# EE6130: Advanced Topics in Signal Processing (Jan-May 2021)

## Computational Imaging & Displays

### Assignment 3 : Optical Glyph Tracking

**Deadline: May 1st, 2021**

In this assignment, you learn how to render a virtual 3D object on live images captured from a smart-phone camera to understand the basics of augmented reality. Detailed instructions regarding the steps can be found below. A detailed report along with the final augmented video have to be submitted in a zip file, apart from the well-commented codes.

**References:**

1. Optical glyph tracking (see the paper given)
2. Camera calibration : Chapter 7 for Camera Matrix basics and Section 8.8 for estimating the camera matrix of the book: Multiple View Geometry in Computer Vision by Hartley and Zisserman.

**Problem Statement:**

Given a 3D object in the form of (x,y,z) positions, you are supposed to use your smart-phone camera to capture a video of an optical glyph in different orientations and project/render the 3D object on the video frames. The optical glyph will act as a reference plane. You are supposed to detect the orientation of the glyph with respect to the camera and project the 3D object on the reference frame according to estimated orientation.

**Procedure:**

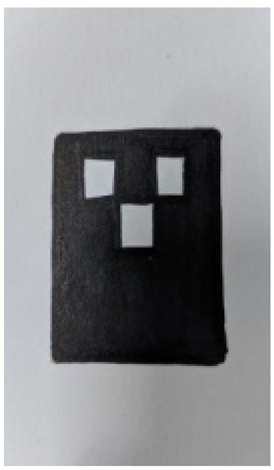
**Camera Calibration:**

You have to estimate the camera calibration matrix for the smart-phone camera you will be using to capture the optical glyph. This can be done using different cues, as mentioned in Section 8.8 of Zisserman's book. A standard way is to use **check-board patterns to calibrate the camera**. You can use any existing software to calibrate, given that the images should be captured by your smart-phone and you understand the basic theory.

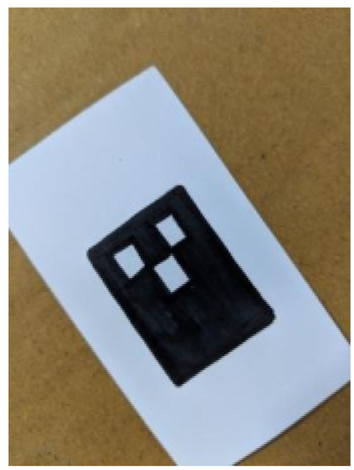
Mention the software you have used in the report and include 2-3 captured checker-board images displaying the detected features.

**Detect Pose:**
In order to continuously project a 3D object on the moving reference plane, we need to find the relative "movement" or pose of the reference plane in the current frame with respect to a base orientation of the reference plane. For example in figure given below, the first images corresponds to a reference image of an optical glyph, whereas the second image could be a random image from the video. To estimate the pose of the glyph in the random frame, we need to find the camera pose. This can be done using homography. Any existing libraries can be used for this task.

Mention the library you have used in the report and include 2-3 frames along with their matching points with the reference frame and the final homography/pose matrices



(a) Reference image



(b) Random frame

**Project the 3D object:**
Using the pose estimated for the corresponding frames, now you have to project/ render the 3D object by performing a perspective transform on the object according to the estimated pose and the camera matrix and render the 3D object into the existing 2D frame. The 3D objects are given as .obj file, description of which can be found here:
**https://en.wikipedia.org/wiki/Wavefront_.obj_file#File_format**

A sample script to load the .obj file can be found here:
**http://www.pygame.org/wiki/OBJFileLoader**

You need to submit a video of the augmented object on the captured video of the optical glyph. Note, you can reduce the spatial resolution of the frames and also the fps to reduce the computational complexity. Mention the details in the report.