Engineering Design
GNG1103

Project Deliverable F: Prototype I and Customer Feedback

McQueen Zhao, Anastasia Kalatcheva, Kenza Tiendrebeogo, Matthew Waite, Anna Kim

November 5th, 2022

Table of Contents

1. Feedback from client

1.1. Feedback Received

- We have the right approach to keep the system simple
- They liked the problem statement because we are focused on the concepts that are important to them
- They liked the idea of having a three camera system that are mounted on the tripods because there are four players on the court, so they will have enough cameras to do the analysis.
- We might not be able to set up the program on multiple cameras
- Wifi connection is hard to implement
  - So we must fully understand how the wifi system works which will take a lot of time
  - So we might not be able to finish the project

1.2. How feedback will be implemented

- We will research how to connect the camera wirelessly
- If we do not succeed we need to implement a wired system between the camera and the computer.
- We need to develop simple and aesthetic interface
-

2. Prototype developed

2.1. Why

- For the prototype 1 we will develop the code in openCV that will be able to identify the ball's contours in the 3D space.
- This prototype will be used to detect the ball position in real life and the data of its position will be used to calculate the needed parameters for analysis (force, speed)
- Also its position data will be used in Unity render where the virtual ball will move in the same trajectory as the real life ball.

2.2. What

- Prototype type:
  - Focused prototype where we test one attribute of the product.
  - Analytical prototype where we analyze the position of the ball through camera detection.
- The prototype is the openCV code that detects ball in a 3D environment
- During the testing process, the code has to be able to detect the contours of the ball and show its position in x,y,z coordinates.
- To do so we will download python and insert the libraries (numpy and Cv2 which are the written programs that open the camera on the computer). The set up is free and can be downloaded in the internet
- So to do the prototype the program needs to be written which detects the data every frame per second.

- To build the prototype we will try to make it detect the ball's shape. So that it provides coordinates of the circles it detects. We will use the videos as our resource to write the program.
- Camera connection should be able to connect to the computer

2.3. When

Testing will happen between November 1 and November 6. Testing will occur mostly after the OpenCV code has been written and a camera connection with the computer has been established. This first prototype will need to be completed before we start working on the second prototype as it depends on the OpenCV code and camera connection being functional.

3. Analysis of critical components

A critical component is the OpenCv code which enables the connection of multiple cameras to the computer at once. The data from the cameras is required to provide more than one point of view.

In the program we are using Cv2 and numpy libraries to use the functions that are useful for opening the camera and detecting the ball shape. Videocapture function opens the web camera on the laptop but we can connect the phone camera to the program using an app.

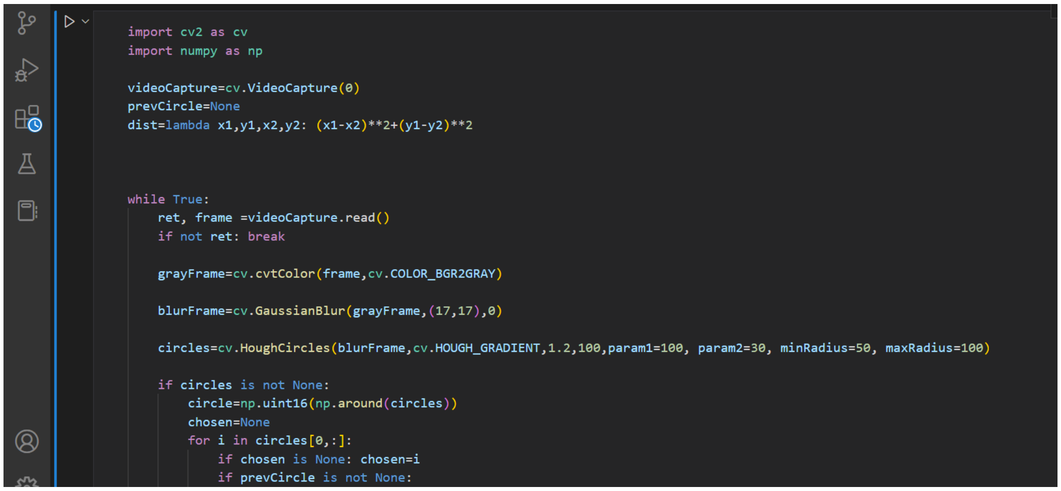
To do the circle detection the following code is written where the function of the circumference is shown.

