Homework 2: EE569

Ishaan Vasant

6989-5065-37

ivasant@usc.edu

Problem 1: Edge Detection

Abstract

Edge detection is identifying points in the image where there are sharper changes in brightness. It has a lot of uses in image analysis, pattern recognition and computer vision. Several edge detection techniques were carried out and compared.

Approach

First the Sobel mask was applied to both images and then a threshold was tuned to obtain the edge map. In the next part, the Canny edge detector with its upper and lower threshold values was used to get the edge map. The Structured Edge algorithm was applied to both images using the provided code and probability edge maps were obtained. These were converted to binary edge maps using threshold values. The edge maps were evaluated using the code provided and precision, recall and f-measure metrics were used to compare their performance.

Results

Fig 1: Sobel Edge Detector applied to Pig and Tiger images

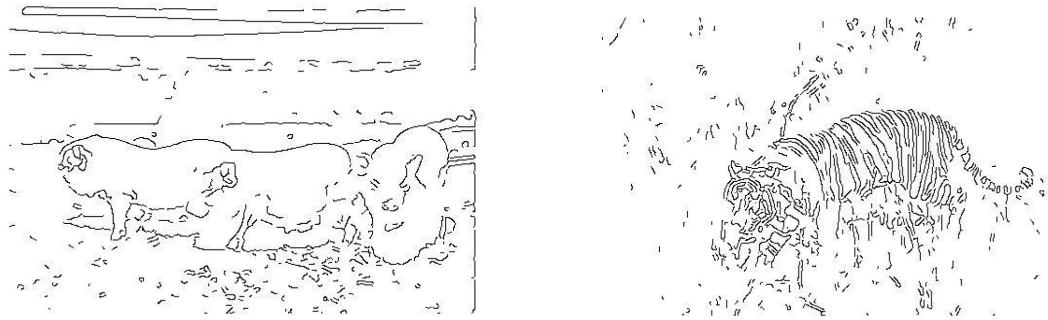


Fig 2: Canny Edge Detector applied to Pig and Tiger images

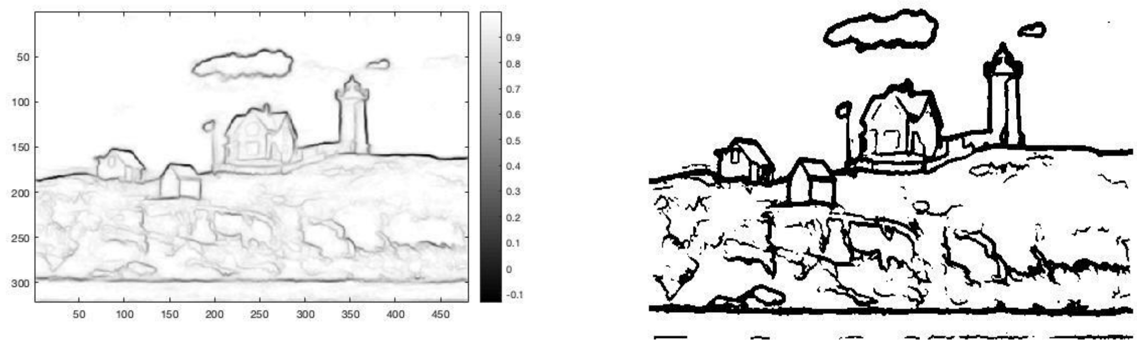


Fig 3: Structured Edge – Probability edge map and Binary edge map of House image

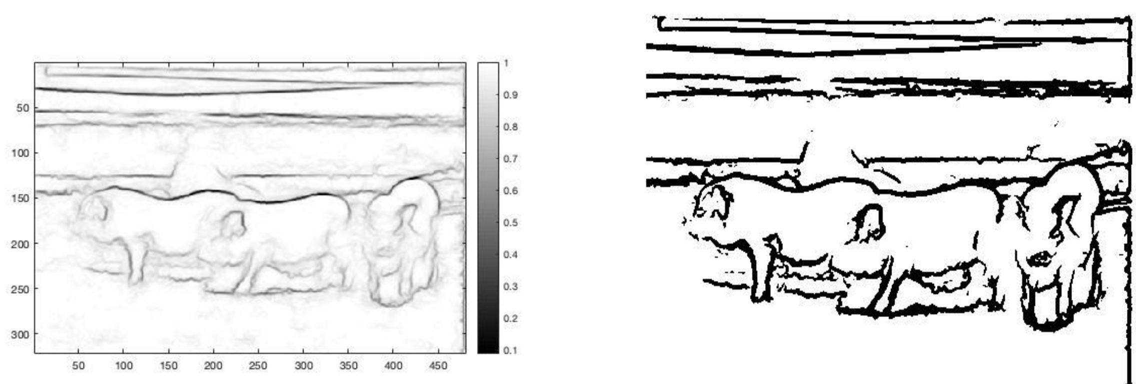


Fig 4: Structured Edge – Probability edge map and Binary edge map of Pig image

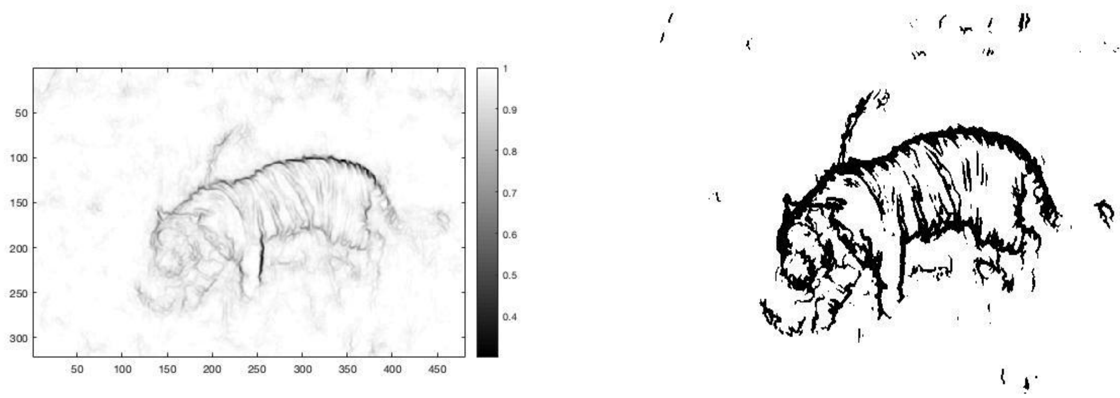


Fig 5: Structured Edge – Probability edge map and Binary edge map of Tiger image

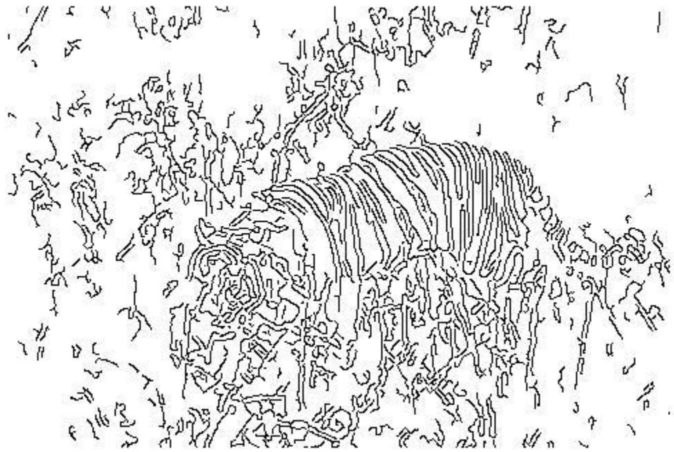
Discussion

a) The Sobel Edge detector was applied to the Pig and Tiger images and the results obtained have been shown in Fig 1. After tuning the threshold parameter for each image these were the best edge maps obtained. The percentage threshold for Pig image was 20% of 255. The percentage threshold for Tiger image was 10% of 255.

