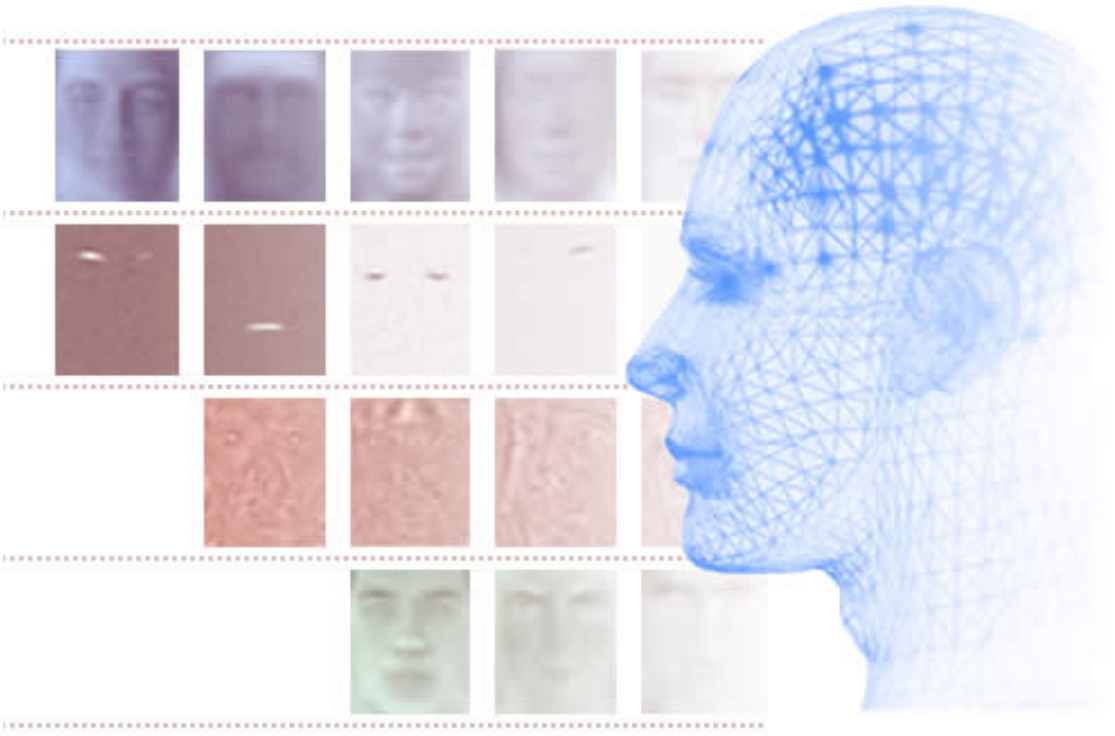
TNM034 - Advanced Image Processing

Course HT 2017



http://www.face-rec.org/

Project:

Face Recognition

TNM034 – Advanced Image Processing

In the course TNM034 - Advanced Image Processing, the students will learn advanced image processing and current research issues, mainly by carrying out a project task in the area. After the course the students will be able to:

- plan how to solve the project goals.
- describe the project context and the theories behind the solution.
- identify and implement a particular solution to the problem.
- compile a report over the project, describing the implementation and the algorithms used.

The main target for the students is to assimilate the literature and independently solve and implement the given course task. The course consists of lectures and practical laboratory work. The lectures will give an overview of the topic and some background and theory related to the given project. The laboratory work is devoted to the accomplishment of the software program. The project work is performed in groups of 2 to 4 students.

Knowledge in writing Matlab programs and the basic course on image processing, TNM087 or TNCG22, are prerequisites for participation in the course.

The list below shows examples of various projects over previous years. The projects have been chosen because of their interesting content of image processing:

- Restoration of image sequences
- Verification of fingerprints
- ”Snakes” and ”Live Wire”
- Stitching and image assembly
- QR-code processing
- Optical Music Recognition

The project task 2015: Face Recognition

The task for this years project is to implement software for face recognition. The task will require solutions to multiple sub problems, such as pre-processing, face detection, feature extraction and decision theory. From an input image of an unknown face, your program should detect the face, extract relevant features and make a decision if the face in the image belongs to a known person and, if so, to whom?

The 16 faces in the folder ‘db1’ belong to persons who should have access to the system (see fig. 1). For faces belonging to this group, the identity number for the image should be returned. For example, your program should return the number ‘2’ for the image ‘db1_02.jpg’. All other persons should be denied access, returning the number ‘0’. The folder ‘db0’ contains examples of 4 faces that should be denied access, which can be used for testing your program. Notice, however, that these 4 faces are only examples of unknown persons, and that your system should deny access (i.e. return ‘0’) for **all faces** beside the persons in ‘db1’.

