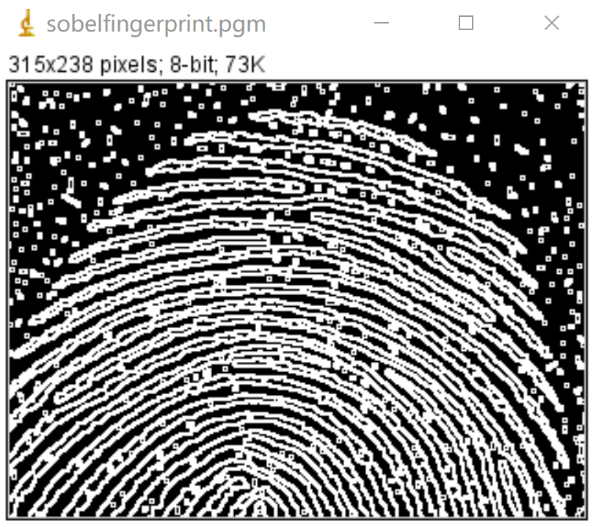
(Sobel)



Result of processing “noisy\_fingerprint”:

Source Image: Result after Edge detection:

(Sobel)



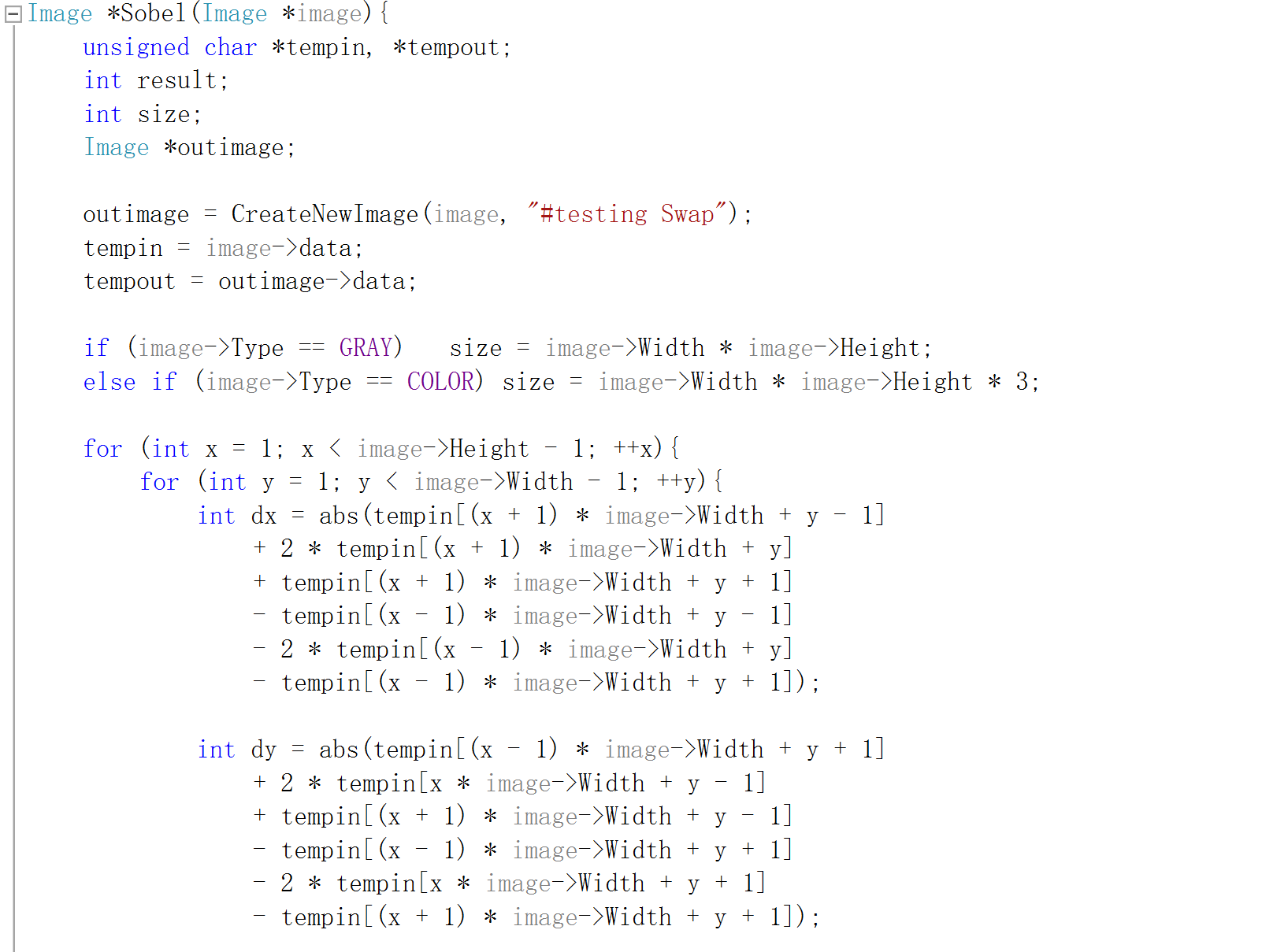
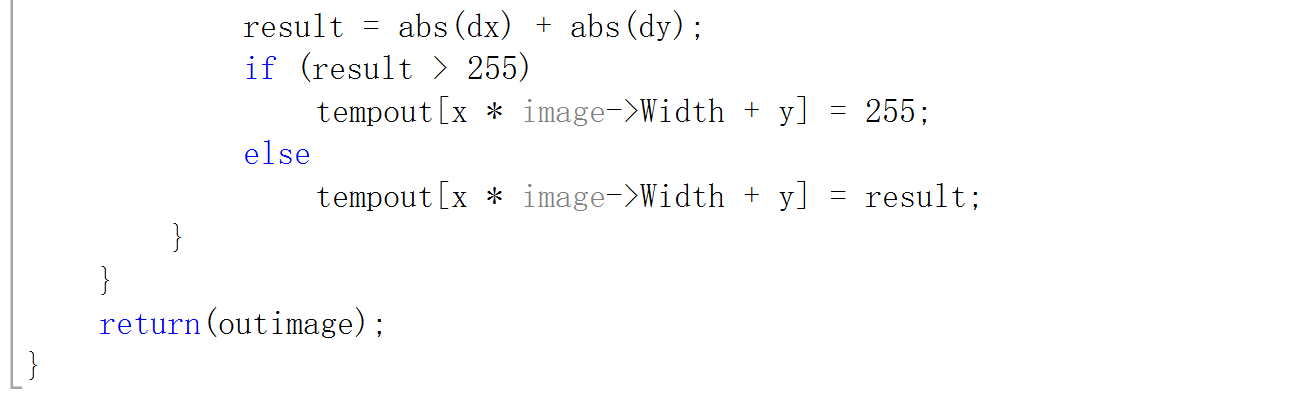
**Discussion:**

Sobel operator is a discrete differential operator for edge detection, which combines Gaussian smoothing and differential derivation. This operator is used to calculate the approximate value of the brightness of the image. According to the brightness of the edge of the image, the specific points in the area exceeding a certain number are recorded as edges. The Sobel operator adds the concept of weight based on the Prewitt operator. It is believed that the distance between adjacent points has a different impact on the current pixel. The closer the pixel is, the greater the impact of the current pixel, to realize the image Sharpen and accentuate edge contours.

The Sobel operator detects the edge according to the gray-scale weighted difference of the upper and lower, left and right adjacent points of the pixel, and reaches the extreme value at the edge. It has a smoothing effect on noise and provides more accurate edge direction information. Because the Sobel operator combines Gaussian smoothing and differential derivation (differentiation), the result will have more noise resistance. When the accuracy requirements are not very high, Sobel operator is a more commonly used edge detection method.

The edge positioning of Sobel operator is more accurate, and it is often used for images with more noise and gradual gray scale. The algorithm template is shown in the following formula, where it represents the horizontal direction and the vertical direction.

**Code:**

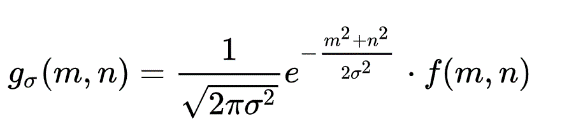
1. **Canny & LoG (headCT\_Vandy, noisy\_fingerprint):**

* Canny

**Algorithm:**

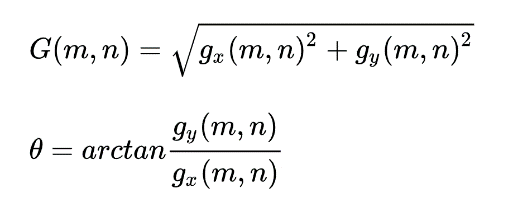
Step 1: Gaussian filtering

Using Gaussian filter to smooth the picture so that can cut the noise.

