Submitted by:

(101683035) Akshay Sharma, BE Third Year, CSE
(101683033) Abhi Mahajan, BE Third Year, CSE
(101503008) Abhishek Sharma, BE Third Year, CSE

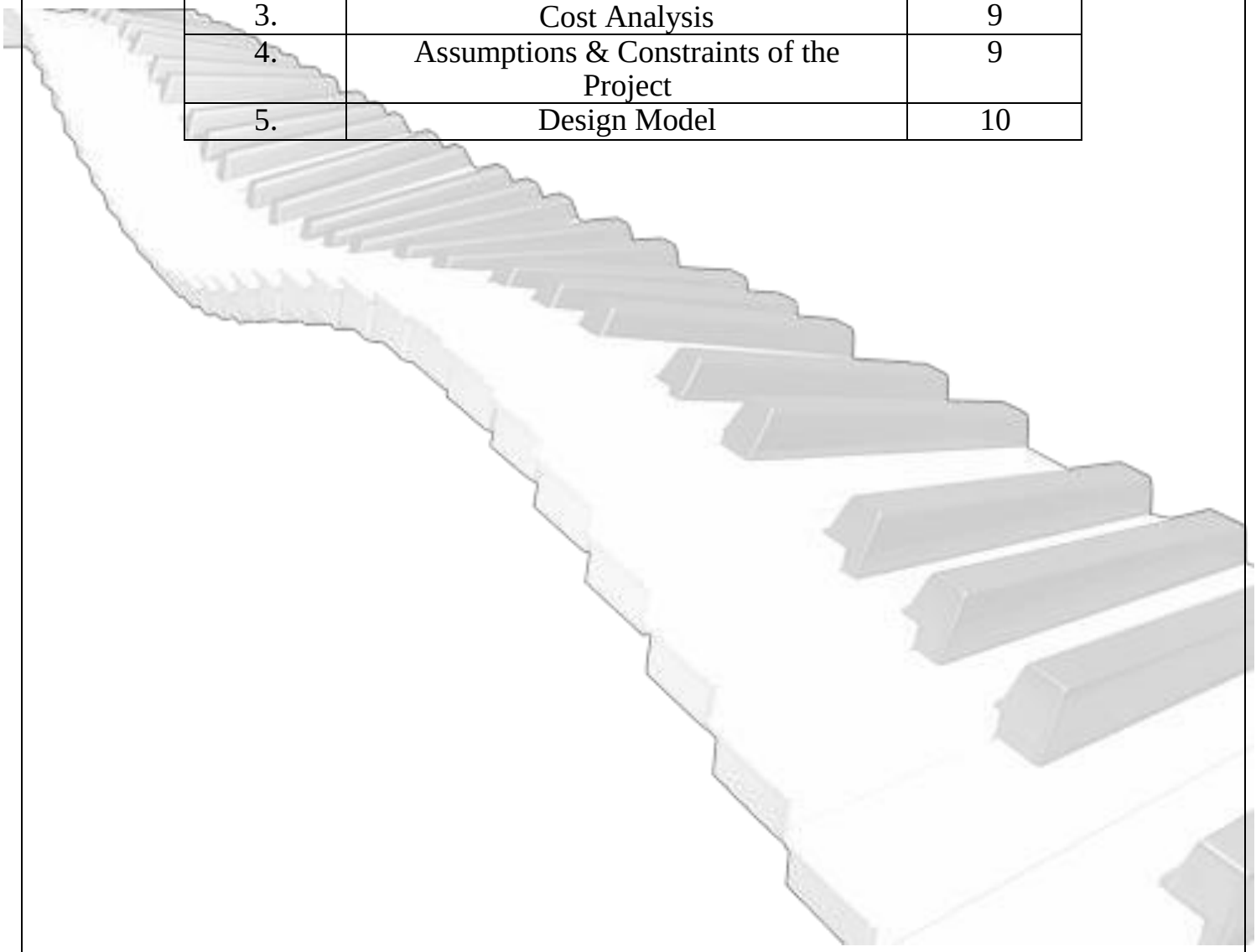
Project Team No. – CPG 84

Under the Mentorship of

Dr. Singara Singh
Assistant Professor
CSED
Thapar Institute of Engineering and Technology, Patiala