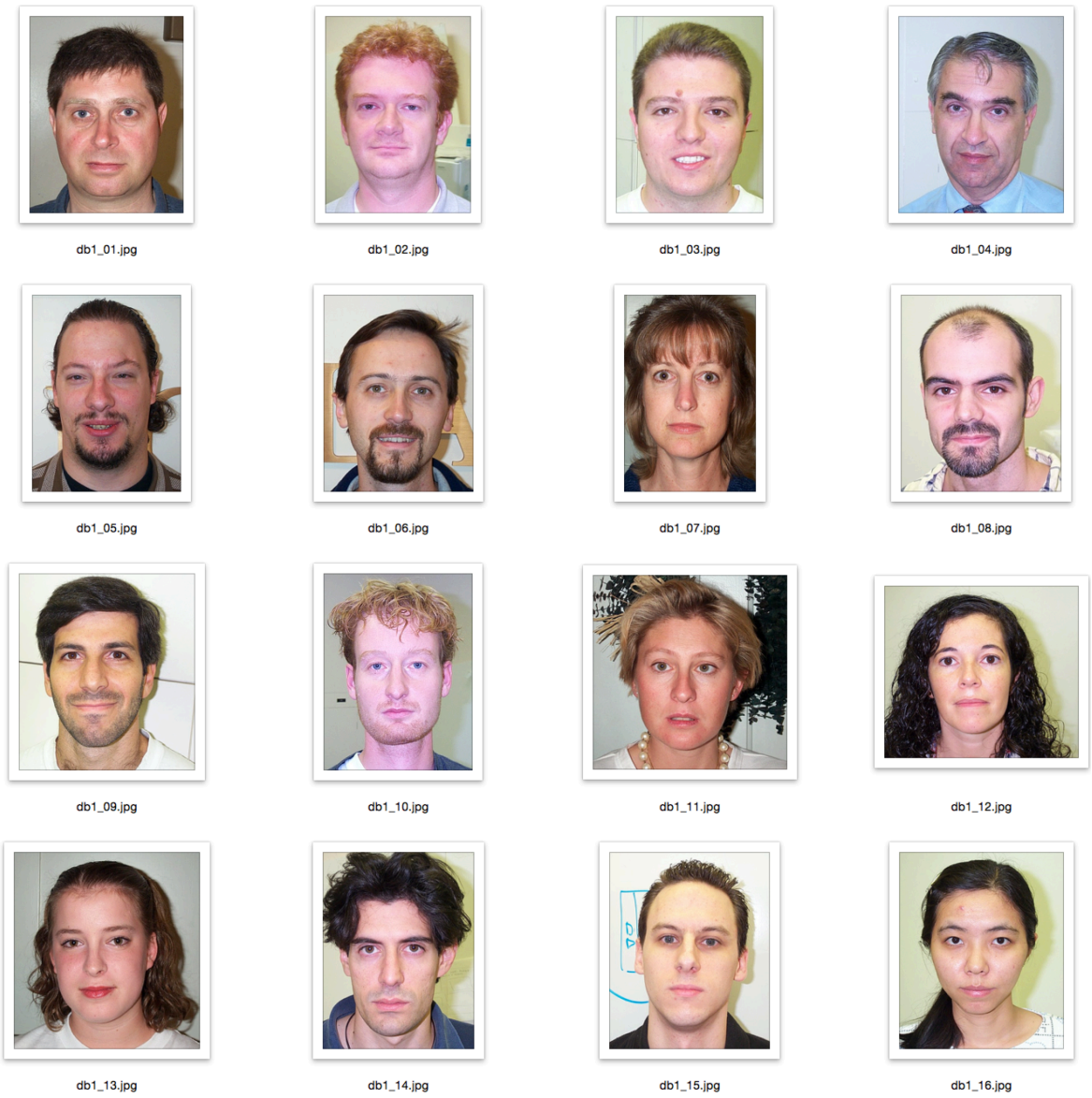


Figure 1. The 16 faces in the folder 'db1', who should be identified and have access to the system.

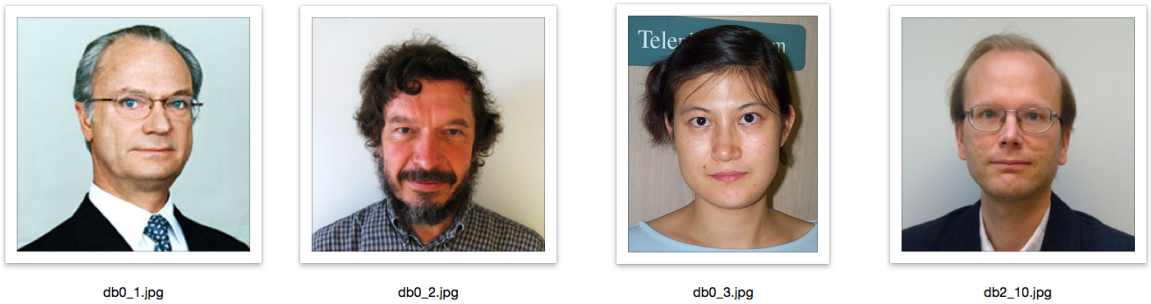


Figure 2. The faces in 'db0', who should not have access to the system (returning '0').