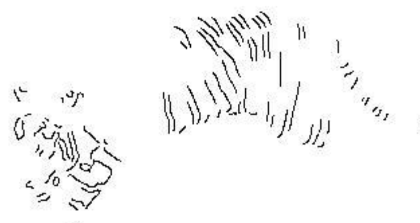
b)

1. Non-maximum suppression is a technique used in Canny edge detection to make the edges thinner by removing pixels that are not considered to be part of the edge. The edge strength of each pixel is compared to the strength of the positive and negative direction gradient pixels. If the strength of the pixel is the largest in comparison, it is preserved. If not, it is suppressed.

2. If the pixel gradient value is higher than the upper threshold, it is preserved and termed as a strong edge. If the pixel gradient value is lower than the lower threshold, it is suppressed. If it is in between the two, it is termed as a weak edge and is preserved only if it is connected to a strong edge. The resulting edge map reported has a threshold value of [0.2 0.25].



Above is the edge map of the tiger image using canny detector and threshold of [0.1 0.2]. This edge map provides too much unnecessary detail, and hence does not fulfill its purpose in identifying the useful edge.

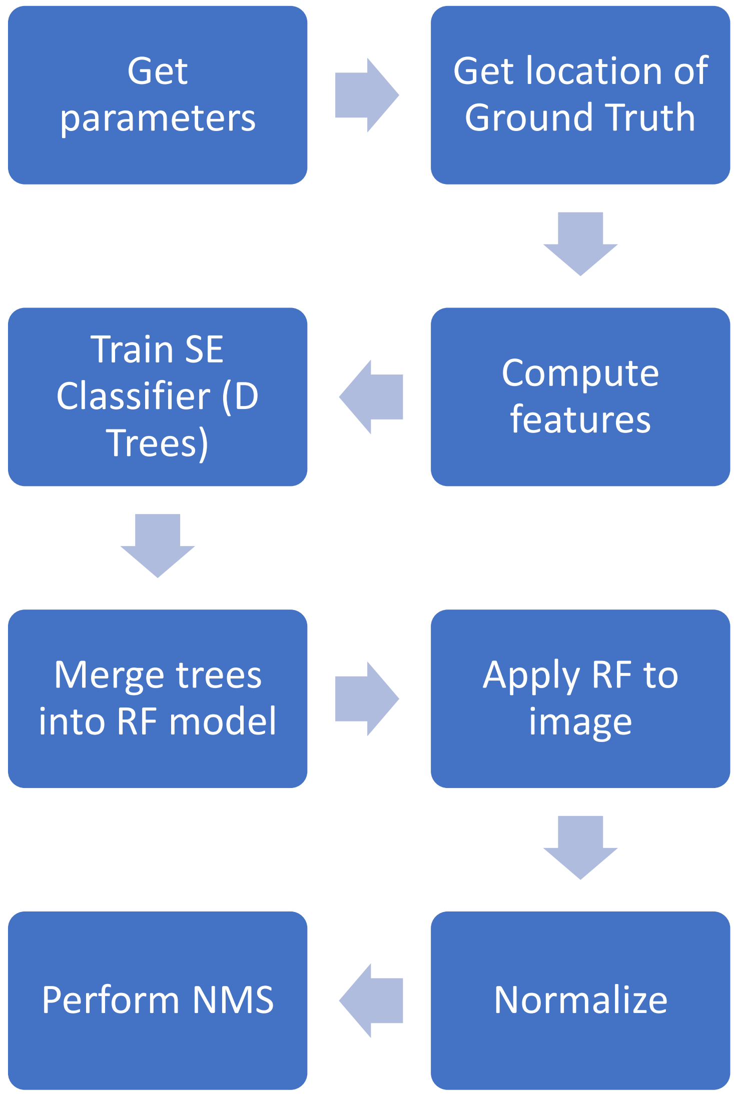


Above is the edge map of the tiger image using canny detector and threshold of [0.5 0.6]. This edge map provides very little detail as most of the pixels are suppressed. This too does not fulfill its purpose in identifying the useful edge.

c)