COMPUTER SCIENCE ENGINEERING DEPARTMENT
THAPAR INSTITUTE OF ENGINEERING & TECHNOLOGY, PATIALA
May, 2018

Index

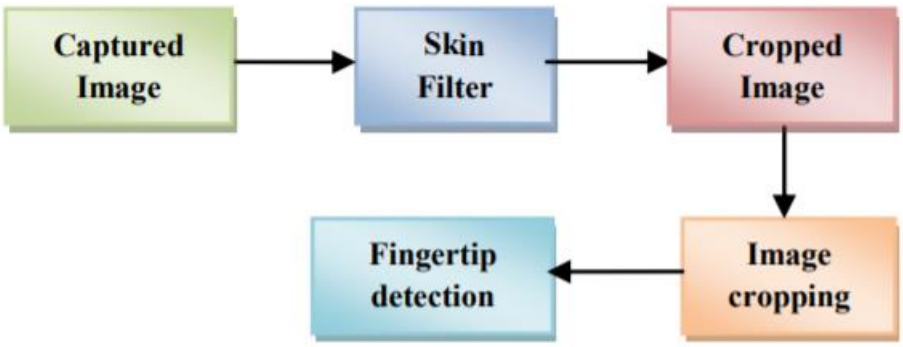


Analysis/Working Principle

1. Introduction

Hand Gesture Recognition (HGR) is a very popular and effective way used for the human machine communication. It has been used in many applications including embedded systems, vision-based systems and medical applications. In HGR, fingertip detection is an important part if image base models are being used. HGR systems face many problems in skin segmentation due to luminance and intensity in images. The fingertips detection models mostly have assumption about the hand direction; this restricts the natural expression of humans. Processing time is another key factor in image-based processing algorithms.

Here we are focusing on direction invariant fingertip detection of natural hand with real time performance. This work requires no glove, sensors or colour strips to detect the fingertips. The only assumption is that user will show the hand to system, facing the palm to the system while the direction of hand is not restricted. User is free to show hand in any direction as naturally hands move. This paper also presents a figure cropping method based on hand size, which will fasten the further process as the processing pixels would be reduced after cropping.



2. Background work

A lot of work has been done in this area for dynamic hand gesture recognition using fingertip detection. A survey on fingertip-based methods could be found out in. There are several limitations in existing approaches. Garg used 3D images in his method to recognize the hand gesture, but this process is complex and also not time efficient. The Processing time is one of the very critical factors in real time applications. Aznaveh presented an RGB vector-based method for skin detection in images. Yang analyses the hand contour to select fingertip candidates, then finds peaks in their spatial distribution and checks local variance to locate fingertips. This method is not invariant to the

orientation of the hand. There are other methods, which are using directionally Variant templates to detect fingertips. Few other methods are dependent on specialized instruments and setup like the use of infrared camera, stereo camera,
a fixed background or use of markers on hand. This paper describes a novel method of motion patterns recognition generated by the hand without using any kind of sensor or marker.

The fingertips detection should be near to real time if it is going to process video. Generally, image-based models work on pixel by pixel and do hand Segmentation and work only on region of interest. However, most hand segmentation methods can't do a clearly hand segmentation under some conditions like fast hand motion, cluttered background, poor light condition. If the hand segmentation is not valid, then detection of fingertips can be questionable. Researchers used infrared camera to get a reliable segmentation. Few researchers in their work limit the degree of the background clutter, finger motion speed or light conditions to get a reliable segmentation. Raheja and few others also used 3D mapping using specialized device like KINECT for hand segmentation. Some of fingertip detection methods can't localize accurately multidirectional fingertips. Researchers assumed that the hand is always pointing upward to get precise localization.

3. Approach to fingertips detection

Video is the sequence of the image frames at a fixed rate. First of all, Images would be captured with a simple camera in 2D continuously and would be processed one by one. An HSV colour space-based skin filter would be applied on the images for hand segmentation. An intensity-based histogram would be constructed for the wrist end detection and image would be cropped so that resultant image would have only hand pixels. The fingertips would be detected in the cropped hand image and would be marked differently.

3.1. Skin Filter

An HSV colour space-based skin filter would be used on the current image frame for hand segmentation. The skin filter would be used to create a binary image with black background. This binary image would be smoothened using the averaging filter. There can be false positives, to remove these errors the biggest BLOB (Binary Linked Object) is considered as the hand and rest are background. The biggest BLOB represents hand coordinates "1" and "0" to the background. The filtered-out hand image, after removing all errors. The only limitation of this filter is that the BLOB for hand should be the biggest one.

