COMPSCI 773 Assignment 3

Due Date: Fri, 5 June 2009 (Demo), Fri 12 June (Report)

The goal of this assignment is:

1. To calibrate a stereo rig of web-cameras
2. To rectify stereo images and test different configurations with PCa=A and LDA
3. To embed all work in a single interface/demo

1. Camera Calibration and Accuracy

Step 1: Design of an interface to control one or two USB cameras

The project will rely on off-the-shelf USB web-cameras to acquire synchronised face images and generate 3D data from the generated stereo pairs. The first critical task is to be able to obtain images from a single USB web-camera plugged on a Windows computer. Your task will be to design an interface that:

- recognises any plugged-in USB camera
- acquires and save images from one or two USB cameras
- will be able to display images from two plugged-in USB cameras
- acquire and save synchronised images from two USB cameras

Step 2: Camera calibration

The accuracy of your calibration process will influence your ability to generate accurate stereo pairs. Your first task is to use the 3D calibration cube (you may design a new calibration object as well) to compute the intrinsic and extrinsic parameters of the calibration matrix for each camera. You will write an interactive program that collects data from your calibration object, the image, performs the calibration and outputs the calibration matrices and parameters. You must also evaluate the accuracy of the calibration.

To use binary machine vision techniques for image processing, the calibration cube should be placed onto a plain background so that the desired disks can be separated from background by thresholding. During the calibration, both cameras must mostly see the same faces of the calibration cube.

Positions of the disk centres (or detectable points of any other calibration pattern you may find suitable) used for the calibration process should be determined mostly automatically. You could, for example, click on one disk to specify desired colour boundaries for binarisation. But all other disks positions and centres should be determined automatically with some interactive correction of wrong matches.

The interactive calibration program has to do the following operations.

- Input and visualisation of the two images forming a stereo pair of the calibration object.

- Computation of points in the two images that correspond to the centers of all the visible disks of the calibration pattern.
- Visualisation of the stereo pair with the overlaid corresponding points.
    - Interactive correction (if necessary) of wrong matches.
- Output of coordinates of the corresponding 3D and 2D points in the format needed by your calibration software.
- Evaluation of the accuracy of the calibration data by comparing the true and reconstructed 3D coordinates of the disks in the calibration pattern.

You should use the cameras to take some stereo pairs of the calibration object and store them as they may be used again to develop and test your program.

Step 3: Calibration improvement

1. Compare real and estimated positions and dimensions of objects in the field:
    - Measure the height of 3D objects within the scene
        - Place an object measured beforehand in the scene
    - Compare to the estimated values you are obtaining from your automatic calibration process if any (automatic means automatic detection/centre computation of patches)
2. Optimise your calibration

        1. By giving less weight to or removing patches (for the calibration process) whose positions are not correctly estimated in your segmentation process.
        2. By iterating your calibration process while removing (or giving less emphasis) to the patches whose positions were not "correctly" estimated in the previous iteration of the calibration and as long as the errors are decreasing
        3. By integrating the distortion parameter(s) in the equation of your calibration process and loop as long as the parameters are changing
            1. For Tsai, equations first consider $K_1 = 0$, then estimate most calibration parameters, then use these estimations to estimate the distortion parameter
            2. For Tsai, try to integrate K_2 (**fourth order radial distortion parameter**)

2. Rectification and 2+3D face

Redo Assignment 2 part 4 this time using your own calibration data and rectification code.

Step 1: Perform the stereo camera rig rectification.

1. Use the calibration parameters computed for the cameras positions during the face database acquisition and unknown faces acquisition to compute the transformation between both cameras.

2. Display the new images in the rectified space

3. Compute the face images depth maps.

Step 2: Rectified image database PCA analysis

1. Using the rectified 2D images, redo the PCA analysis from assignment 2.
   a. You should produce deeper analysis of PCA with confusion matrix scores, false positive, false negative, etc when testing images not used for training
   b. You also test the recognition score on a subset say 10 of untrained images when increasing the training number of images (say one images per person or expression, then 2 images per person or expressions, etc...)
2. Using the depth map images only, redo 1.
3. Combining 2D and depth map images, redo 1.
4. Conclude on the advantages of depth data for face recognition.

Step 3: LDA Analysis

1. Redo step 2 this time using LDA instead of PCA
2. Use ID as classes
3. Use Expression as classes
4. Compare LDA and PCA outcomes

4. Overall demo

Embed all code and parts in a single demo/interface. Reflect on the project's achievements and where you believe you could have made some improvements and how. Provide suitable bibliographic references supporting your claims.

Submission

You should submit your assignment as a printed and electronic report (one per group) before the due date. Once again the title page of your report should identify all the students and the corresponding parts of the report each student wrote. Introduction, conclusion and comments should be the group work. The report should follow IEEE latex or word standards or any other standard supported by major publishing companies.

The report itself should include the following parts:

The new parts developed with respect to the work previously done in the previous assignment
Description of algorithms used for the different parts.
Description of your programs.
Description of your experiments.
Conclusions and/or comments.
Submit as well your interface/assignment code. It should be able to compile on my laptop and I shall be able to test extensively your interface. To allow this, you should write a README and HowtoUse files. Your code should be well commented and understandable. Code borrowed from the web should be commented as well.

Demos

Each group will show a demo on the week the assignment is due.