Practical Assignment 2
Marker-based Interaction & Face-based interaction

Interaction Technology Course, 2013-2014

Start Date: 10/9/2013
Deadline: 23/09/2013, at 23.00

Requirements:

1. Hardware: computer + webcam.
2. Software: Microsoft Visual Studio 2010 C#, EmguCV.
3. **Your submitted solution must compile and run on the workstations in the practical rooms**, otherwise your submission can not be graded.
4. In order to ensure the needed libraries and references for compiling and running on the workstations are in your solution, we strongly encourage you to use the empty project provided by us as a starting point. The empty project is provided in the practical2.rar file (available on the course website in the Lab Schedule section) and can be found in the ”Empty Project - Practical2 - Face Detection” folder.

Instructions:

Part 0: (optional)

1. If you want to install EmguCV, visit: http://www.emgu.com
2. Learn how to use the basic commands in EmguCV by compiling, running, modifying the examples given on http://www.emgu.com/wiki/index.php/Code_Gallery.

Part 1: (Webcam)

Write a program that:

1. Captures the video from the webcam using the Emgucv functions.
2. Displays the video.
3. Writes the captured videostream to a file and load this file and display the video.

Part 2: (Face detection and interaction)

Write a program to detect a face in the webcam capture using the Viola-Jones algorithm (see the section "useful EmguCV information" below), according to the following instructions:

Create a window displaying the webcam capture and divide it into 4 rectangular regions.

Identify the location of the detected face and change the color of the region's border in which the face is detected to red.

Part 3: (Marker-based Interaction)

Implement the following scenario:

1. Take a piece of paper with a round shape, a distinct color (for example completely red, or green, or blue) and put it on your finger. This will function as your marker.
2. Wave your finger in front of a white background and write a program using EmguCv functions that will perform the following functions in real-time:
    a. Identify the location of the marker (to detect and track the marker).
    b. Change the color of the marker to black (or any other color that is different from the paper's original color), and display it in real time.

Useful EmguCV Information:

1. Image(TColor, TDepth).And Method
   can be used to merge or overlay two images, for further documentation, see: http://www.emgu.com/wiki/files/2.3.0/document/html/0c9448a1-43a9-963c-65b3-1a0b74019f97.htm
2. Image(TColor, TDepth).Draw Method
   Can be used to perform drawing, for further documentation, see: http://www.emgu.com/wiki/files/2.3.0/document/html/c4b3f4fd-ed41-8e05-dff9-f82450195ca5.htm http://www.emgu.com/wiki/index.php/Working with Images#Drawing Objects on Image
3. Image(TColor, TDepth).Convert Method
   Can be used to convert Images from and to Images with specific properties, for further documentation,see: http://www.emgu.com/wiki/files/2.3.0/document/html/b36c28ee-7dc5-edd7-97c8-894f244e5cb7.htm http://www.emgu.com/wiki/index.php/Working with Images#Generic Operation
4. Image(TColor, TDepth).DetectHaarCascade Method

Needed for implementation of Viola-Jones algorithm, for further documentation, see:
http://www.emgu.com/wiki/files/2.3.0/document/html/a9800794-f3d3-ed32-bab7-92a8cc1cff88.htm

5. HaarCascade Class
   Needed for implementation of Viola-Jones algorithm. When initializing a HaarCascade object, use the .xml file provided in the empty project (haarcascade frontalface default.xml) for initialization.
   For further documentation, see:
   http://www.emgu.com/wiki/files/2.3.0/document/html/8071f5d0-e119-4ba7-308c-0002cb8fb90a.htm http://www.emgu.com/wiki/index.php/Face detection
6. Complete EmguCV API:
   Contains all the Classes of EmguCV, see:
   http://www.emgu.com/wiki/files/2.3.0/document/Index.html

Practical tips

1. Sometimes when using EmguCV camera captures, the captures produced will be lagged and contain noisy blocks. This is due to a program that is started when the webcam is accessed. It's called ffdshow and should be turned off at all times. When such captures are encountered, proceed as follows:
   a. don't close your VS2010 build, and click on the ffd icon which has just appeared in the system tray.
   b. choose the manager option, choose the Directshow control option on the left, and select the option "only use ffdshow for the following programs".
   c. apply new settings, close ffdshow, and re-build the project.
   d. At this point, ffdshow will ask to be started again. Choose the option "No".
2. In certain cases Microsoft Visual Studio will not be able to build your project, complaining about a certain process which is already running. In this case, copy your project folder form your network share to the local drive of the workstation, this should stop this error from occurring. Don't forget to copy your folder back to your network share when you're finished, otherwise you won't be able to access it from another workstation.
3. The provided Empty project is for use on 32 bits machines, as is the case in the practical rooms. If you want to use these solutions on a 64 bit

environment, choose from the Debug dropdown menu (right next to the build button) the option 'configuration manager' and select for platform 'x86'.

To Submit:

1. The project files, containing all necessary code and libraries (dll files).
2. The executable programs.
3. Names and student numbers of the group members.
4. Use the Submit system, located at http://www.cs.uu.nl/docs/submit/ , to submit your solution. The maximum allowed file size of your submission is 40MB. Note: if the size of the videos is larger than 40MB, put them in your website, and submit only the link. This applies only for videos. Other files must go through the official Submit system.

Grading Criteria:

1. Only part 2 and 3 will be part of the grading.
2. If the code:
   (a) performs correctly as instructed: 8
   (b) performs properly but crashes now and then, or is not fully functioning: 5-7
   (c) performs correctly and has features beyond the instructions: 9-10
   (d) otherwise: 0-4.
3. The grade also depends on the completeness of the submission (as mentioned in ”to submit”
   above).
4. Questions or other issues related to the practicals (including late submissions) should be sent to: interactiontechnology2013@gmail.com
5. If you find that any members of your group do not work properly, report this to us. We will reduce their grade, and might increase yours (depending on the circumstances).