3.2. Wrist End Detection

To crop the image, we need to find out the direction of hand and wrist end. First of all, an intensity histogram of the binary silhouette would be formed. Histograms functions are: Here imb represents the binary silhouette and m, n represents the row and columns of the matrix imb.

$$H_x = \sum_{y=1}^{n} imb(x, y)$$

$$H_y = \sum_{x=1}^{m} imb(x, y)$$

To find the direction of hand, after a 4-way scan of image, we choose the maximum value of „on" pixels coming out of all scans ("1" in the binary silhouette). it was noted that maximum value of "ON" pixels represents wrist end and opposite end of this scan end would represent the finger end. Similarly, the green bar corresponds to right scan, red bar corresponds to down scan, and pink bar corresponds to up scan of „on" pixels in the binary silhouette. Now, it is clear that red bar has higher magnitude than other bars. So, we can infer that the wrist end is in downward direction of the frame and consequently the direction of fingers is in the upward direction. Here the direction from wrist to finger is known.

3.3. Hand Cropping

Hand cropping minimizes the number of pixels to be taken into account for processing which leads to minimization of computation time. In the histogram generated, it was observed that at the point where wrist ends, a steeping inclination of the magnitude of the histogram starts, whose slope, m can be defined as:

$$m = \frac{y2 - y1}{x2 - x1}$$