When testing your program for the examination, the facial images will be deliberately modified, to different levels of difficulty. The modifications will include translation of the

face in the image, rotation (max +/- 5 degrees), scaling (max +/- 10%) and change of tone values in the image (max +/- 30%). Examples of such modifications are displayed in fig. 3. Your program will also be tested using unknown faces, not included in 'db0'. To test the robustness of your implemented solutions it is therefor recommended that you test your program also for modified versions of the images, and for additional images of unknown faces. This is the minimum requirement for passing the course, i.e. for grade 3.

Figure 3. Examples of modifications of a facial image. From left to right: original, translation, rotation, scaling, tone value variation, and combinations of all.

For grade 4 and 5, your code should also work for other images of the same persons as in 'db1', captured under slightly different conditions. Figure 3 shows examples of images of 'person 1', with an unfocused (blurred) face (prefix: 'bl'), a more cluttered background ('cl'), with a different facial expression ('ex') and captured under different illumination properties ('il'). All of these different conditions introduce challenges that need to be addressed, either in the face detection, or in the recognition part, of your implementation.

Figure 4. Images of 'person 1' captured under different conditions, all available in 'db2'.

In 'db2' you will find a set of images of some of the persons from 'db1' captured under different conditions, which you can use to test your code. The id-numbers for the persons correspond to the ones in 'db1'. When evaluating your implemented code for grade 4 and 5, also other images of the same persons, similar to the ones in the in 'db2', will be used.

Carrying out the project work

The work consists of reading applicable literature, testing ideas and finally carrying out the programming according to your own ideas. The lectures will provide an overview of the area of face recognition and some theoretic background related to different problems that need to be addressed in the project. The aim is to give you an understanding of the general problem and some ideas of different ways to approach it, not to tell you "how to do it". The approach to solve the problem must be decided by you and your group yourselves. There are very few restrictions on how to do it. The project group should have **four members maximum**. All group members should take part in the work in all aspects, and be able to describe the design of the software and the methodical approach.

In order to facilitate the testing of your program you have to use a certain functional form. Below you can find the skeleton of how your program should be designed:

```
%%%%%%%%%%%%%%%%%%%%%%%%%%
function id = tnm034(im)
%
% im: Image of unknown face, RGB-image in uint8 format in the
% range [0,255]
%
% id: The identity number (integer) of the identified person,
% i.e. '1', '2',…,'16' for the persons belonging to 'db1'
% and '0' for all other faces.
%
% Your program code.
%%%%%%%%%%%%%%%%%%%%%%%%%%
```

The program you write must be written using Matlab. In the Matlab environment you will find many valuable functions that you may call in your program. Other program libraries are not allowed. You can use any number of functions in your file, but the format of the calling function should adhere to the skeleton above. Our testing of your program will be done automatically in a batch run. Therefore your code should not include any pop-up windows or any other kind of interaction. Feel free to divide your code into several different functions, and to store pre-computed properties of known faces in .mat-files, which can be easily loaded when testing the program.

Besides the code, you should also deliver a report that contains a description of the problem, the chosen methodology for how to solve the problem, and a description of your program implementation. The report should be written in English or Swedish.

Course organization

All students registered to the course will have access to the course space on Lisam. Here we will share lectures notes and some relevant literature, and also announcements regarding the course. You will also find the data sets of images for training ('db1') and testing ('db0' and 'db2'), referred to in this instruction.

Under the course space on Lisam, it is also possible to discuss problems and solutions between the groups, and to share interesting references. Helping each other and discussing solutions to problems within the project is allowed, and encouraged. However, sharing and copying code between groups is not allowed.

Examination

The course will give 6 hp with the grades U, 3, 4 or 5, where grade 5 requires an optional individual oral exam.

The grades are given according to the evaluation of your report, the implementation of the software, and the performance of the implemented methods. The number of students in each group will also be considered during the evaluation. The following requirements must be fulfilled:

- **Grade 3**: A report describing your approach, including references to used methodology. Your program should work according the instructions for images provided in 'db1' with modifications, such as translation, scaling, rotation and tone values. Images of faces not included in 'db1' should return '0'.

- **Grade 4 & 5**: Your report should be more detailed and must further include discussions on alternative approaches to solve the problems. Your program should work also for other images of the same faces as in 'db1', captured under different conditions. This includes images available in 'db2', as well as additional images, not included in the training sets, but similar to the ones in 'db2'.

Students that fulfill the requirements will be given the opportunity to take an individual oral exam for grade 5. During the oral exam (approximately 15-20 minutes), you will be asked to explain your work, motivate the selection of methods and relate them to alternative solutions, and to answer questions to demonstrate a good understanding of the topic. The oral exam is given in English or Swedish.

Literature

The lectures will partly be based on selected research papers, which will be made available on the course space on Lisam. They are by no means the only background material for the project, but should rather be seen as a starting point for further exploration of the literature. Face recognition and face detection are large and active research areas. A quick search for "face recognition" on Google Scholar (http://scholar.google.com) returns 3 510 000 hits and "face detection" gives 3 190 000, so you will not have to worry about finding reading material.

Deadline

The program code and the report must be uploaded, using the submission function on Lisam, no later than **Sunday, December 10, 2017**. Don't forget to include the identities of all the members of the group.

Good luck!

Daniel Nyström (examiner)