The circumferences are detected on the frame and the position in x,y,z is recorded every millisecond.

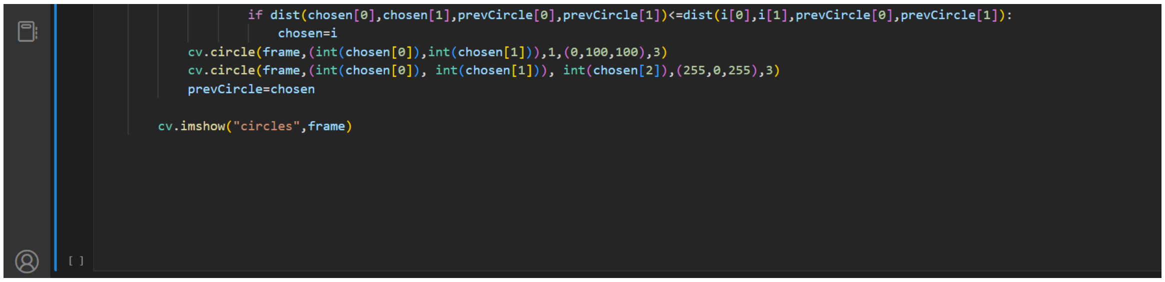
The imshow function shows the screen and how it detects the image of the ball's contours.

4. Documentation

4.1. Analysis

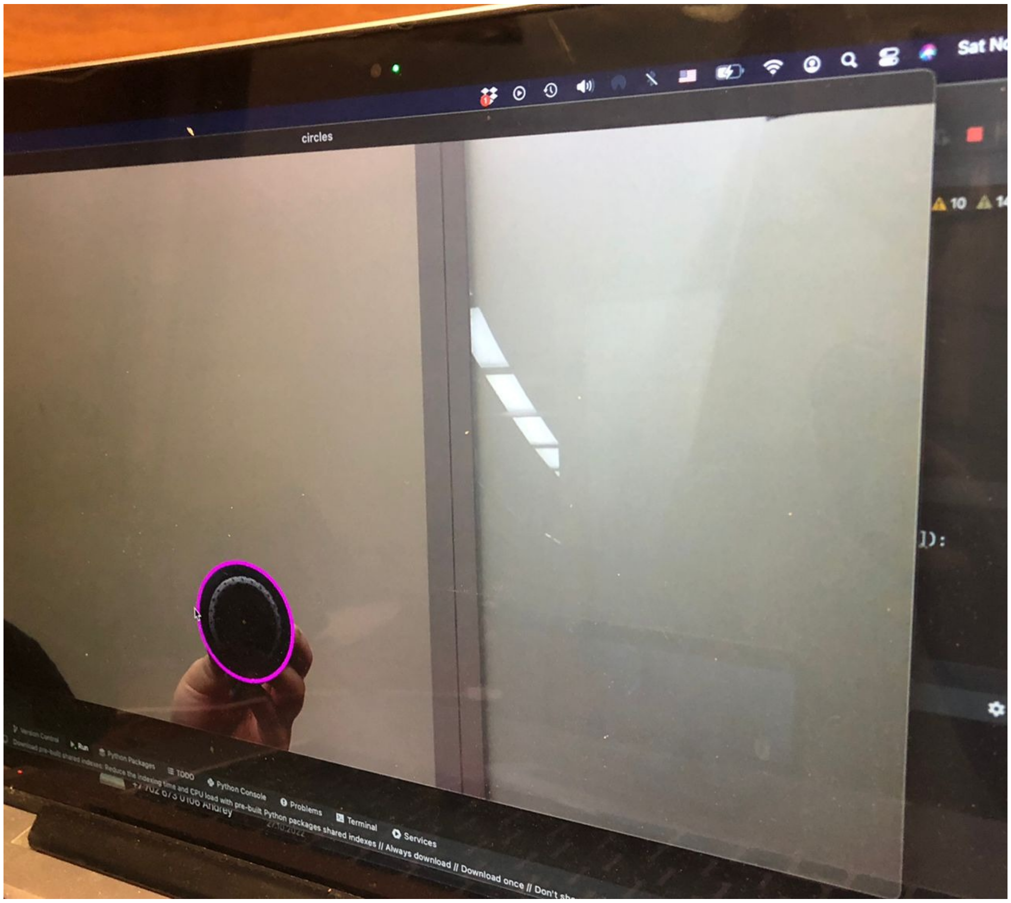


```
import cv2 as cv
import numpy as np

videoCapture=cv.VideoCapture(0)
prevCircle=None
dist=lambda x1,y1,x2,y2: (x1-x2)**2+(y1-y2)**2



while True:
    ret, frame =videoCapture.read()
    if not ret: break

    grayFrame=cv.cvtColor(frame,cv.COLOR_BGR2GRAY)

    blurFrame=cv.GaussianBlur(grayFrame,(17,17),0)

    circles=cv.HoughCircles(blurFrame,cv.HOUGH_GRADIENT,1.2,100,param1=100, param2=30, minRadius=50, maxRadius=100)

    if circles is not None:
        circle=np.uint16(np.around(circles))
        chosen=None
        for i in circles[0,:]:
            if chosen is None: chosen=i
            if prevCircle is not None:
```



```
                if dist(chosen[0],chosen[1],prevCircle[0],prevCircle[1])<=dist(i[0],i[1],prevCircle[0],prevCircle[1]):
                    chosen=i
        cv.circle(frame,(int(chosen[0]),int(chosen[1])),1,(0,100,100),3)
        cv.circle(frame,(int(chosen[0]), int(chosen[1])), int(chosen[2]),(255,0,255),3)
        prevCircle=chosen

    cv.imshow("circles",frame)
```

