*The OpenCV Assignment in the class of Sensor 2017*

Student Number: 1526708

Author: Masashi Kaneko

**Introduction**

OpenCV (Open Source Computer Vision Library) is a widely used software library specifically for computer vision and machine learning. The library has more than 2500 optimized algorithms and they can be used for several purposes, such as face detection/recognition, object classification, video tracking, etc. It supports major environments (Windows, Linux, macOS, Android) and interfaces (C++, C, Python, Java, MATLAB).

**Purpose**

There are two goals for this assignment and they are accomplished by writing a OpenCV included C++ code.

* To record the face size data when the face stays 1 m away from the camera.
* To calculate the distance between the camera and the face by using the data collected in the previous step.

**Environment**

OS : Ubuntu 16.04 LTS

Compiler: g++ 5.4.0

OpenCV : version 3.2.0

**Programs**

assignment/

|- main.cpp

|- Makefile

|- haarcascade\_frontalface\_alt.xml

|- video.mp4

The directory includes four files as shown above. “main.cpp” file is a source code that contains all the processes for this assignment. “Makefile” is simply for easier compilation. “haarcascade\_frontalface\_alt.xml” is a build-in cascade classifier file copied from OpenCV directory. This is used for classifying faces and there are other classifiers such as “haarcascade\_eye.xml” in the OpenCV directory. “video.mp4” is a video file. (I tried to use a build-in camera on my laptop for real-time processing. However, the main code could not detect the camera as a usable device. This problem is probably due to the Linux system itself. Unfortunately, I did not have enough time to solve this issue, hence I used a video file for the purpose).

**Z-transform for Pinhole Camera Model**

hi: The object’s height projected on the virtual screen.

f: The constant distance between the lens and virtual screen.

h: The actual size of my face, which is measured manually.

d: The distance between the lens (camera) and my face.

The formula of this problem is simply,

hi/f = h/d

Before calculating d, the value f should be solved by initially given constants.

h = 0.25 m

d = 1 m

hi = 170 pixel

f = hi \* (d / h) = (170 pixel) \* ( (1 m) / (0.25 m) ) = 680 unit

After that process, d can be solved by the constants (h, f) and data from camera (hi),

d = h \* (f / hi)

**Result**

The result video is available on YouTube. As shown in the video, the OpenCV function of face detection seems working perfectly and the distance from the camera to the face is calculated.

**Conclusion**

The data gathered from the experiment is probably useful for implementing a test, in which compares the calculated distance and the actual distance. However, measuring the actual distance in precise was a tough task to be done. Because there are no methods or systems to measure the real distance.

**References**

[**http://www.opencv.jp/sample/object\_detection.html**](http://www.opencv.jp/sample/object_detection.html)