COMPSCI 773 Assignment 2

Due Date: Thu, 14 May 2009

The goal of this assignment is:

1. To design an interface to support all parts of the project
2. To calibrate a stereo rig of web-cameras
3. To rectify stereo images

This assignment continues the previous one and is a further step towards the overall project goals.

1. Interface design

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

2. 2D Face recognition

You will have to create a list of faces that you want to recognize. You should have at least 20 different faces for your project. Your goal is to classify any given/acquired new face image as part (or not) of your face database.

Experimental conditions

Restricting the experimental conditions, will help you minimize the variability of your database. Make sure all the subjects face the camera (you could even setup a frame within the image that the subject must sit in) from approximatively the same distance keeping first a neutral stance. Although for small database, empirical methods could be enough to separate faces, in the case of large databases, only statistical analysis techniques allow proper results.

Guidelines

The task here can be separated in 3 steps. Although the following provides solution proved successful, it is up to you to diverge from the proposed scheme if you feel confident enough. Step 0: Database acquisition (as described above)

Step 1: Normalisation of the face images (All faces must have the same size with the face features roughly placed at the same position)

1. Locate the face in each image of the database using color segmentation (color predicate, K-means, gaussian mixture, etc...)
2. Find the smallest ellipse (or any other geometric shape) which contains the face
3. Align and center all faces using center, major and minor axis of the ellipse
    1. **Alternatively manually detect eyes and use this axis to align the faces and the eyes location to center the faces.**
4. Rescale all the ellipse so as to have the same major and minor axis length for all images
    1. Alternatively you might consider obtaining a face mask which has got the same area
5. Cut the images down to the smallest image enclosing the ellipse (the background should contain black pixels only)
6. All face images should have the same size.

Step 2: Use PCA to analyse the database using outcomes from Step 1

    1. Alternatively, the input for each face image could be a reduced set of features instead of the pixel images. This way you could reduce the size of the database. Features could be a list of corners, edges, Gabor jets, Fourier descriptors, etc....
    2. Analyse your set of eigenvectors, offer a reduction of your data-space. See how this introduces errors in the reconstructed faces.

Step 3: Recognize a "new" image using outcome from Step 2. Use procedure from Step 1 to process the "new" image so as to be process using outcome from Step 2.

Step 4: Use the face expression images to recognise an expression rather than the Image ID.

3. 2+3D face

Here you will use the rectified version the acquired stereo image pairs of faces (as provided by Edwin) to allow the computation of faces depth map. You will then integrate the depth map data to the PCA analysis of your face database.

1. Using the rectified 2D images only, redo 2.
2. Using the depth map images only, redo 2.
3. Combining both 2D and depth map images, redo B.
4. Conclude on the advantages of depth data for face recognition.

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 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.