The scan would start from the end where skin pixel was found and move till an inclination would be found. The points correspond to the first skin pixel scanning from other three sides were found, crop the image at that points. The equations for cropping the image are:

$$imcrop = \begin{cases} origin_{image}, & for\ Xmin < X < Xmax \\ & Ymin < Y < Ymax \\ 0, & elsewhere \end{cases}$$

Where imcrop represents the cropped image, Xmin, Ymin, Xmax, Ymax represent the boundary of the hand in the image. Some results with processing steps for hand cropping. The arrows showed in the main frames indicate the direction of scan which was found from wrist end detection step. In all the histograms it is clear that at the wrist point, a steeping inclination starts in the scanning direction.

3.4. Fingertip Detection

At this point we have one smaller image which conation only skin pixels (hand gesture shape). We will figure out fingertips in the cropped hand image. Start scan the cropped binary image from wrist to fingers end and calculate the number of pixels for each row or column based on the hand direction in up-down or left-right. Assign the intensity values for each pixel from 1 to 255 in increased manner from wrist to finger end in equal distribution. So, each "on" pixel on the edges of the fingers would be assigned a high intensity value of 255. Hence all the fingertips contain pixel value 255. The fingertip detection can be represented mathematically as;

$$pixel_{count}(y) = \sum_{X=Xmin}^{Xmax} imb(x,y)$$

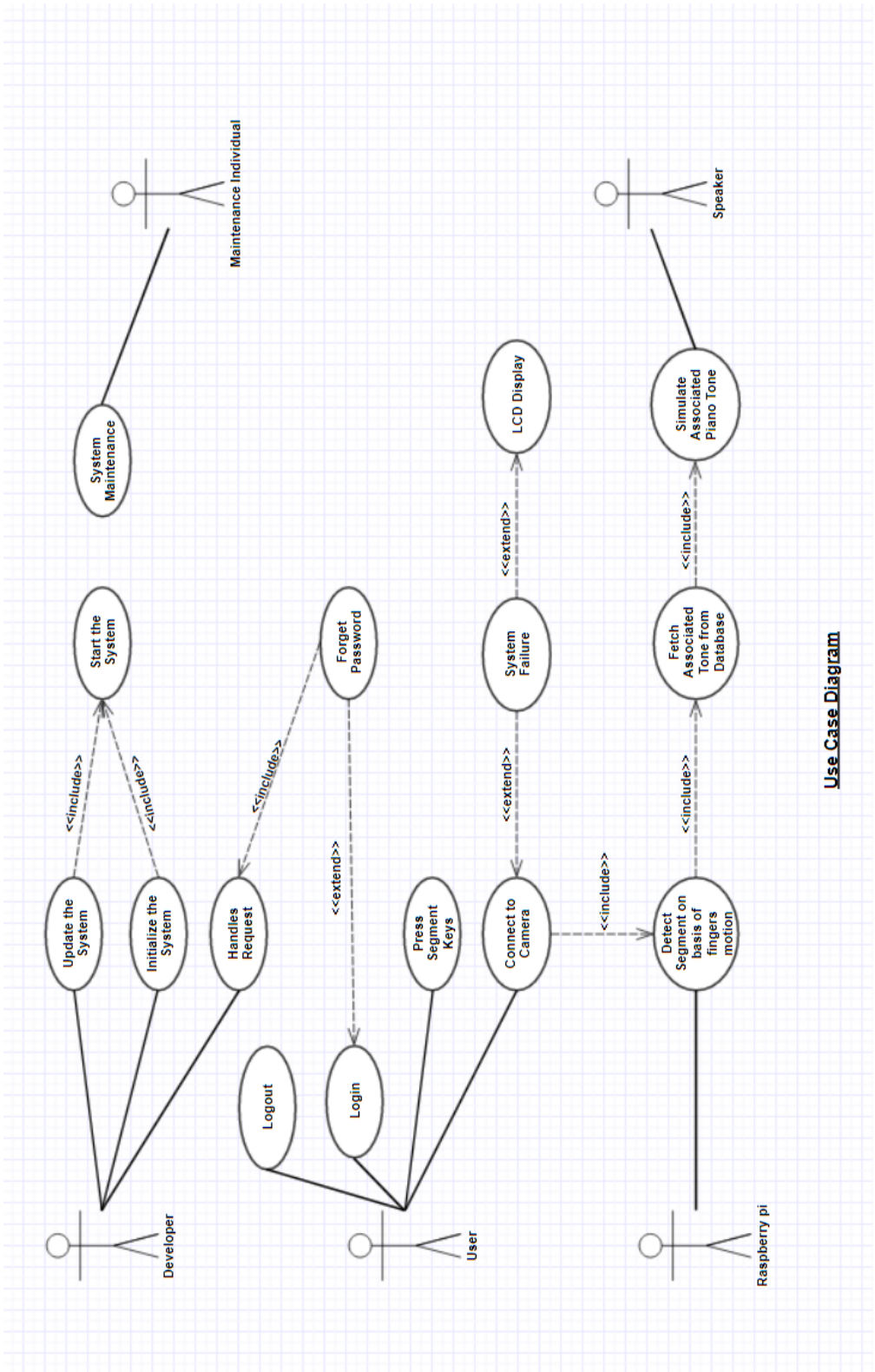
$$modified_{image}(x,y) = round(\, x * 255 / pixel_{count}(y)\,)$$