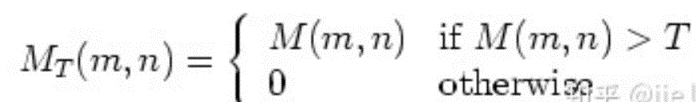


Step 2: Calculate the gradient value and direction

Optional templates: soble operator, Prewitt operator, Roberts template, etc.;



Step 3: Non-maximum suppression



Step 4: Selection of dual thresholds

Choose the coefficients TH and TL, the ratio is 2:1 or 3:1. (Generally take TH=0.3 or 0.2, TL=0.1);

Abandon the points less than the low threshold and assign 0; mark the points greater than the high threshold immediately (these points are determined edge points) and assign 1 or 255;

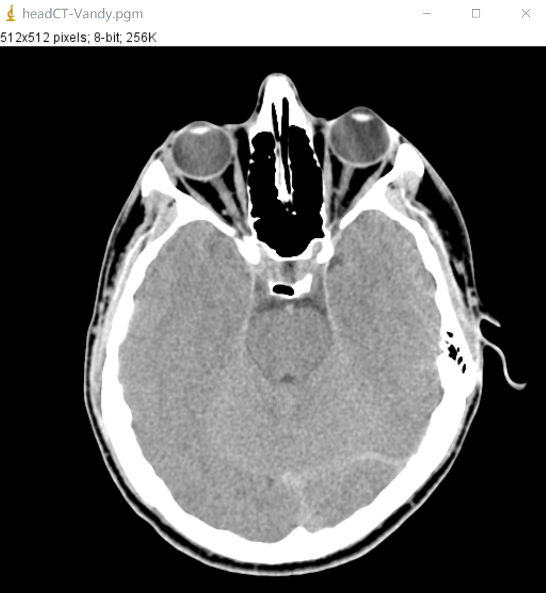
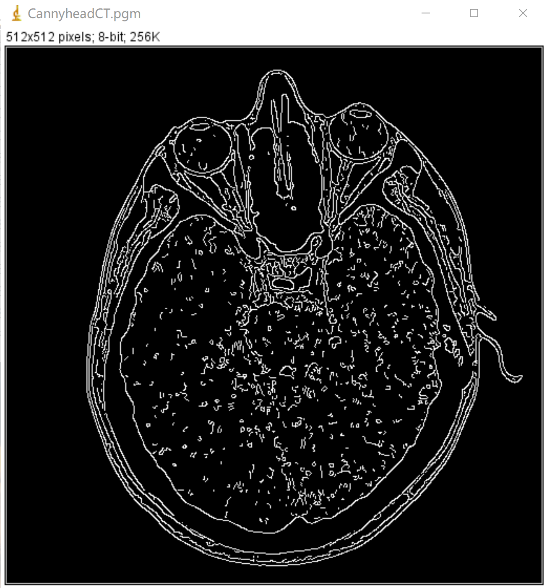
The points that are smaller than the high threshold and greater than the low threshold are determined using 8-connected regions (that is: only when connected to TH pixels will be accepted and become edge points, assigned 1 or 255)

Step 5: Edge detection

**Results (including pictures):**

Result of processing “headCT-Vandy.pgm”:

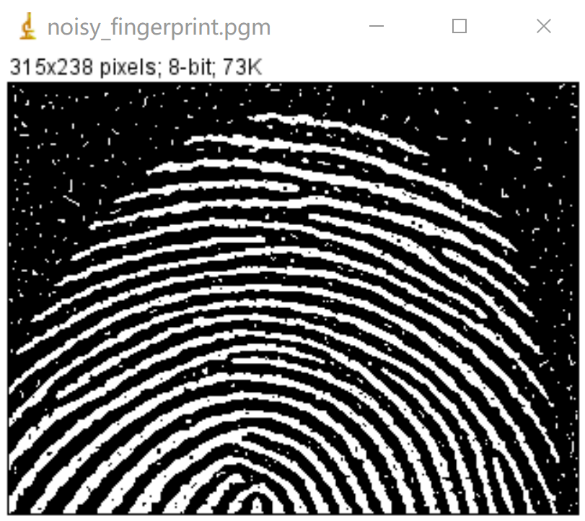
Source Image: Result after Edge detection: (Canny)



Result of processing “noisy\_fingerprint”:

Source Image: Result after Edge detection:

(Canny)