1. The Structured Edge detection algorithm is a data driven technique which exploits the idea that patches of edges exhibit well-known forms of local structure. The Random Forest classifier and BSDS dataset is used for this method. In the algorithm, first the parameters of the image are considered. Following this, the locations of the ground truth is found. Based on this the relevant features are computed. Using these features, a structured edge classifier using random decision trees is trained. After training the trees, they are merged into the random

forest model. This random forest model is applied to the image. The image is then normalized, and the edge maps are finalized. In the end, non-maximum suppression is performed to make sure the edges are thin. This is depicted in the following flow chart: -



2. A decision tree classifier is constructed as a flow chart which starts from the root node. The classifier puts a threshold on a feature and splits it into 2 branches so that class labels can be assigned after testing on it. In this manner, each data point is classified based on thresholding the different features. The random forest classifier is an ensemble method which trains multiple decision trees and classifies data points on the basis of all the decision tree outcomes.

3. To convert the probability edge map to binary edge map for the tiger image, a threshold of $p<0.92$ which translates as **p>0.08**. For the pig image, a threshold of $p<0.88$ which translates as **p>0.12**. The same threshold of $p>0.12$ was applied to the house image. In comparison to the Canny edge detector, the SE detector seems to have thicker edges but it shows the relevant edges in a much better way.

d)

1. Structured Edge

   Tiger:-

Ground Truth	Mean Recall	Mean Precision	Mean F-measure
GT4	0.134	0.477	0.13
GT5	0.14	0.10	0.14

   Best F-measure = 0.575

   Pig:-

Ground Truth	Mean Recall	Mean Precision	Mean F-measure
GT1	0.0097	0.124	0.017
GT2	0.155	0.255	0.193
GT3	0.21	0.178	0.189
GT4	0.2203	0.180	0.197
GT5	0.443	0.218	0.293

   Best F-measure = 0.473

   The SE detector seems to perform better on the Tiger image than the Pig image.

2. It would seem to be easier to get a high F-measure on the pig image than the tiger image because the tiger image has so much more detail and it could be harder to properly identify those close edges.

3. The F-measure is designed in such a way so that it gives equal importance to both precision as well as recall. In the formula $2PR/(P+R)$, if R is significantly less than P (i.e. $R<<<P$), then the denominator and be approximated to P and hence the F-measure would equal 2R. Since R is negligible, the F-measure would be very small as well. Maximum F-measure is 1. Using the formula, for max F-measure, $2PR=P+R$ or $2=1/R + 1/P$. This implies $R=P=1$. It is equal since F-measure is a harmonic mean of precision and recall.

Problem 2: Digital Halftoning

Abstract

Halftoning is a technique which uses dots to generate a gradient in an image. It is of huge importance to the printing industry. A few halftoning techniques on grayscale and color images have been performed and reported.

Approach

For the first part, dithering was performed using a random threshold for each pixel value. Dithering was also performed using I_2, I_8 and I_{32} dithering matrices. In the next part, halftoning was performed using error diffusion matrices - Floyd-Steinberg matrix, Jarvis, Judice, and Ninke (JJN) matrix as well as Stucki matrix. A color image was halftoned using the separable error diffusion technique where the image was transformed from the RGB color space to the CMY color space and error diffusion was carried out on each channel individually. Finally, halftoning for a color image was done using the MBVQ based error diffusion technique. All results were compared.