$$Finger_{edge}(x,y) = \begin{cases} 1 & if\ modified_{image}(x,y) = 255 \\ 0 & otherwise \end{cases}$$

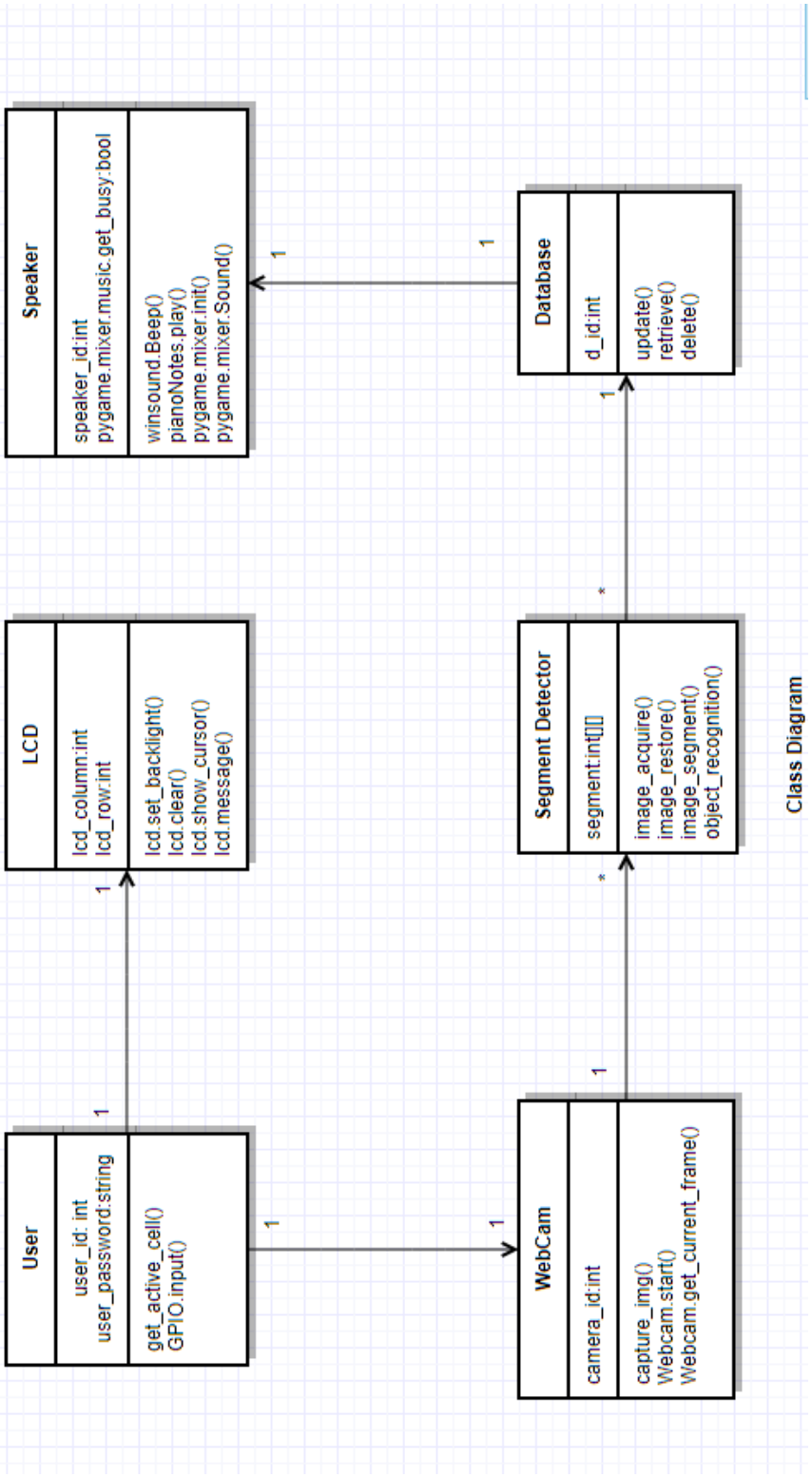
The line having high intensity pixel, is first indexed and check whether differentiated value lie inside a threshold, if it is then it represents a fingertip. The threshold value changes toward the direction of hand. That threshold can be set after the detection of the direction of hand to the finger which we already know. where detected pixels are marked with different colour. any direction, only palm should face the camera. The fingertips would be detected irrespective of user orientation. The Movement of user's finger will control the robot hand and its working, by moving hand in front of camera without wearing any gloves or markers.

