Image Processing and Pattern Recognition Projects, Winter semester 2008/09

Course supervisor: Pawel Strumillo, PhD, DSc (pawel.strumillo@p.lodz.pl , room 216A)
Project supervisor: Aleksandra Królak, MSc (aleksandra.krolak@p.lodz.pl , room 206AB)

Guidelines:

- Each topic can be elaborated only by one group.
- Each group may consist of up to 2 students.
- Literature and consultation on any given topic is fully available.
- There are 10 topics which are going to be randomly assigned to student groups.
- Students may exchange topics or suggest an optional topic instead, but only after consulting the project supervisor and if the reason for the change is important.
- The students will obtain a set of test images for every project, unless specified otherwise. However, we strongly encourage students to work with their own images.
- Each method devised by a given group will be verified by the supervisor against a second set of images (unless specified otherwise).
- The end result of every project is:
  - a working method, as specified in the project topic
  - a report summarizing and concluding the work performed
  - a presentation given at a special session in the end of the current term
- There will be an additional mid-project meeting to verify your progress and ensure that the groups are properly approaching a given topic.
- In case of projects that are required to be elaborated using the Intel OpenCV open source library, students are free to choose the programming environment, depending on their prior experience (both MS Visual C++ and Borland C++ Builder are accepted, other environments must be consulted with the project supervisor)

Deadlines:

1. Midterm consultations on progress of the projects– 7-12 December 2008
2. Projects presentations and submissions of reports – 14th January 2009 (those who submit the reports earlier can get their final marks on the 14th)

Project topics

1. **imageJ software package for processing images**

Difficulty: *,***
Tools: imageJ software package (available at http://rsbweb.nih.gov/ij/)

ImageJ is a public domain Java image processing program package. It can be run in two modes: as an online applet or as a downloadable application on any computer with Java. imageJ can display, edit, analyze, process, save and print different image file formats and supports standard image processing and analysis functions. It is design with open architecture in mind and allows definition of new user functions by Java plugins.

The aim of the project is to provide trials of the imageJ, indicate its advantages and limitations in image processing tasks. Ambitious project should contain development of user developed Java plugin (for solving a chosen image processing/analysis application) to be run within imageJ.

2. **Pedestrian tracking in image sequences**

   Difficulty: ***
   Tools: OpenCV

   The aim of the project is to write an image analysis procedure for tracking a pedestrian walking along a predefined path. The images are to be registered by a stationary camera fixed to the wall of a campus building, hence the tracked path should be corrected for perspective distortions. From the tracked path an instantaneous walking speed, path line and walked distance for a predefined time interval should be computed.

3. **Analysis of stereovision sequences**

   Difficulty: ***
   Tools: C++, OpenCV, stereovision library

   Stereovision is an image acquisition technique allowing for reconstruction of 3D structure of a scene. This imaging technique is also used e.g. for controlling robots mobility. You will be given a stereovision sequence of a corridor scene (of known dimensions) shot by a stereovision rig moving along a straight line. Your objective would be to work out the path of the rig in terms of the angle of its movement versus corridor walls, and e.g. sound an alarm if the rig is close to impact with the wall.

4. **Automatic background removal in video**

   Difficulty: **
   Tools: OpenCV

   In television and movie industry a so-called “bluebox” technology is used, when an object in the foreground (e.g. a person) needs to be placed on a different background (e.g. a computer-generated 3D scene). However, it is often not economically justifiable to build a “bluebox”, therefore other methods can be used to extract the foreground object and remove or replace background. Your task will be to develop a method for automatic background removal (with a calibration procedure) that will work on a live video feed from a webcam. Your method should not severely affect the frame rate of the video transmission.

5. **Person authentication based on ear biometry/hand geometry**

   Difficulty: **
   Tools: OpenCV/Matlab

   Biometrics is the term describing the set of methods for uniquely recognizing humans based upon one or more intrinsic physical or behavioral traits. Most commonly used procedures include face recognition, fingerprint identification, retina scan, iris scan, signature and voice analysis. However, two other biometrics can be used: ear and hand geometry. Your task is to collect a database of ear/hand images and develop a method for extracting such features from ear/hand images that will allow for person authentication. The developed algorithm should also include input image normalization (e.g. size, rotation). Two groups can be assigned for evaluation of this topic: one for ear-based person authentication and the second one – for hand-based authentication.

6. **Matrix code (2D bar code) reader**

   Difficulty: ***
   Tools: OpenCV

   Everyone knows the linear barcodes present on almost all products in the shops. In recent years the new type of bar codes were introduced – the matrix codes (2D bar codes). Your task is to work out a procedure for reading the matrix codes generated for 9-digit numbers (like phone numbers). You can find the Datamatrix Generator here: http://datamatrix.kaywa.com. The developed method must deal with different lighting conditions, rotations and perspective distortions.

7. **Font recognition**

   Difficulty: **
   Tools: OpenCV/Matlab

   The word editors offer a large set of different fonts. Sometimes it can be hard to define what font was used in the document we want to “clone”. Your task is to work out an algorithm recognizing the type of font in the image. Your method should recognize at least 5 types of fonts, including bold and italic. It should also deal with different sizes of text.

8. **Head-movement controlled human-computer interface**

   Difficulty: **
   Tools: OpenCV

   Among us there are people form whom operating a computer mouse or a keyboard is impossible due to their disabilities. You can help them by creating an application allowing for hands-free communication with the computer. Your task is to develop a method for detecting and interpreting certain head movements in order to browse through the image gallery, change the sizes of the photos, rotate them... Feel free to add other functionality of the application.

9. Generation of motion blur for image elements

Difficulty: ***
Tools: OpenCV

It is easy to take a photo of a moving object and achieve the effect of a motion blur. But what about the motion animation? How to get an image of a synthetically generated scene with the effect of capturing moving objects? Your task is to develop a method for simulating motion blur for particular objects in the image.

10. Bone detection in the carcass

Difficulty: ***
Tools: OpenCV/Matlab

Image segmentation can have many applications, sometimes quite surprising. One of them is detection of bones in the images of carcass. Your task is to develop a procedure for extracting the contours of the vertebras in the given set of carcass images and assessing the size of the detected bones. Your method should deal with different lighting conditions.

11. Mouse movement metering software

Difficulty: **
Tools: OpenCV/Matlab

When you work at the computer you don't notice that your hand covers a really large distance when using a mouse. Your task is to work out a method that will detect and track your hand carrying a computer mouse. The output of your program should show the user what is the distance that he makes while using his computer mouse. Use a webcam placed above the mousepad (the exact size of the mousepad is known) to analyze the distance frame by frame. For detecting the hand use either skin segmentation algorithm or write a procedure finding a coloured tracking point placed on the top of the palm.