Asli Gulen 112887-5
Kayhan ince 116246-0
Submitted to Turan Yuksel

CENG 466 HW#3 REPORT

PART 1: BOUNDARY EXTRACTION

As stated in the assignment, first of all we detected the edges by using Prewitt filtering with a 3-by-3 mask, which is shown in the Appendix part. To take the transform we needed to binarize the image with a threshold of 100.

To the image obtained from the preprocessing procedure stated above, we applied the Hough transform of straight-line equation with the representation of

$x*\cos(\theta)+y*\sin(\theta)=r$.

In the implementation of Hough Transform, first of all we created a matrix for parameter plane having (2* (max. r value)+1) rows for r values and 46 columns for theta values. Maximum value of r is calculated as $\sqrt{(imWidth)^2+(imHeight)^2}$. This matrix stores the accumulation numbers of (r,θ) pairs. For every point of the binary image having intensity value 1, we calculated the r values for all 46 theta-values in the range of $[-\pi/2,+\pi/2]$ using the equation above. We incremented the accumulation number corresponding to that r and θ pair.

Then we selected the points with maximum accumulation numbers. For each selected point, we found the corresponding line in the x-y plane using the equation:

$x=(r-y*\sin(\theta))/\cos(\theta)$ if $\theta\in[-\pi/4,\pi/4]$

or

$y=(r-x*\cos(\theta))/\sin(\theta)$ if $\theta\in(-\pi/2,-\pi/4)\cup(\pi/2,\pi/4)$

and connected components on that line by iterating y or x respectively. If the length of connected components is larger than a specified value (20 in our implementation) we sketch the line component on the output image.

PART 2: REGION SEGMENTATION

Seed Pixel Selection

Since the color information is kept in 24 bits, there is the chance of obtaining 2^{24} different color types. Because of such a large number, two main problems occurred:

i) Number of colors is very large and finding the color that belongs to the maximum number of pixels in the image (color that has the peak value in the histogram) requires very large memory block.

ii) As the color range is very large (0 to 2^{24}-1) very similar colors (i.e. 25,25,26 and 23,25,26) will be very apart in the histogram .

In order to solve these problems, first of all, we tried to reduce this number. Hence, we shift each of the 8-bit RGB values by two bits to right in order to let colors have a

closer representation. Then we can easily regard similar colors to belong to the same region.

We selected the color value corresponding to the peak element of the histogram of 2^{18} color values. Calculating such an histogram requires:

2^{18} *sizeof(int) = $2^{18} * 2^2 = 2^{20}$ bytes = 1 MB memory

which is not very large but in order to reduce this number, instead of calculating the maximum by creating the histogram we performed several passes over the image and in each pass we count the number of several color values. In every iteration a maximum value is found. The overall maximum is the max. of these values. This process reduces the memory needed by (1/n) for n passes.

After selecting the color of seed pixel, next step is finding an appropriate place for the seed pixel. Choosing any pixel having the seed color would lead some error because that pixel could be a noise. We consider the colors of the neighbors of the candidate seed pixel as well. Pixels that fall into the 21 X 21 frame (having the candidate seed pixel in the center) are considered as neighbors.

While traversing a 21 X 21 frame over the entire image, we take the difference of color value of each pixel in the frame and the seed color value and sum these up. (The method of calculating the difference is explained later.) We determine the pixel, which has the minimum value of this sum as the seed pixel.

Similarity Criterion

Similarity criterion is the difference of two color values calculated as:

$$diff((R_1, G_1, B_1), (R_2, G_2, B_2)) = \sqrt{(R_1 - R_2)^2 + (G_1 - G_2)^2 + (B_1 - B_2)^2}$$

Preprocessing Procedure

First a 3x3 median filter is applied to the image in order to reduce the noise.

After performing noise reduction, we calculate the standard deviation of each color channel (RGB). Threshold value is calculated using these standard deviations. Threshold is proportional to the average of standard deviations of RBG channels. Constant of the proportion is determined by evaluating some candidate values.

As described before, intensity values of RGB channels are shifted 2 bits right.

APPENDIX I

1	0	-1
1	0	-1
1	0	-1

Vertical Prewitt Filter Mask

1	1	1
0	0	0
-1	-1	-1

Horizontal Prewitt Filter Mask