SRS with Finalized Analysis Model

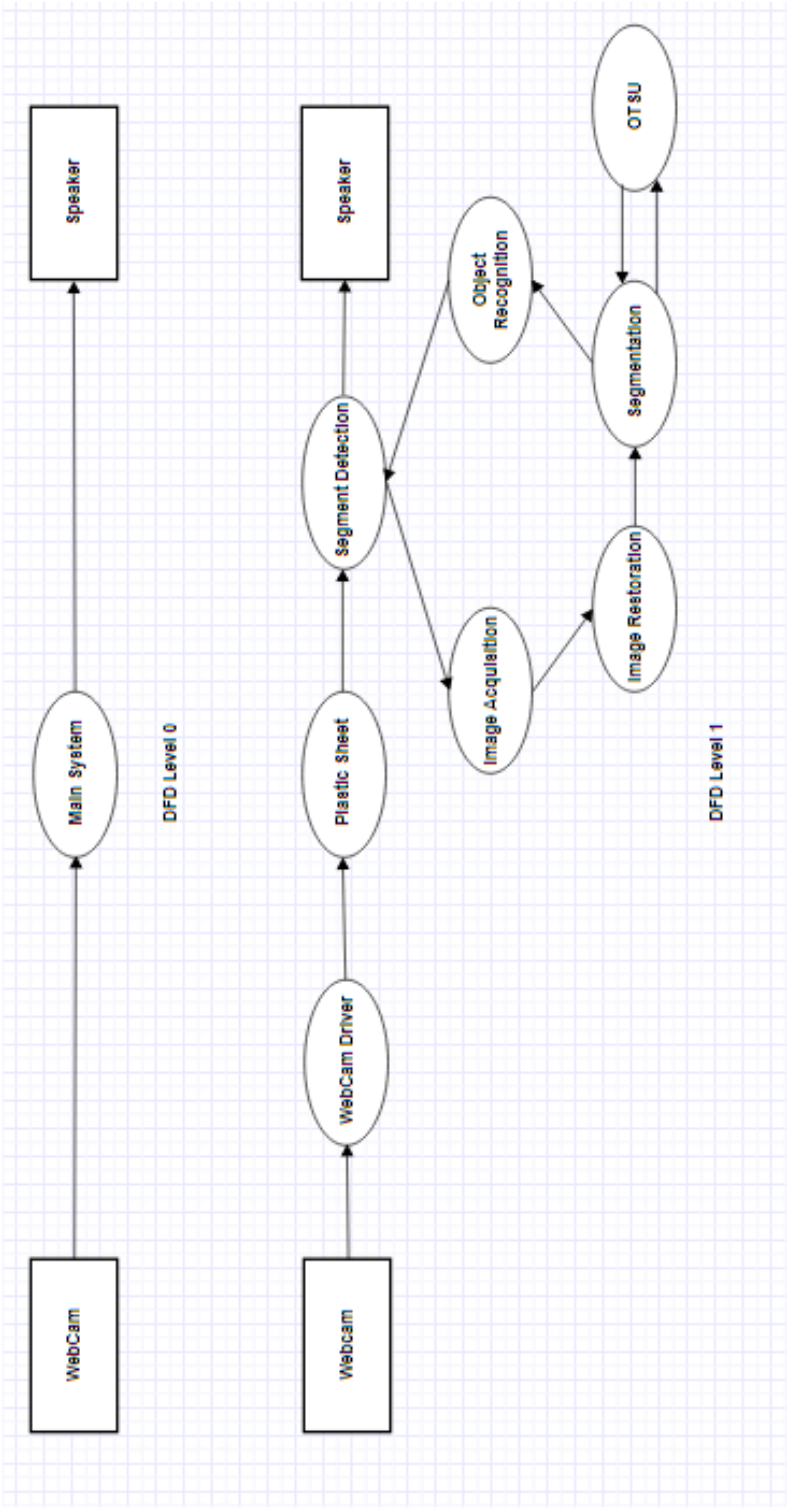
1. Use Case Diagram



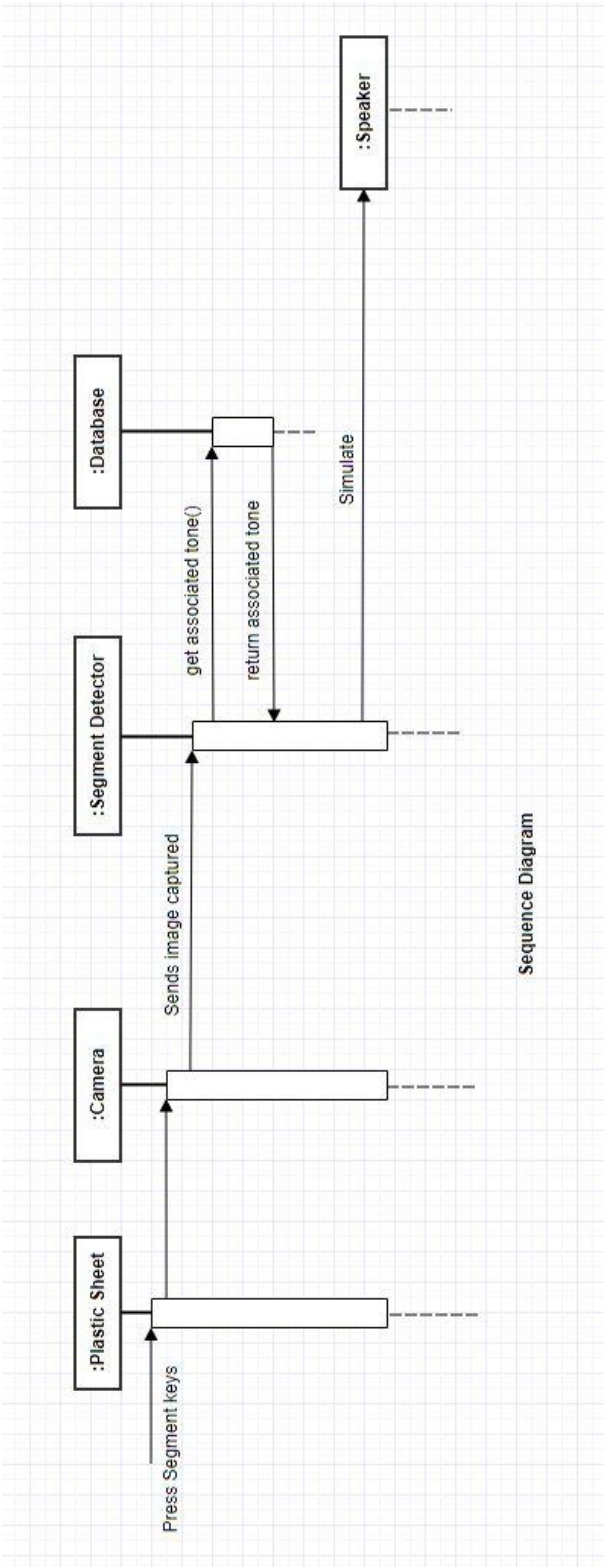
2. Class Diagram



3. Data Flow Diagram



4. Sequence Diagram



Cost Analysis

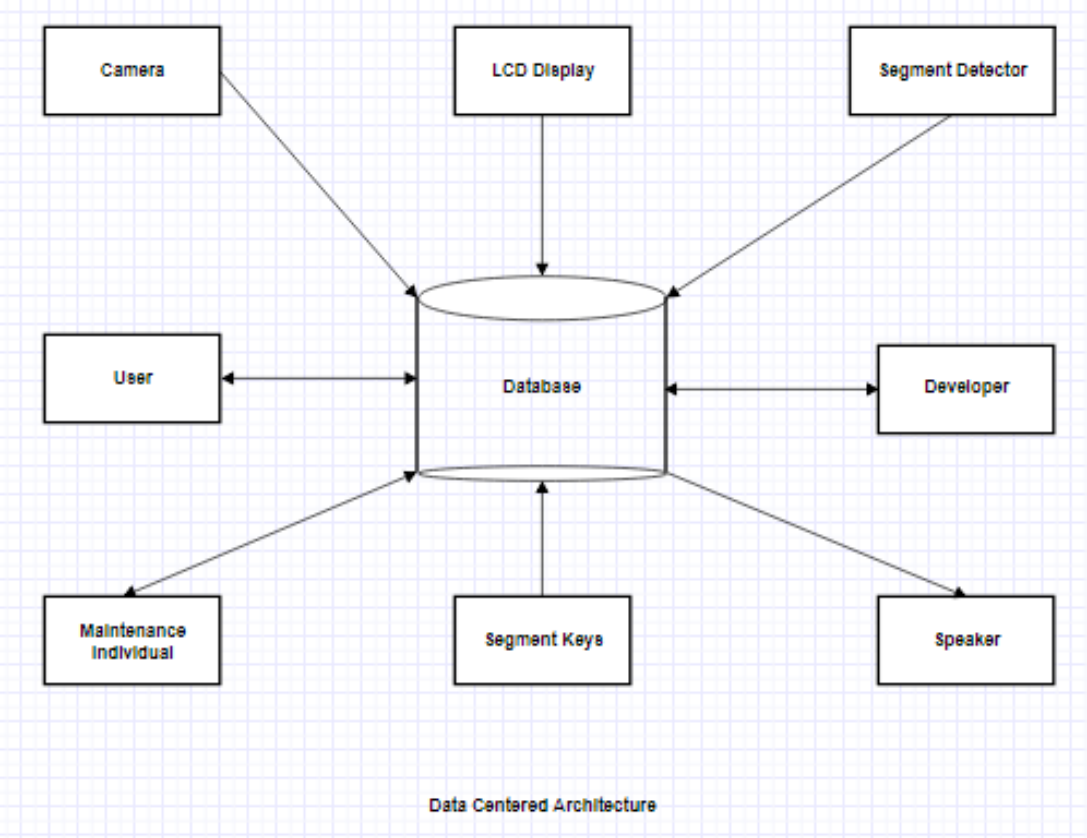
Tools Required	Cost(₹)
Raspberry Pi	₹ 3,649.00
PCB Board	₹ 185.00
Speaker	₹ 200.00
Web Camera	₹ 850.00
Soldering Iron	₹ 299.00
Digital Multimeter	₹ 250.00
LCD Display	₹ 1,410.00
Wire Stripper & Cutter	₹ 58.00
Miscellaneous(Soldering Iron,Plastic Sheet,Resistors & Capacitors)	₹ 399.00
Total Cost	₹ 7,300.00

Assumptions and Constraints of the Project

- Camera should be mounted properly
- The camera should be mounted in such a way such that it focusses on each segment properly
- The camera should be calibrated before the system starts
- The piano should be set up in bright lighting conditions
- The size of the piano will depend on the coverage area of camera
- The system should fulfil all the basic requirements needed for the project