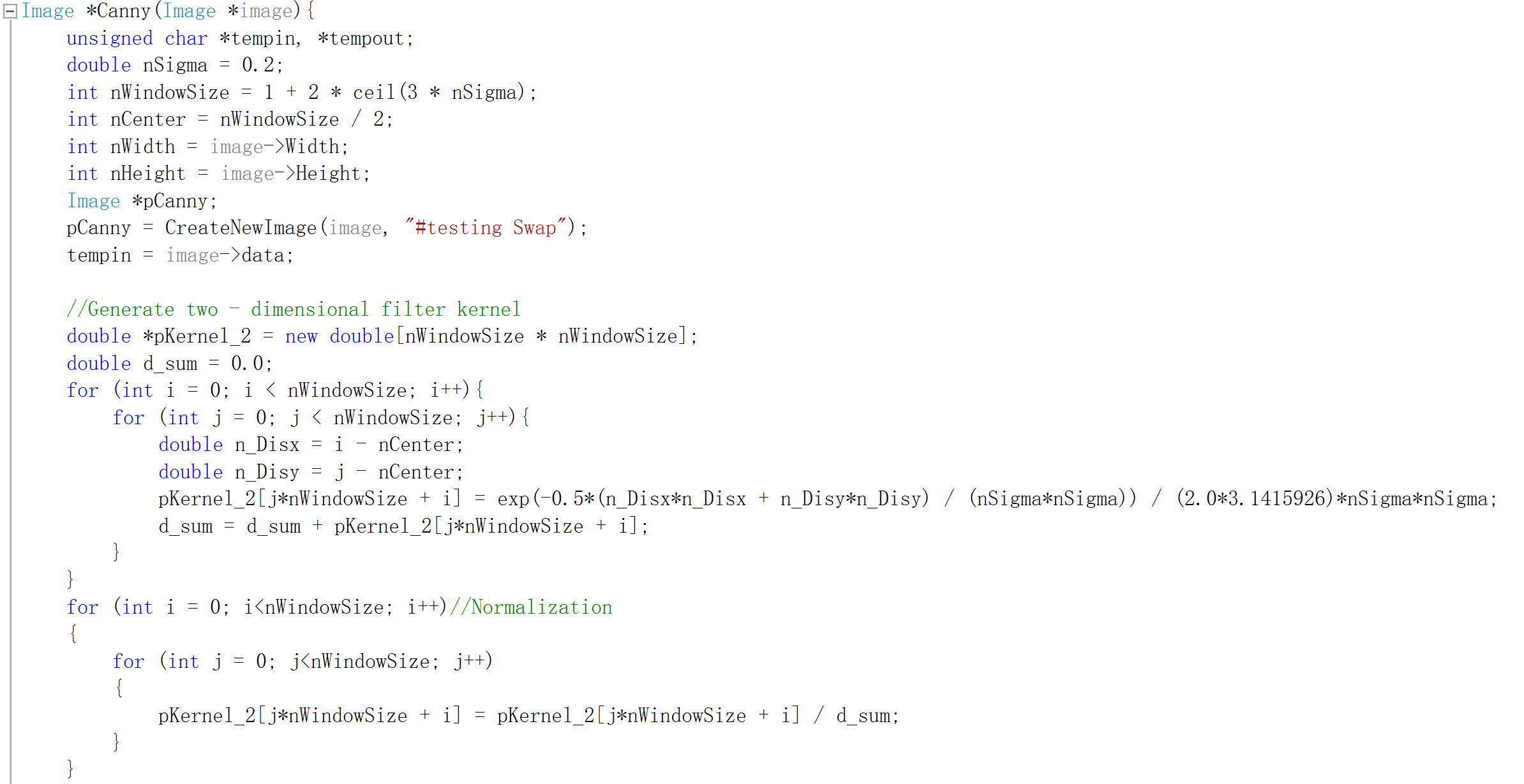
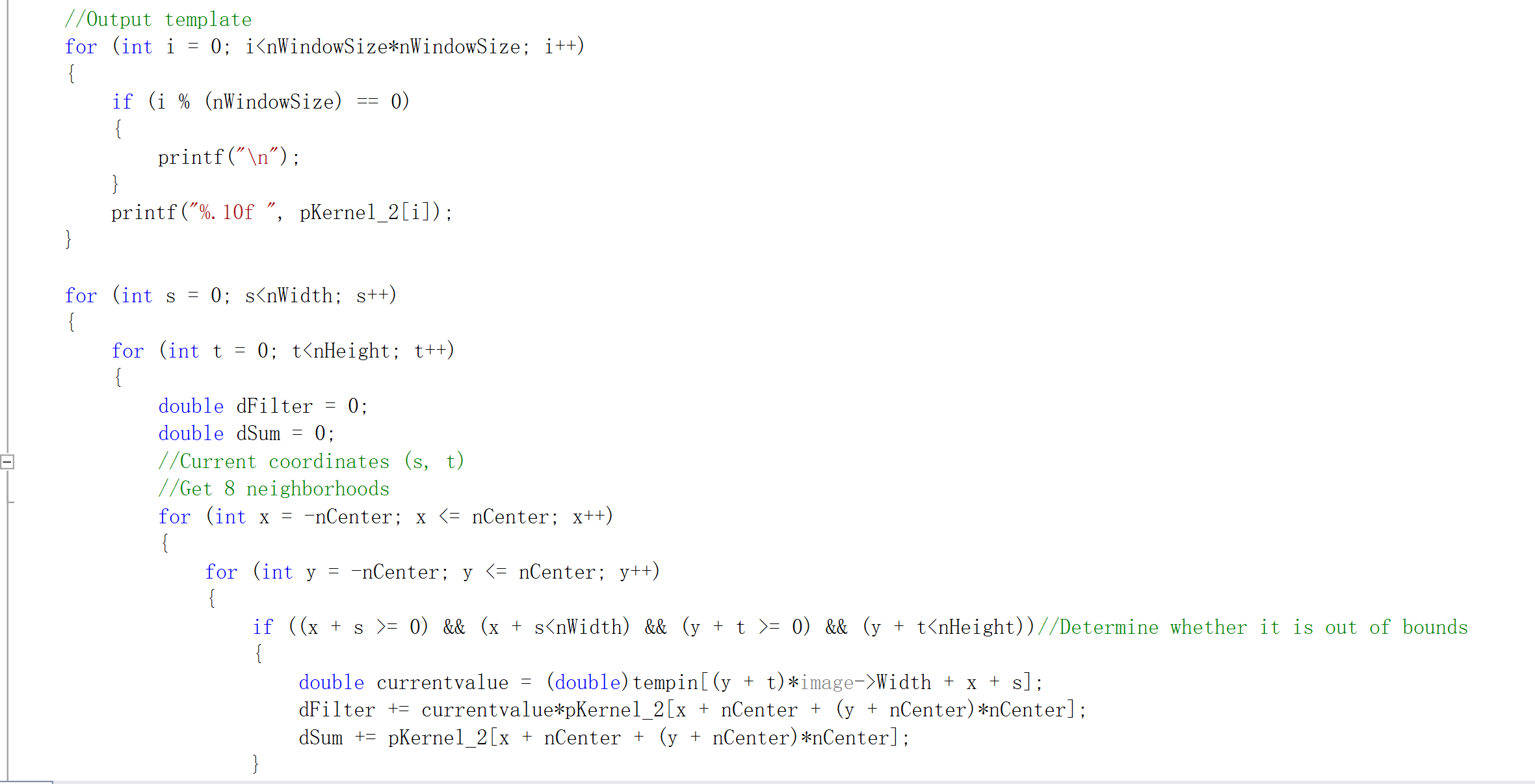
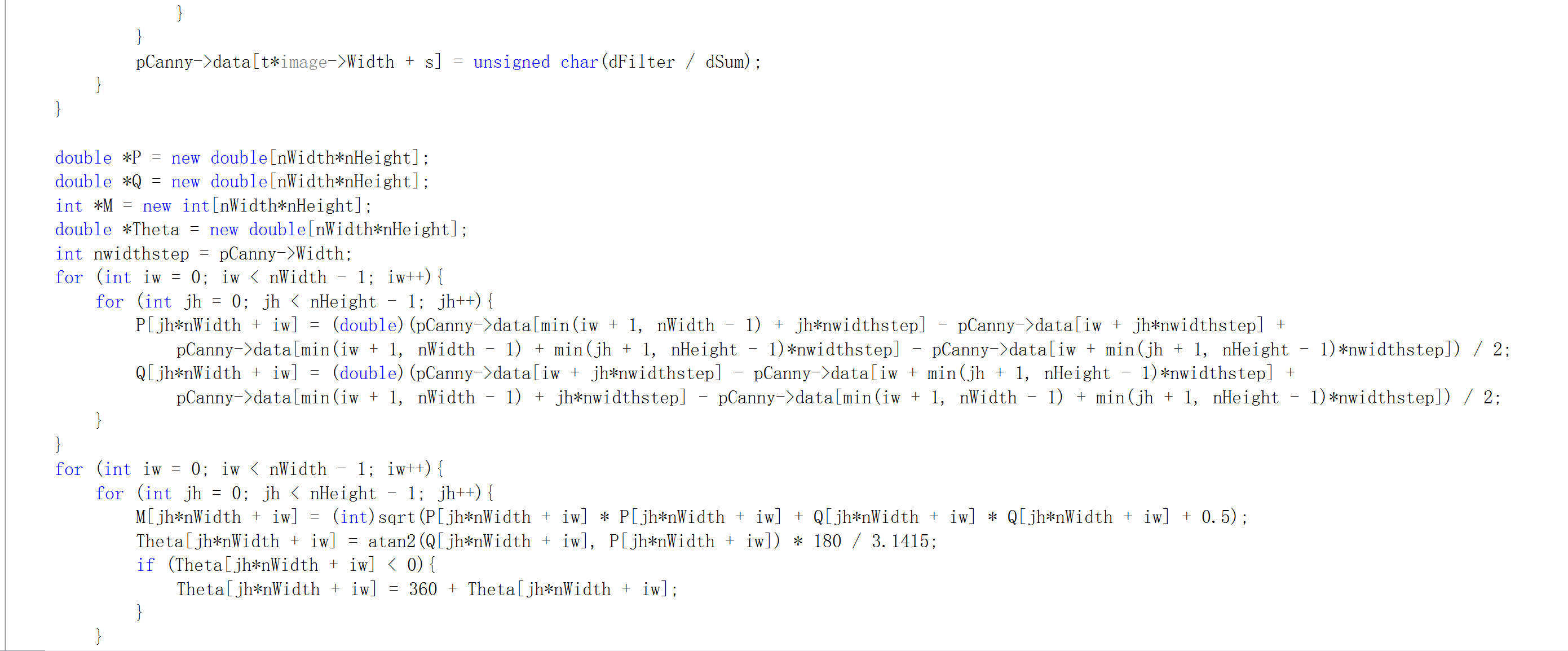
