CLEMSON UNIVERSITY

Canny Edge Detection

Assignment 3, ECE 847

Ujwal C Naik

Assignment Number 3

Report for the 3rd assignment for Digital Image Processing

Introduction:

The aim of this assignment is to implement the Canny edge detector. Additionally, chamfer distance calculation is performed in order to implement template matching.

Method:

The working of the Canny edge detector can be explained in 3 steps:

1) Gradient Estimation.
2) Non-Maximal Suppression.
3) Edge-Linking.

Gradient Estimation: For gradient estimation the image is convolved with a Gaussian kernel and the derivative of the Gaussian kernel as explained below:

$g_x = Image * G_x * G_y' \ldots\ldots\ldots(i)$

Here the image is convolved vertically with a Gaussian kernel and the result of this operation is convolved horizontally with the derivative of the Gaussian kernel to get gradient image in the X-direction.

Similarly, gradient in the Y-direction can be calculated as

$g_y = Image * G_x * G_y'$.

The magnitude of the gradient can be calculated as

$g(x,y)=\sqrt{(g_x(x,y)^2+g_y(x,y)^2)}$

and the angles formed by each gradient is also calculated as

$\Theta(x,y)=(\tan)^{-1}(g_y(x,y)/g_x(x,y))$.

Non-Maximal Suppression:

It is the process of suppressing all the components in the gradient image that aren't local maxima.

Based on the orientation/direction of the gradient we decide which neighboring pixels we will analyze. If Θ is negative we add 180 to its value so as to have all angles in the positive range.

Θ	Pixels to be analyzed
$\frac{-\Pi}{8}\le\Theta\le\frac{\Pi}{8}$	(x-1,y) && (x+1,y)
$\frac{\Pi}{8}\le\Theta\le\frac{3\Pi}{8}$	(x-1,y-1) && (x+1,y+1)
$\frac{3\Pi}{8}\le\Theta\le\frac{5\Pi}{8}$	(x,y-1) && (x,y+1)
$\frac{5\Pi}{8}\le\Theta\le\frac{7\Pi}{8}$	(x-1,y+1) && (x+1,y+1)

The value of any given pixel will only be retained if it is the maximum amongst the other two pixels considered.

Edge Linking:

The thresholds are computed automatically based on the histogram. Then edge linking is performed by using seed points from the high threshold image and performing floodfill on the low threshold image. Effectively performing double thresholding.

Chamfering and Template Matching

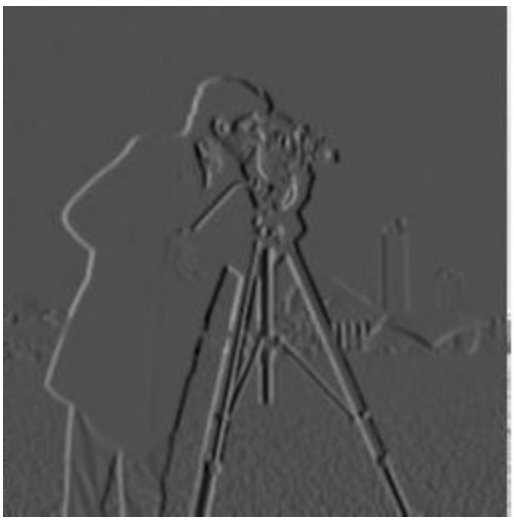
The second part of the assignment includes computing the chamfering distance which is essentially the distance of all the points from the nearest edge. Chamfering distance is used to find the location of the template in the image.

Algorithm:

1) Calculated X-Gradient and Y-Gradient images.
2) Calculated phase and magnitude images.
3) Performed Non-Maximal Suppression.
4) Automatic thresholding to get t_{high} and t_{low}.
5) Edge linking with floodfill.
6) Computed chamfer distance image.
7) Computed probability map using chamfer distance.

RESULTS

Results on the cameraman image:



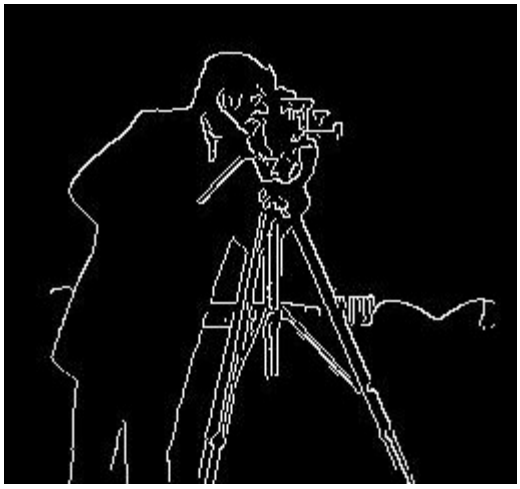
gx

gy

Mag

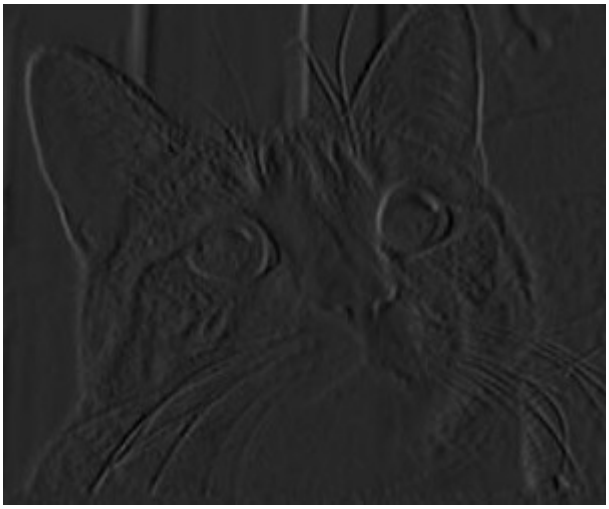
angle

NonMaximum Suppression

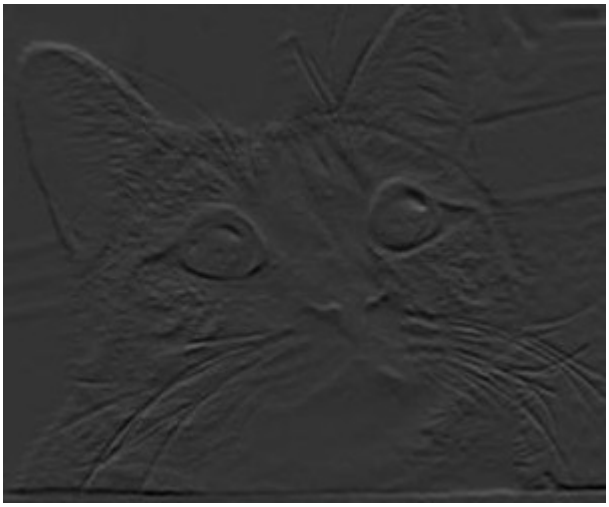


Canny

Results on Cat.pgm



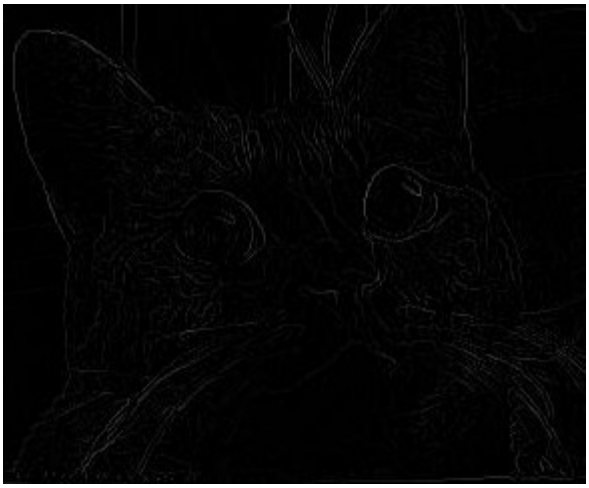
gx



gy

Magnitude

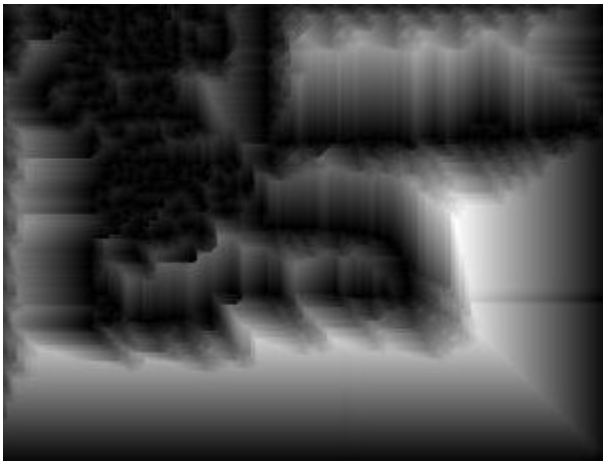
Phase



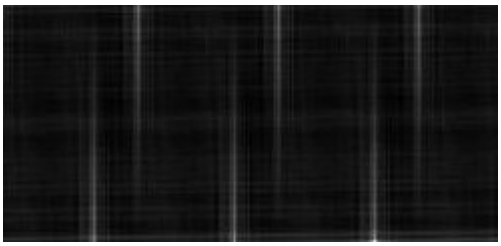
Nonmaximum suppression

Canny Edges

Results for Template Matching



Chamfering result



Probability Map

Final Result.

Conclusion

The canny edge detector implemented works satisfactorily for given test images.

There are various factors that affect the performance of the edge detector. Most natural images which have low contrast will have a few edges missing due to the lack of variation of color. Another factor affecting the outcome of the edge detector is the size of sigma, with more edges being lost due to smoothing. The thresholding values decided upon by the user also affect the quality of the output.

Chamfer distance based template matching was also reasonably successful for different values of sigma.