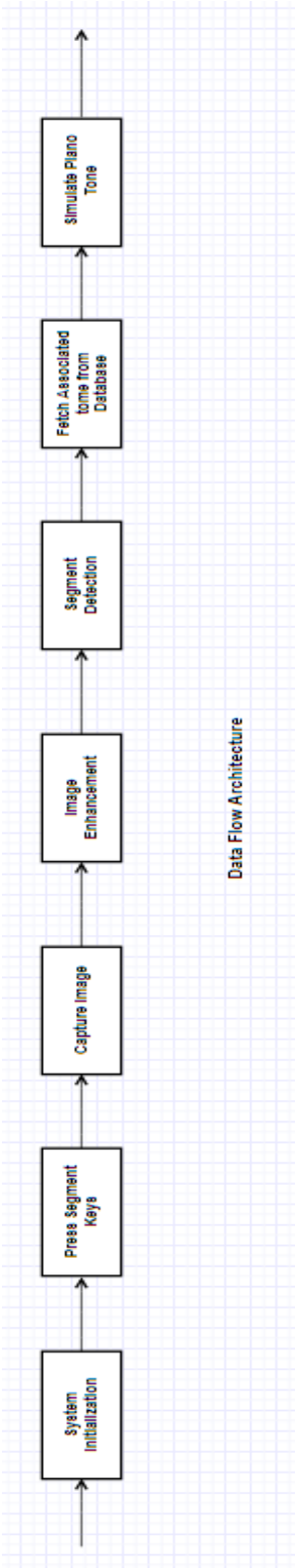
Design Model

1. Data Centered Architecture

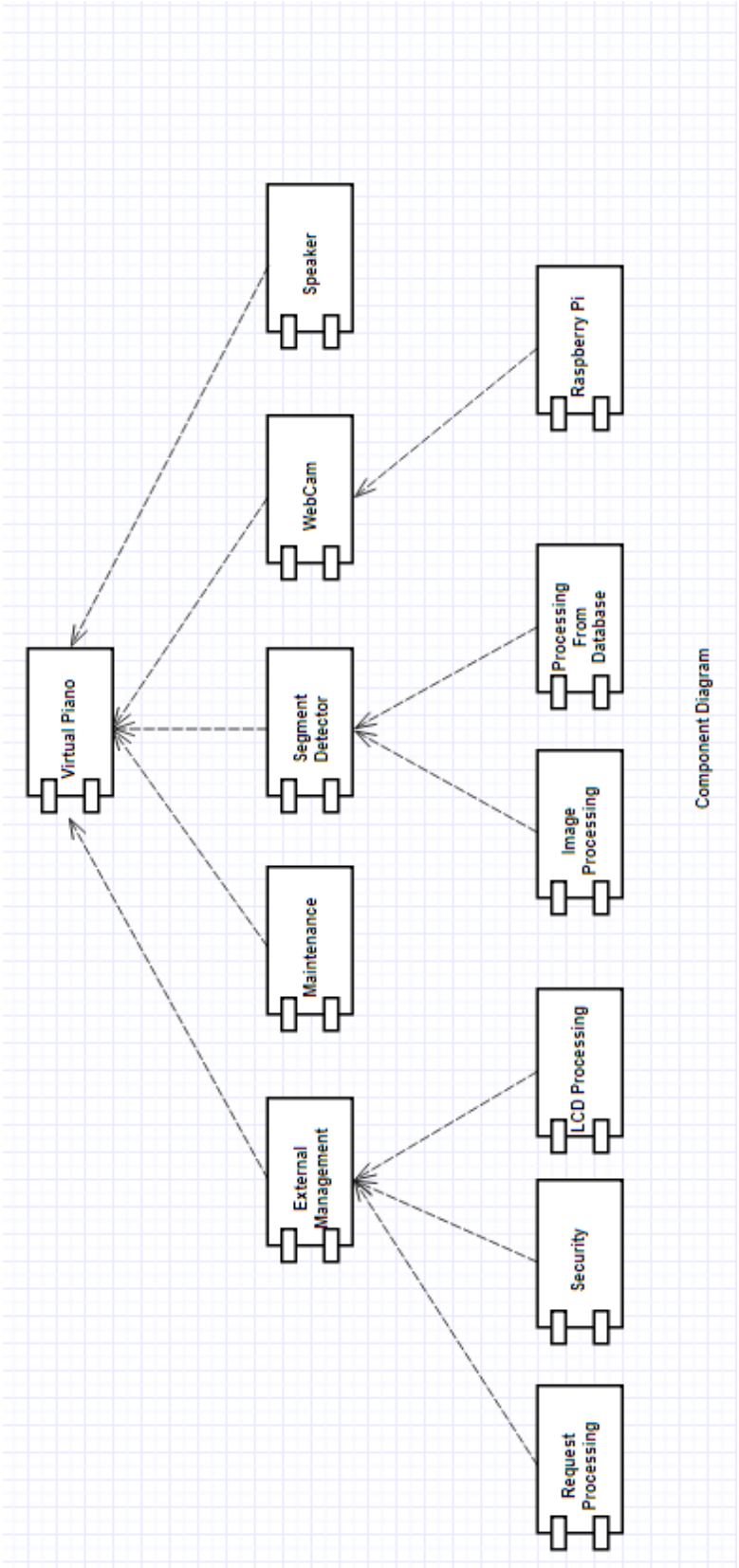


Data Centered Architecture

2. Data Flow Architecture



3. Component Diagram



4. Context Diagram