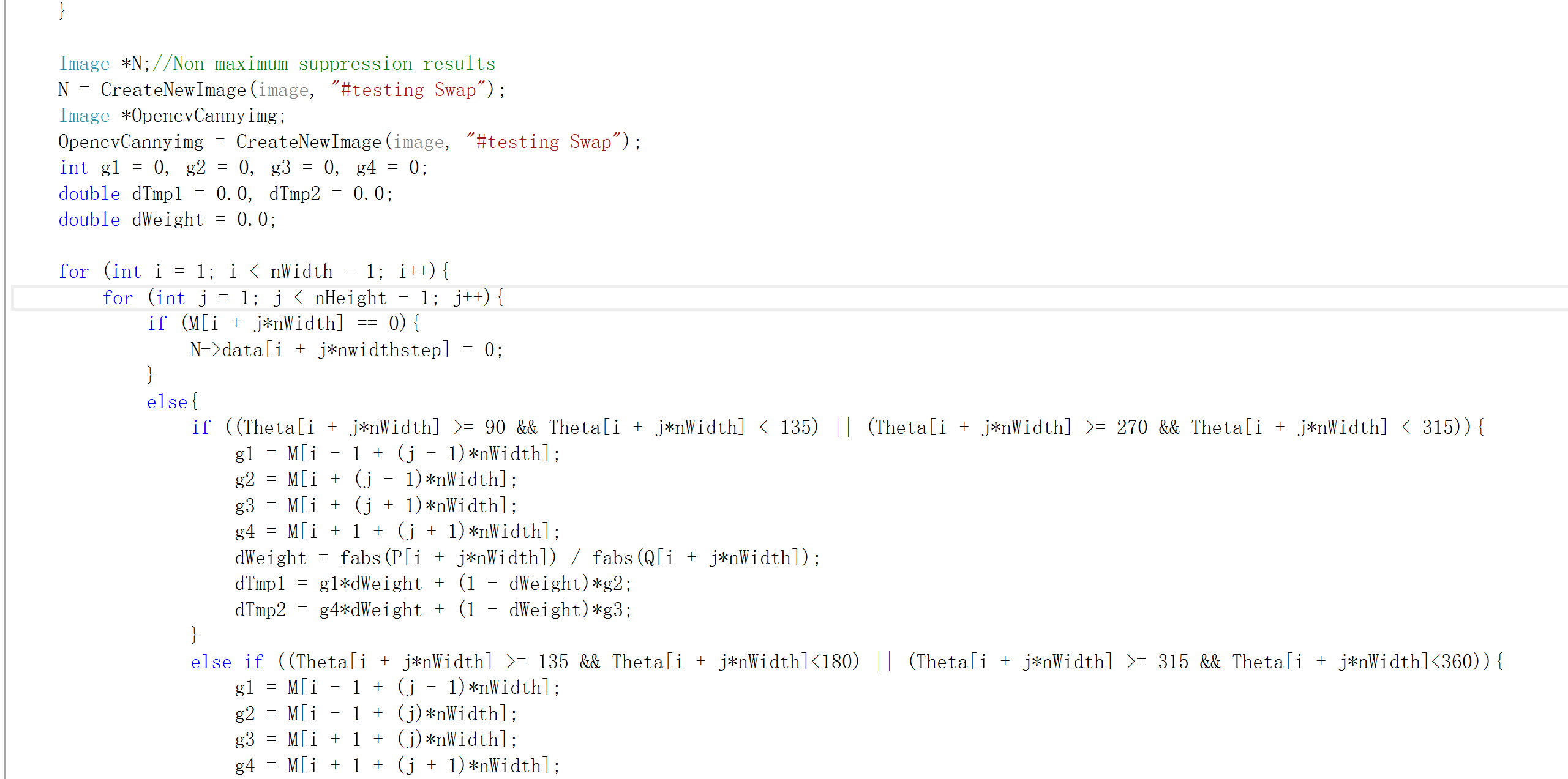
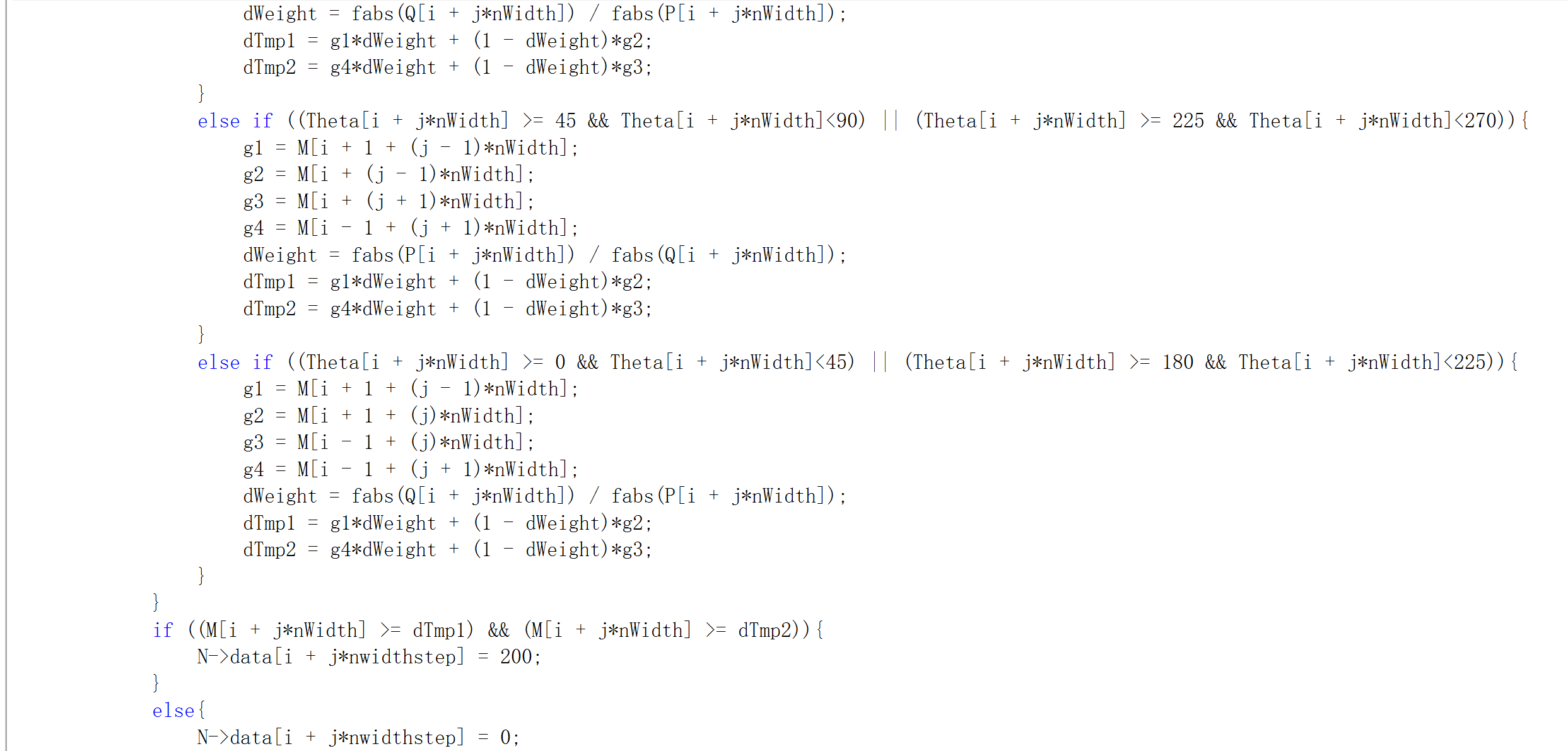
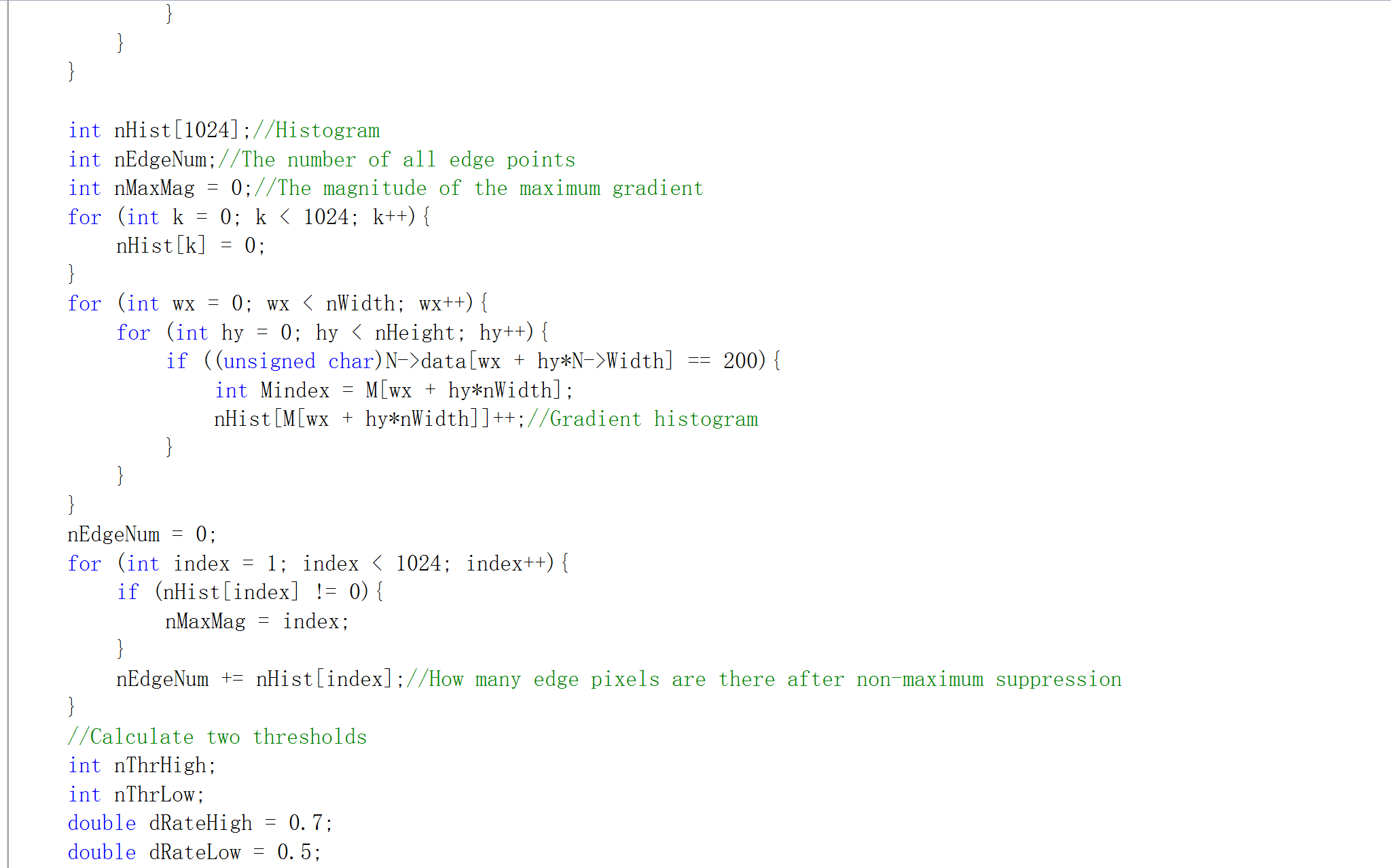
