

CAP 5400 Digital Image Processing: Assignment # 1

Palm Segmentation

Due 3:30 pm on September 10, 2001 (100 pts)

1 Data Gathering

Collect color images of your palm using the PCs in the Center Four Labs. First, collect an image of the scene without your palm – this will be your background image against which you will compare to find the palm. Let us refer to this image as I_{Back} . Next, collect three images of your palm, one with your fingers open to the widest possible distance, second one with your fingers close together, and the third one with your fingers open to a relaxed position. For all these three cases, your palm should be directly (90 degrees) facing the camera, vertical, and should occupy at least half of the image area. We will refer to these three images as I_{Open} , I_{Shut} , and I_{Half} . (You are, of course, free to gather and experiment with more images, but these three must be used in your report.)

Convert the acquired images into PPM format for further processing. I have provided an C++ image class, with functions to read and save images in PGM and PPM formats, which can use in any manner.

2 Ground Truth Creation (20 pts)

For *each* of the three palm images manually create ground truth masks representing the palm pixels. These mask would basically be black and white PGM images, with black pixels denoting palm location and white pixels denoting background.

You can use the *gimp* image processing software to do this part.

1. Type: `gimp MyImage.pgm` At least two windows will open up: one with the image and the other will be a command window.
2. Right click on the image window. Select “Layers” and then “Layers and channel” from the sub-menu. A new window will pop up.
3. In the layers’ window select, click on the left most button just above the close button. A new window will crop up. This will help you create a new layer.
4. Select the “Transperant” button on the new window and then Select “OK”. The new window will

disappear and you will be left with three windows.

5. In the command window, click on the button in the 7-th row and 1-st column (“Draw Sharp Pencil strokes”).
6. Go to the image window and left-click on the image pixels belonging to the palm. Black blobs would be painted on the image. Delineate the entire palm region in this manner.
7. Go to the “Layers and Channels” window and select the “Background layer” by clicking on it. Then click on the button with the “X” mark on it, to delete the background layer. The original image will disappear from the image window.
8. Right click on the image window. Select “Layers” and then “Flatten Image”.
9. Go to the image window and right click on it, to select the “File” option followed by the “Save As” option to save the mask. Give a filename with a “.pgm” extension and it will save the mask in PGM format. (Make sure you do not inadvertently click the left button in this step.)

3 Coding (30 pts)

You have to code (in any programming language C, C++, Java, or Ada but NOT Matlab or other similar platforms) to separate the palm pixels in the image from the non-palm pixel, i.e. *segment the palm from the image*. Your output will be a black and white PGM image, with black pixels denoting the palm pixels and white pixels denoting the background pixels.

The specific steps of your algorithm will be

1. First compute the difference images of the three palm images with the background image, i.e.,

$$\begin{aligned} DiffI_{Open} &= I_{Open} - I_{Back} \\ DiffI_{Shut} &= I_{Shut} - I_{Back} \\ DiffI_{Half} &= I_{Half} - I_{Back} \end{aligned} \quad (1)$$

Note that for each of the three *DiffI* images, each pixel will have three values, one for Red,

one for Blue, and one for Green. Process each channel independently of the others. Thus, the operation $DiffI_{Open} = I_{Open} - I_{Back}$ is actually,

$$\begin{aligned} DiffI_{Open}^R &= I_{Open}^R - I_{Back}^R \\ DiffI_{Open}^G &= I_{Open}^G - I_{Back}^G \\ DiffI_{Open}^B &= I_{Open}^B - I_{Back}^B \end{aligned} \quad (2)$$

2. Compute the difference magnitude images as

$$\begin{aligned} MagDiffI_{Open} &= \sqrt{\begin{matrix} (DiffI_{Open}^R)^2 + \\ (DiffI_{Open}^G)^2 + \\ (DiffI_{Open}^B)^2 \end{matrix}} \\ MagDiffI_{Shut} &= \sqrt{\begin{matrix} (DiffI_{Shut}^R)^2 + \\ (DiffI_{Shut}^G)^2 + \\ (DiffI_{Shut}^B)^2 \end{matrix}} \\ MagDiffI_{Half} &= \sqrt{\begin{matrix} (DiffI_{Half}^R)^2 + \\ (DiffI_{Half}^G)^2 + \\ (DiffI_{Half}^B)^2 \end{matrix}} \end{aligned} \quad (3)$$

Note that the $MagDiff$ images have just one channel.

3. Threshold the $MagDiff$ images with respect to an user specified threshold to put the image pixels into two classes, palm and non-palm.
4. Repeat the above steps, but this time first convert the RGB into HSI and use just the Hue and Saturation channels to perform the segmentation.
5. Write code to compare a ground truth mask (I_{Mask}) with the corresponding segmentation output (I_{Seg}) in terms of the false alarm (P_f) and detection rates (P_d). Let
 - the number of palm pixels in the ground truth mask (I_{Mask}) be N_g ,
 - the number of palm pixels in the segmented image be (I_{Seg}) N_s ,
 - the number of palm pixels in the segmented image (I_{Seg}) that are not marked as palm pixels in (I_{Mask}) be N_{false} .
 - the number of palm pixels in the segmented image (I_{Seg}) that are marked as palm pixels in (I_{Mask}) be N_{detect} .

Then, the false alarm and detection rates are defined as

$$\begin{aligned} P_f &= \frac{N_{false}}{N_s} \\ P_d &= \frac{N_{detect}}{N_g} \end{aligned} \quad (4)$$

4 Report (50 pts)

You have to produce a 6 to 8 pages report in IEEE two-column format, including all text, figures and references. A link to the format specifications is provided from the web-page, along with a link to a Latex Macro that conforms to the specifications.

Your report should have the following sections. Note: figures and images that are not discussed in the text will not be considered during grading.

1. An abstract summarizing what your report is about and a brief synopsis of your conclusions (read the instructions in the abstract writing guideline link from the class web-page).
2. A short, one paragraph introductory section that should include a synopsis of the prior work on recognition from hand images and products that are available for such.
3. A section describing the code you have written, a block diagram illustrating the interactions between the various functions, and a brief example of how an user would use and interact with your source code.
4. A section describing the programming bugs that you ran into.
5. A section showing *best* and *worst* segmentation results on the three set of images.
6. An analysis section studying
 - (a) the effect of changing the threshold value on the resultant segmentation. How hard is it to choose the value? Is the segmentation very sensitive to the choice of the threshold? Perform this analysis both for the RGB representation and the HSI representation. Produce 6 plots of P_d versus P_f as you change the threshold for each of three images with RGB or HSI as input.
 - (b) Compare the best performances of the RGB and the HSI based segmentations.
7. A concluding paragraph.

5 Submission

1. Submit your source code as email attachment(s) to Tong Luo at tluo2@eng.usf.edu. Make sure your code runs on `babbage` or `suntan`. It is your responsibility to make sure that it does. You might have to demonstrate your code to us if we deem it necessary.
2. Submit your written report to me.