Results

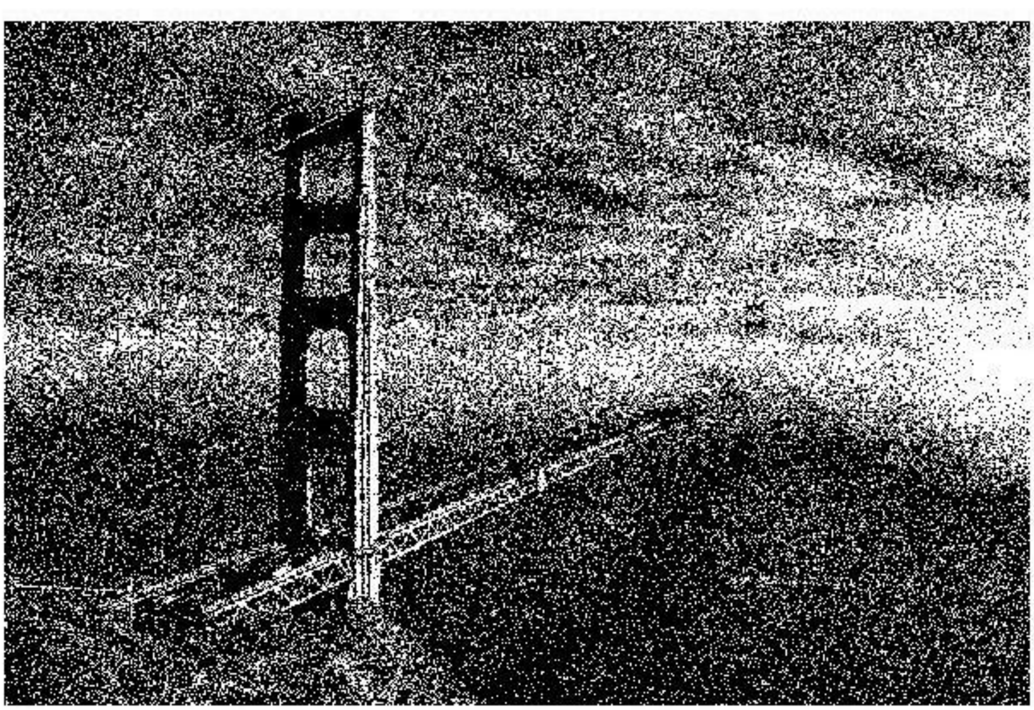


Fig 6: Halftoning using Random Thresholding

Fig 7: Halftoning using I_2 dithering matrix

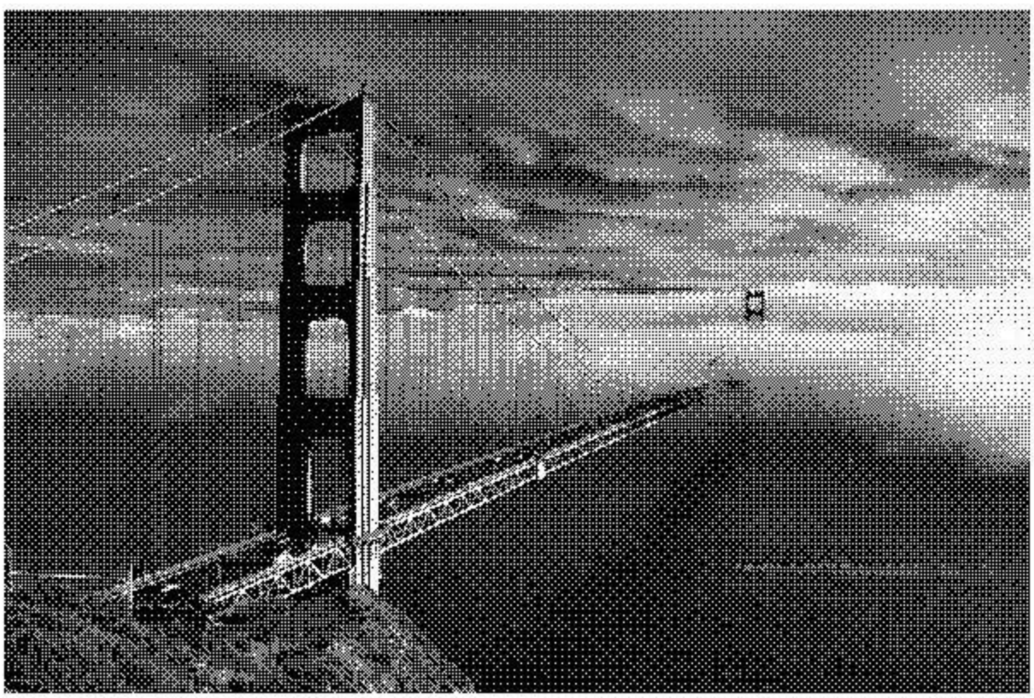


Fig 8: Halftoning using I_8 dithering matrix

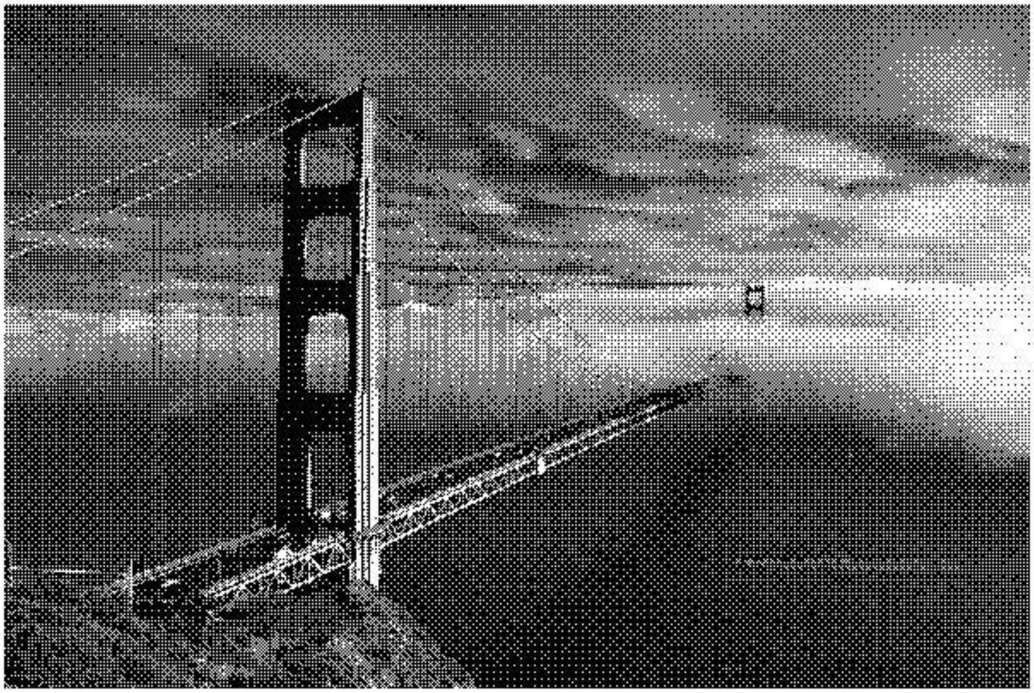


Fig 9: Halftoning using I_{32} dithering matrix

Fig 10: Halftoning using Floyd-Steinberg error diffusion

Fig 11: Halftoning using JJN error diffusion

Fig 12: Halftoning using Stucki error diffusion

Fig 13: Bird image in CMY color space

Fig 14: Color halftoning using Separable Error Diffusion


Fig 15: Color halftoning using MBVQ-based Error Diffusion

Discussion

a) On comparing the halftoning performance of dithering with random thresholding, I_2, I_8, I_{32} matrices, it can be said with certainty that random thresholding performs the worst. The I_8 and I_{32} matrices result is very similar and both show a clearer square pattern in the image when compared to I_2.

b) There does not seem to be too much of a difference in performance between JJN and Stucki error diffusion with Stucki seeming to do just a little bit better in the area with the water body. Both of these two perform better than FS error diffusion in the area of the clouds and in the area with vertical lines of the bridge. And I would prefer either of the error diffusion techniques instead of dithering since there is no obvious pattern seen in the image. An idea to improve results could be to first perform error diffusion where the matrix diffuses error in all directions. As a second step, one could perform the error diffusion in the standard manner. This could reduce the error and provide accuracy.

c) 1. The main shortcoming is that all 8 colors are used to render colored pixels leading to high halftone noise. The image looks darker than it should due to the dark halftones.

2. The MBVQ method renders colors using only 4 colors as opposed to 8. This is using defined color quadrants. It chooses the color for which the brightness variation is minimum. Since this method reduces the number of participating halftone colors from 8 to 4, it also reduced the halftone noise and brings it to a minimum. Therefore, the output obtained is not as dark, the halftone noise is reduced, and the dot pattern are more regular.