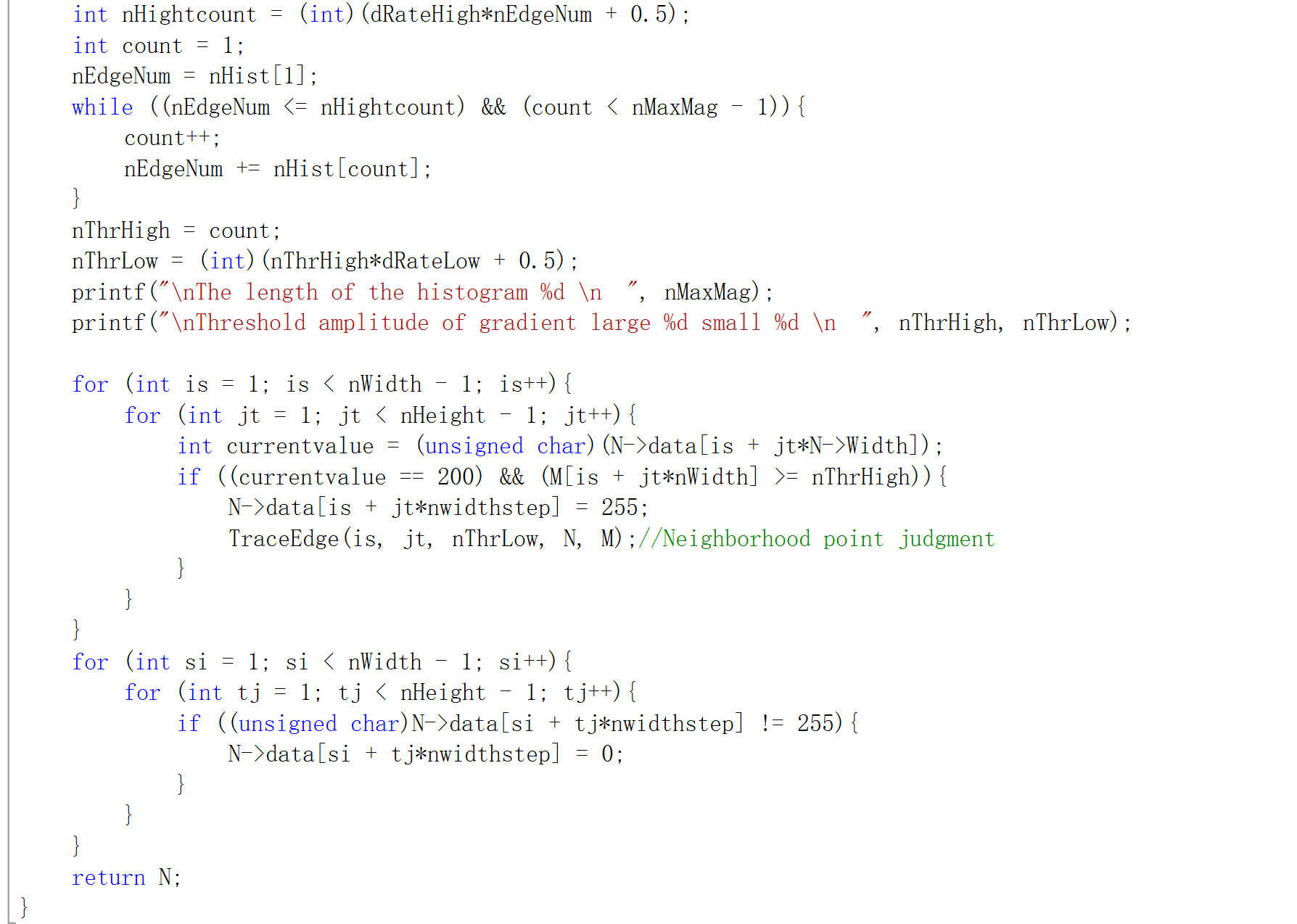
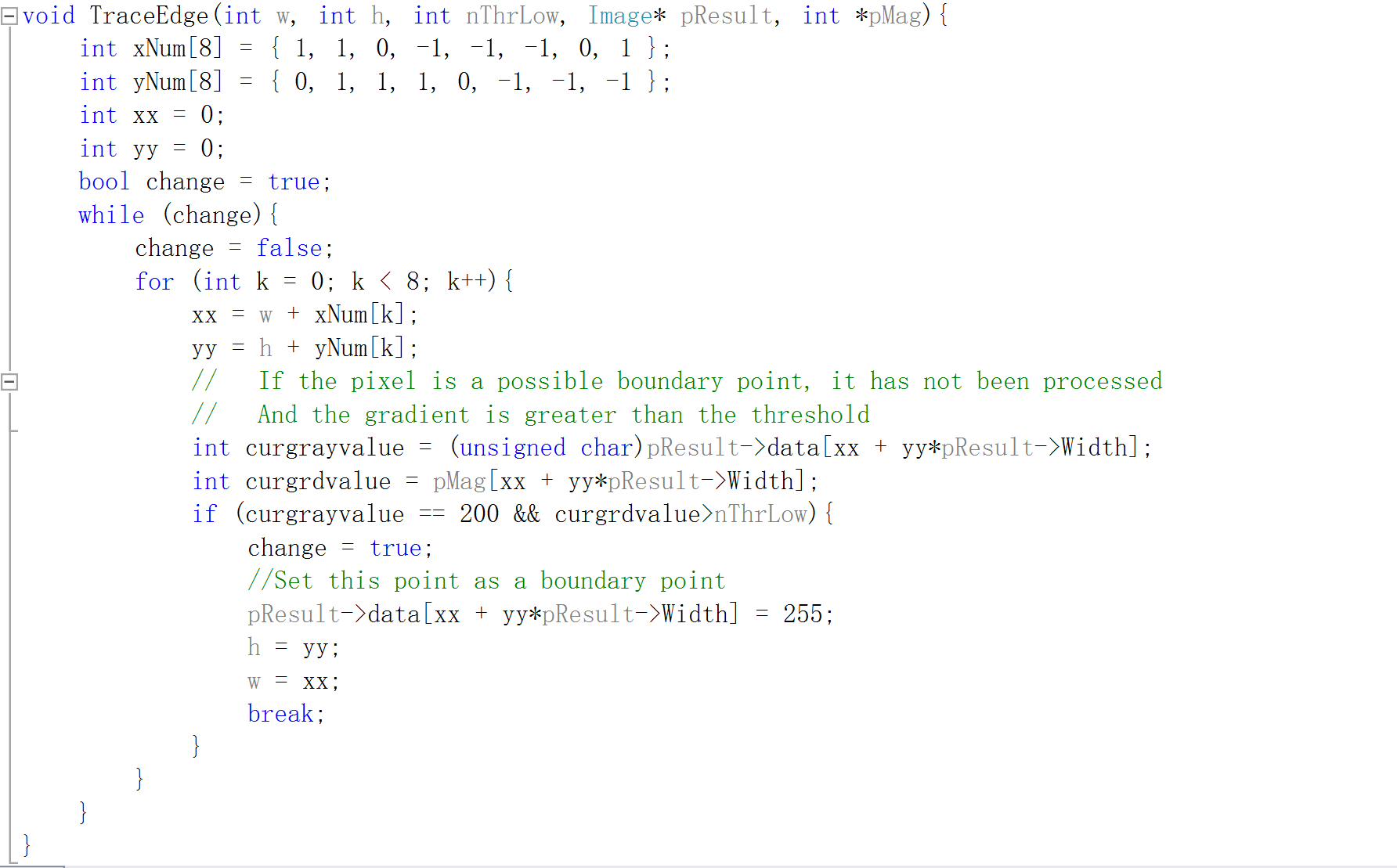
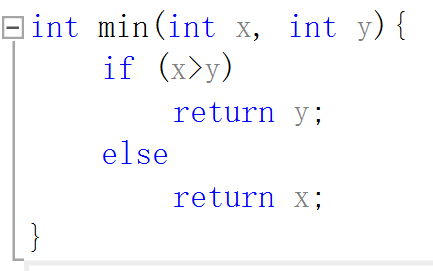
**Discussion:**