4.2. Results



4.3. Feedback and comments gathered

-Circles were not detected perfectly
-The system was not focusing on the ball presented to the camera
-The system was trying to identify circles in surroundings as well

5. Updated target specifications, detailed design and BOM (if needed)

After having reviewed the target specifications, the detailed design, and bill of materials, we considered them to still be consistent with our current objectives and therefore do not need to update them.

6. Prototype 2 test plan

Test ID	**Test Objective (Why)**	**Description of Prototype used and of Basic Test Method (What)**	**Description of Results to be Recorded and how these results will be used (How)**	**Estimated Test duration and planned start date (When)**	**Stopping criteria**
1	Prototype trial	Analytical method as the code in OpenCv can be modified in case the results do not measure up to expectations.	The main result was the recognition of circular shapes. This will be used to recognize the ball during games.	Test duration: 30-45 minutes Start date: 11/05/2022	System recognizes shapes (circles)
2	Open CV data storage	As the prototype, the video of the game will be uploaded to the program and the position,	The data of the position of the ball, velocity and force will be recorded in excel spreadsheet for	Test duration: ~30h Start date: 11/09/2022	Open CV data is transferred to the excel spreadsheet

		velocity, force and angle will be recorded in excel.	game analysis		
3	Open CV unity connection	The ball was recorded on the computer camera, moving the ball in unity in the same fashion.	Open CV will work at the same time as unity. As the position of the ball is being recorded, the coordinates are transferred to the virtual ball which will move in the same position coordinate but scaled to virtual court.	Test duration: ~30min Start date: 11/09/2022	Position coordinates detected from Open CV will be transferred to the virtual ball in unity, so that it moves with the same position coordinates.
4	Unity court render	The prototype will be tested on the padel court. The position of the ball on the court should match the position of the ball in unity court render.	In unity, the court will be used to show the relative position of the virtual ball.	Test duration: ~15min Start date: 11/11/2022	A virtual padel court is created and shows movement of ball in real-time
5	Establish wifi (or other type) connection from phone camera to computer	Camera is to be successfully connected wirelessly to the computer in order to transmit data input into OpenCV and Unity	The video that is being recorded on the phone camera is shown on the screen of the connected computer. The resulting video is run through the open CV code, to detect the position of the ball.	Test duration: ~15min Start date: 11/10/2022	When connecting between phone and computer is simple and connection between the two is stable

Wrike Link

https://www.wrike.com/workspace.htm?acc=4975842&wr=20#folder/968841859/timeline3?filters=status%3Dactive&overlayFullScreen=0&showInfo=0&spaceId=-1&viewId=109685341