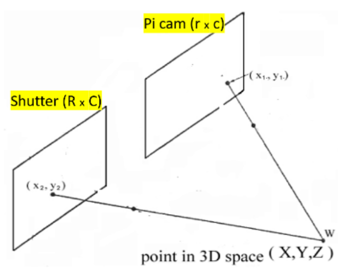
# Mapping Camera Coordinates to a Shutter Coordinate System


## Abstract:

In this project, a generalized theory for mapping from camera coordinates to a shutter (LCD Shutter) coordinate systems has been proposed. To prove the concept, a mathematical model has been shown for a particular camera (240x320) and LCD shutter display (320x240) setup. In the setup, the camera has been placed to the next of the LCD shutter without any rotation angle and by capturing image of the LED which is focused to the shutter, the coordinates of the LED are mapped to the shutter. By doing the calibration for nine different points of the shutter, it is shown that the mapping is accurate.


## Problem Statement:

They idea is to find out the coordinates of a particular location in a shutter (real world) coordinate system based on the information from a camera system. This mapping is necessary to run automatic area selection algorithm in visible light communication systems.




A VLC System

## Approach:

A Pi Camera has been placed to the right next of the LCD shutter with zero degree rotation. Using object detection algorithm, the LED locations are identified as pixel coordinates. Now, this pixel coordinates of the camera has been mapped to the shutter. The following steps I have followed to find out the pixel coordinates on the shutter coordinate:

- Convert pi cam image (r x c) to (R x C) shutter image
- Find out the pixel difference along X [dx] and Y axis [dy] (Translation vector Tx and Ty)
- Calibrate the camera for at least 9 points and get the new coordinates of these 9 points.
- Based on this calibration, find out ΔX and ΔY which are resolution difference along rows and columns.
- Get the new resolution (R' x C') for the shutter image
- Now, calculate the each coordinates on the shutter image based on this following equation:

$$\mathbf{X}_{s\,(R' \times C')} = \Delta X + \mathbf{X}_{s\,(R \times C)} / R * R' \qquad \mathbf{X}_{s\,(R \times C)} = [\mathbf{X}_{s\,(R' \times C')} - \Delta X] * R / R'$$

$$\mathbf{Y}_{s\,(R' \times C')} = \Delta Y + \mathbf{Y}_{s\,(R \times C)} / C * C' \qquad \mathbf{Y}_{s\,(R \times C)} = [\mathbf{Y}_{s\,(R' \times C')} - \Delta Y] * C / C'$$

**Results**:

In this work, I have demonstrated the accuracy results of my proposed mapping theory. Here, the feasability study has been demonstrated.

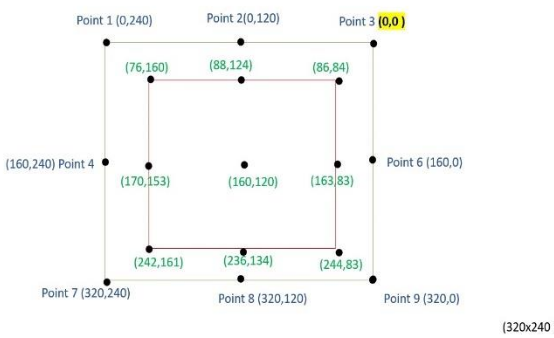
The first step is to determine the pixel difference along X and Y axis (dx, dy) by focusing the LED at the centre of the shutter.

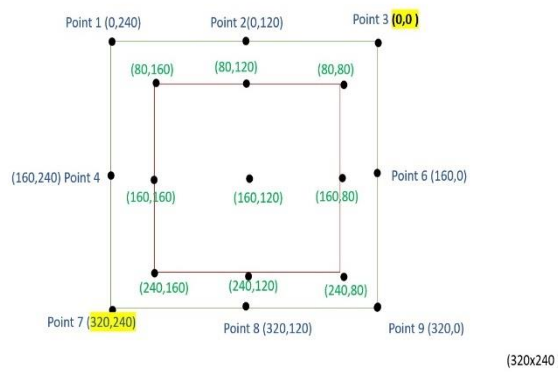

From calculation, dx= 160-156= 4    dy= 120-46= 74

In pixel coordinates system, for 240x320: (117,61)

In 320x240 , it would be (156,46)

After selecting calibrating points and applying proposed steps of mapping, I found the following results:




Now, we get a 160x80 resolution, in which these 9 points are verified. So, the new coordinates equation would be,

$$Xs\ (160x80) = 80 + Xs\ (320 \times 240) / 320 * 160$$

$$Ys\ (160x80) = 80 + Ys\ (320 \times 240) / 240 * 80$$

$$\mathbf{X}_{\mathbf{s}\ (320 \times 240)} = [\ \mathbf{X}_{\mathbf{s}\ (160x80)} - 80\ ]*\ 2$$

$$\mathbf{Y}_{\mathbf{s}\ (320 \times 240)} = [\mathbf{Y}_{\mathbf{s}\ (160x80)} - 80\ ]*\ 3$$

Here,
ΔX = 80 [resolution difference along rows]
ΔY= 80 [resolution difference along columns]

Xs and Ys represent number of rows and columns in pixel coordinates

References:

1. Computer Vision (Csc 8980) Slides from Dr. Ashwin's lectures.

Github link: The object detection script has been uploaded in the following link:

https://github.com/rashedrimuz/CV_8980_Team-7

Submitted By-

MD Rashed Rahman