In terms of surface effects, the Canny algorithm is a further refinement and more accurate positioning of the effects of operators such as Sobel and Prewitt.

1. Advantages of canny operator: The method can identify as many actual edges in the image as possible; the identified edges should be as close as possible to the actual edges in the actual image.

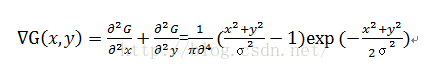
2. Disadvantages of the canny operator: the edge in the image can only be identified once, and possible image noise should not be identified as an edge.

**Code:**

* LoG

**Algorithm:**



is the smoothing function (Gaussian function) selected when processing the image; x, y are integer coordinates; σ is the mean square error of the Gaussian distribution. Do Laplace transform on the smoothed image , we get:

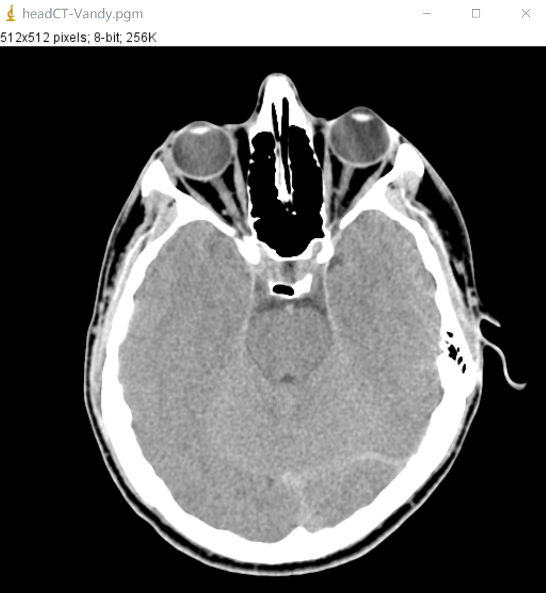
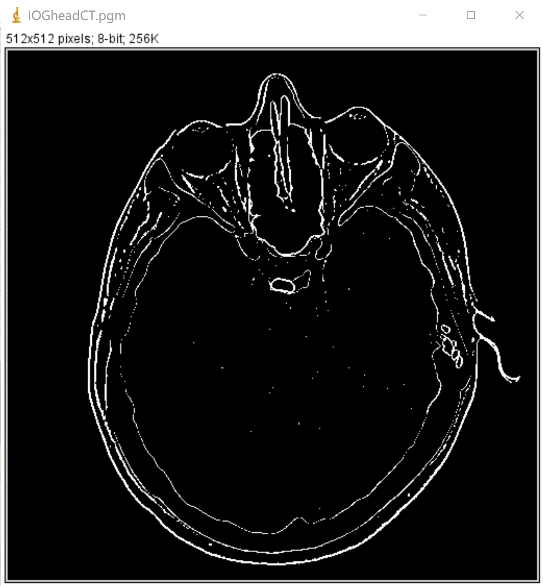


That is, the image is first smoothed, and then the Laplace transform is used to obtain the second-order differential, which is equivalent to applying the Laplace change to the smoothing function to obtain a template with both smoothing and second-order differential functions, and then convolve with the original image. The advantage of using Marr-Hildreth template and image for convolution is that the template can be calculated in advance, and the actual calculation can only be convolution.

**Results (including pictures):**

Result of processing “headCT-Vandy.pgm”:

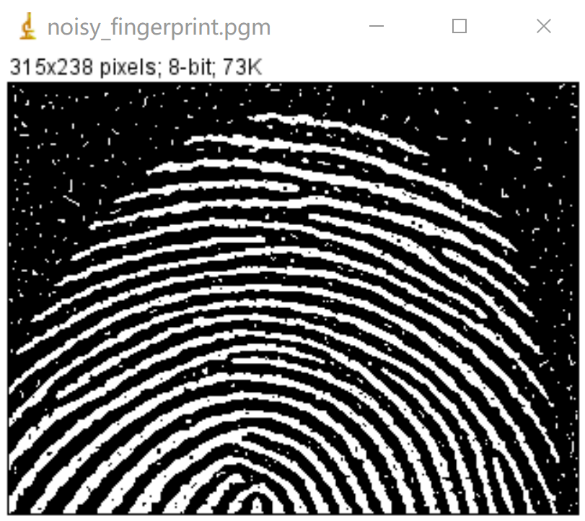
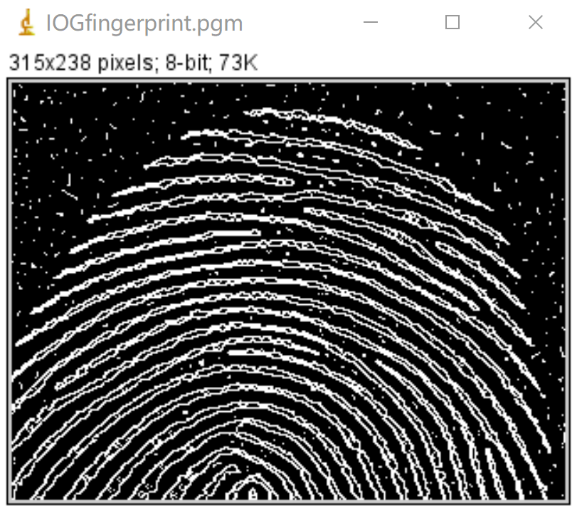
Source Image: Result after Edge detection: (LoG)



Result of processing “noisy\_fingerprint”:

Source Image: Result after Edge detection:

(LoG)



**Discussion:**

LOG filter has the following characteristics:

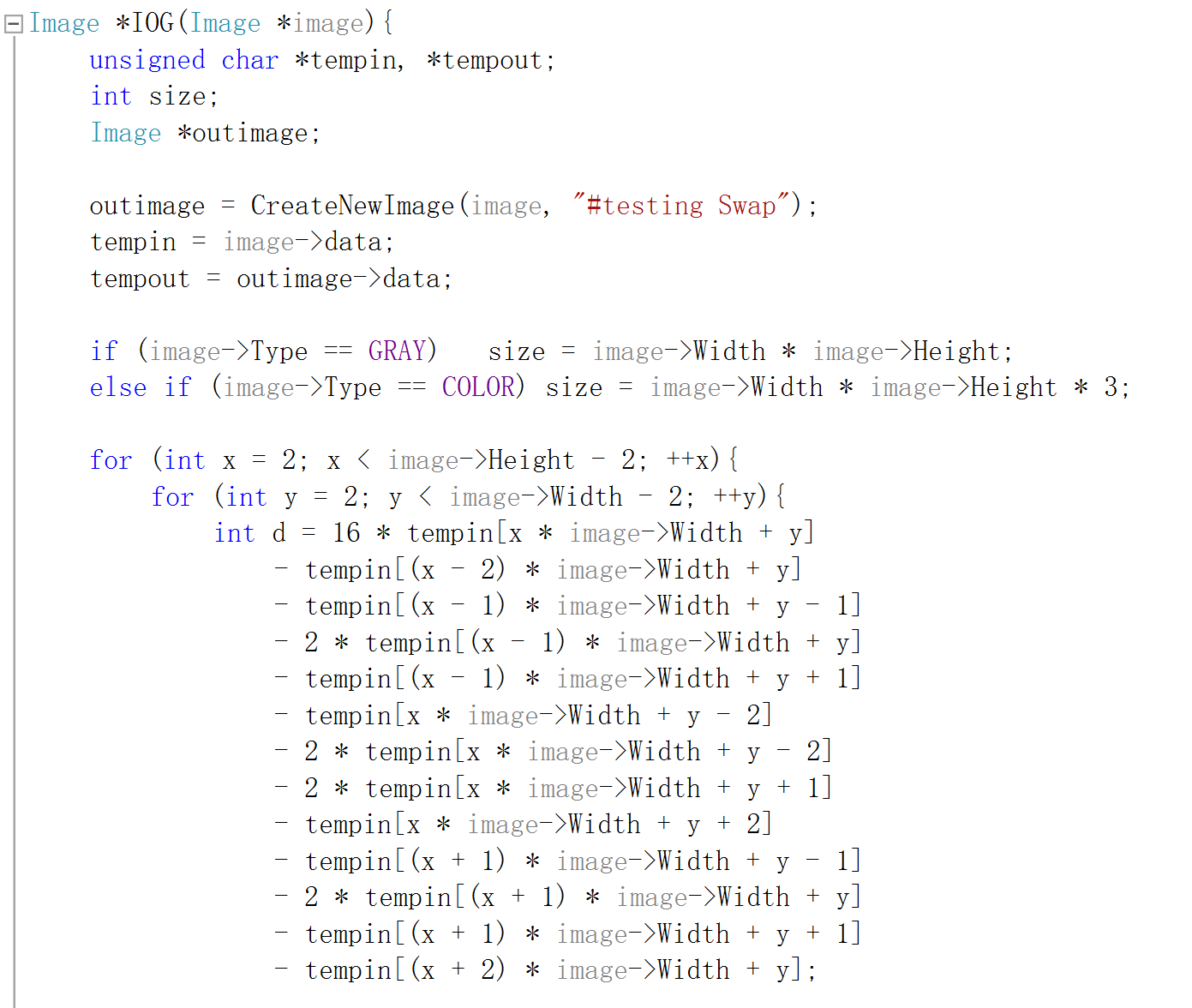
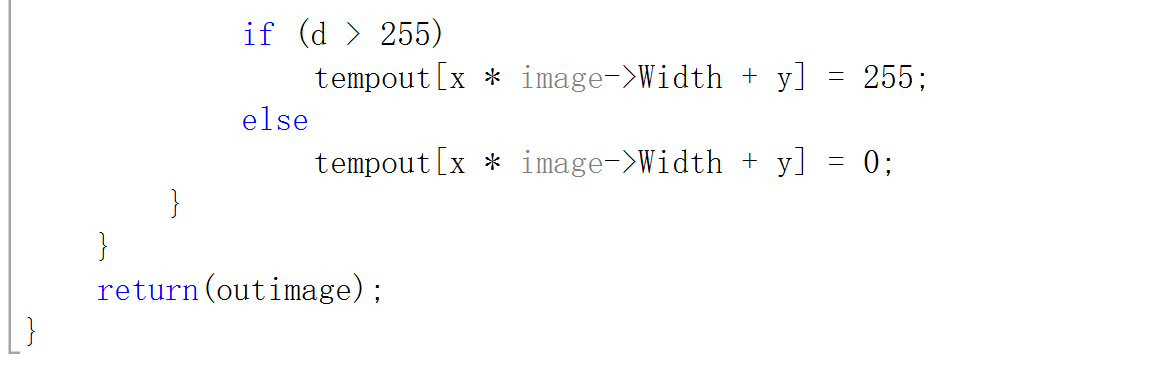
(1) Through image smoothing, all image intensity changes whose scale is smaller than σ are eliminated;

(2) If you use other differential methods, you need to calculate the differential in different directions, and it has no directionality, so you can save the amount of calculation;

(3) It has high positioning accuracy and good edge continuity, and can extract edge points with weak contrast.

The LOG filter also has its disadvantages: when the width of the edge is smaller than the width of the operator, the details will be lost due to the slope fusion at the zero-crossing point.

**Code:**

1. **Global thresholding (polymersomes, noisy\_fingerprint.):**

**Algorism:**

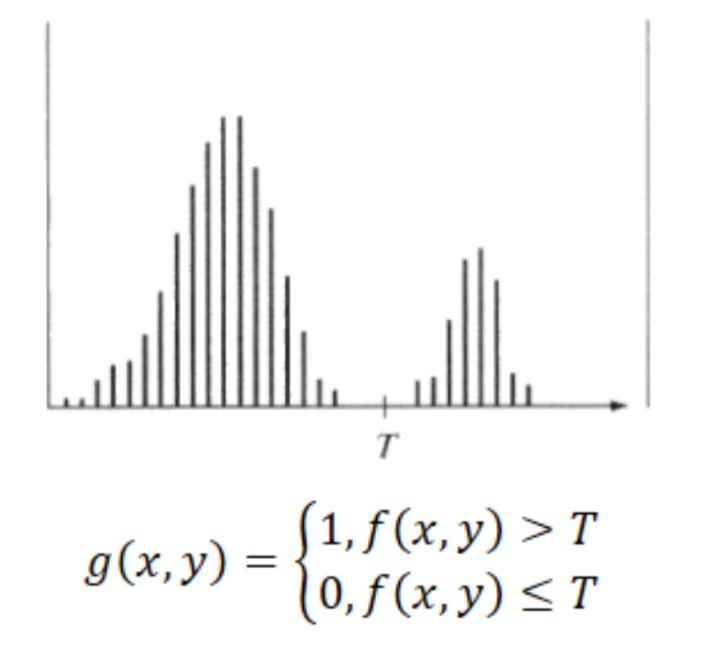
Step 1. Choose an initial estimate for the global threshold T

Step 2. Divide the image with T to produce two sets of pixels: G1 is composed of pixels larger than T, and G2 is composed of pixels smaller than T

Step 3. Calculate the average gray values m1 and m2 for the pixels of G1 and G2, respectively

Step 4. Calculate the new threshold T = 1/2 \* (m1 + m2)

Step 5. Repeat steps 2-4 until the difference of T values in successive iterations is less than a predefined parameter ΔT



**Results (including pictures):**

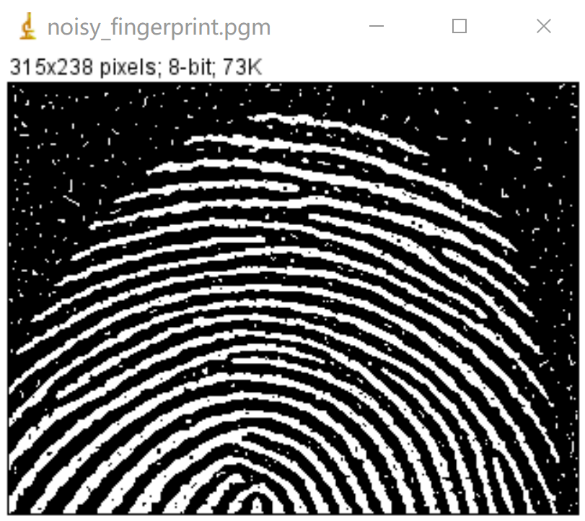
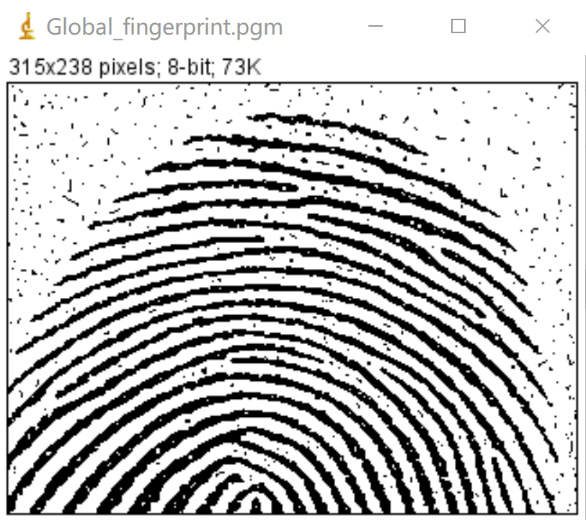
Result of processing “polymersomes.pgm”:

Source Image: Result after global thresholding:

Result of processing “noisy\_fingerprint.pgm”:

Source Image: Result after global thresholding:

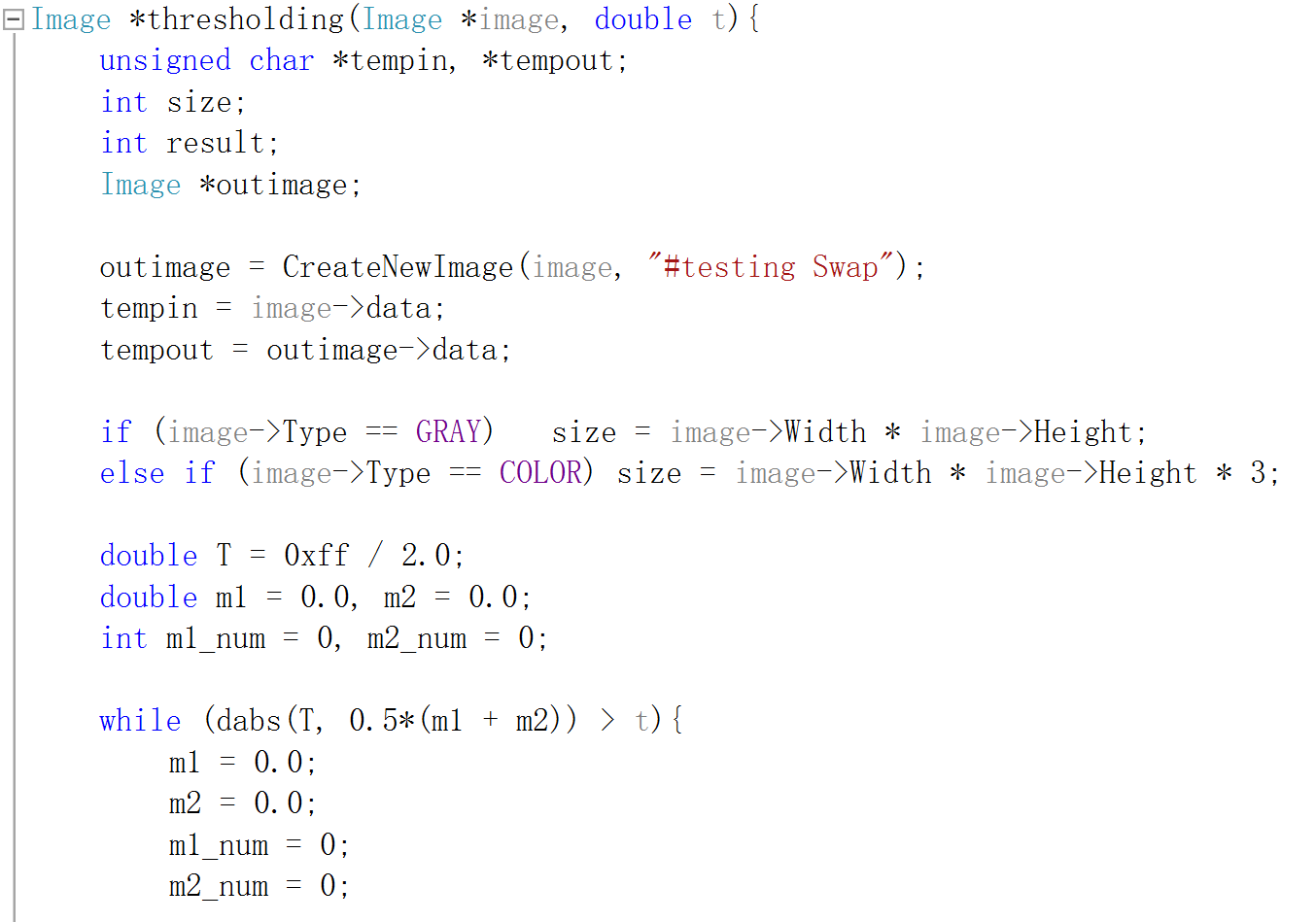
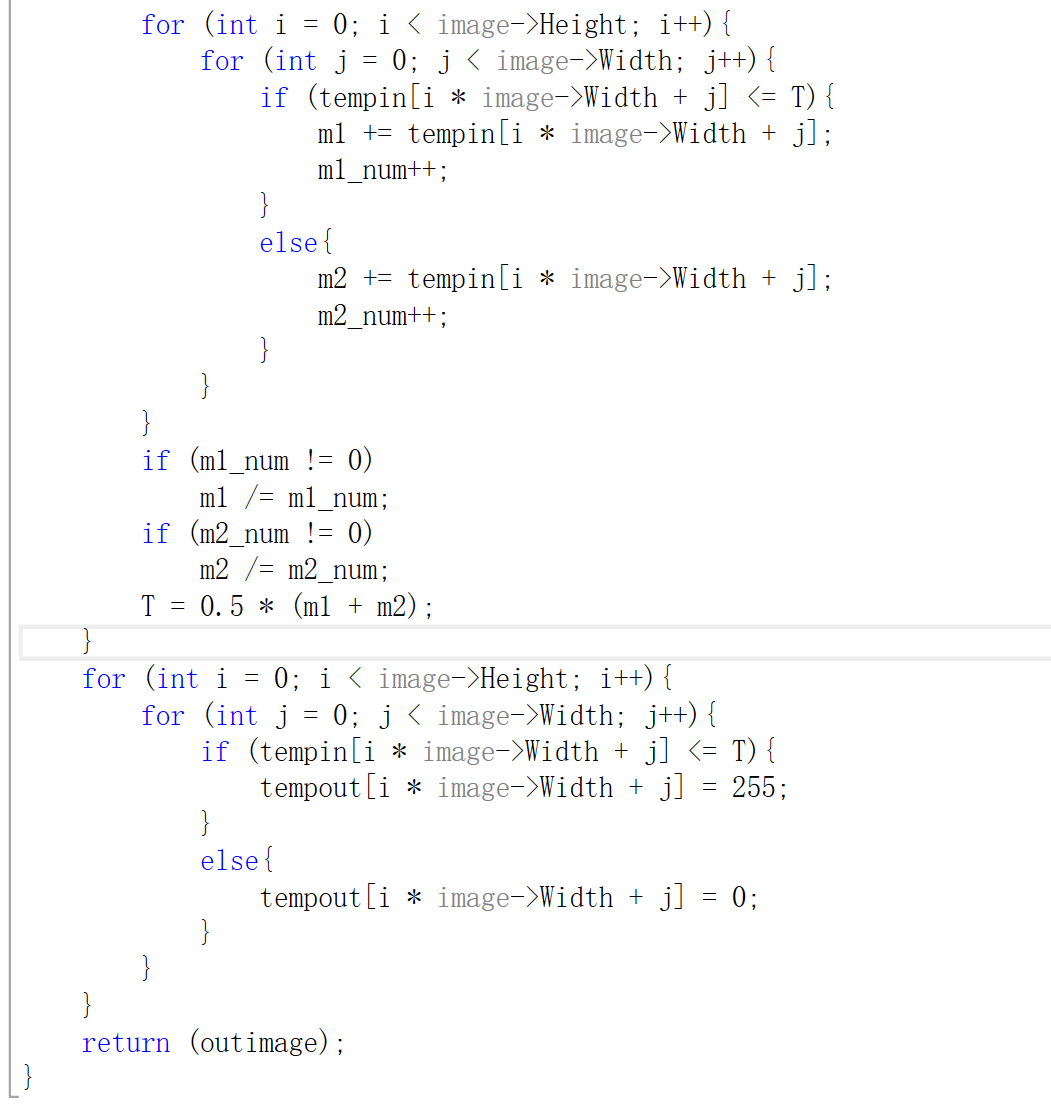


**Discussion:**

This simple algorithm works well when there is a clear trough between the histogram patterns related to the object and the background. In situations where speed is important, the parameter \Delta TΔT is used to control the number of iterations. The average gray level of the image is a good initial choice for T.

The shortcomings of this method are also obvious. It can only segment images with obvious double peaks, and cannot achieve good results for complex images.

**Codes:**

**** **** 