|  |  |  |
| --- | --- | --- |
| **Computer Vision Challenge** | | |
| 130\_Recruitment | Date: | 2/1/2025 |
| Performed by: | (insert name) |

A person with a football ball on a field

Description automatically generated

# Introduction

The Computer Vision challenge has the purpose of showing basic object tracking and prediction skills, while also combining some aspects of Mathematics. This document showcases the development and testing of a football object detection system, capable of processing RGB information from a camera and outputting the ball trajectory, position and velocities, relative to the camera.

To complete this challenge, any and every technology is accepted. Use whatever you consider suits the job. While the task can be done quite quickly with a low-quality level, try to accomplish it to the best of your skills.

# Challenge Description

Together with this document, there is an AVI file which depicts a person kicking a ball. The purpose is to detect ball, track its position and predict its trajectory.

As the task requires multiple steps, several sub-challenges are described:

## 2D Football detection

As the football is the core of the challenge, it must be detected using a system of your choice. Object detection algorithms, color tracking or shape tracking are just some of the options available for performing this sub-challenge.

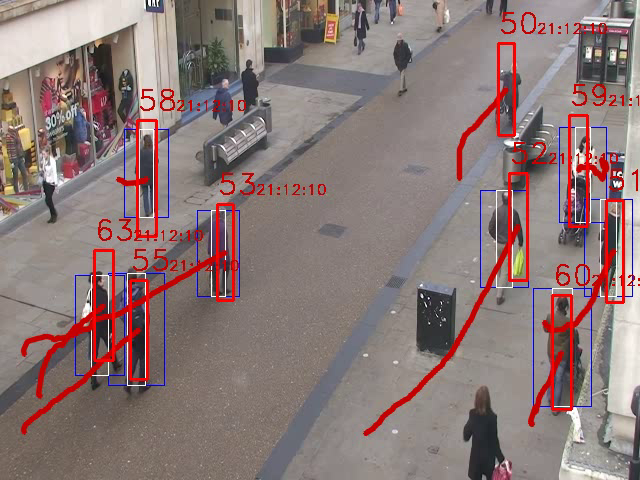
The output is a video file, showing how the ball is detected, in the RGB video file.

## 3D Football detection

Continuing the previous challenge, the distance towards the ball and its position must be computed using the available video. The output of this sub-challenge is a file such as an excel, which shows the position of the ball (x,y,z) and the distance towards it, in meters, relative to the camera, versus time. As a hint, the used camera to film the scene was the RGB camera of an Intel RealSense D435i [1], and the ball is a standard diameter football.

## Trajectory tracking

The trajectory the football is performing, must be indicated, with a simple video overlay. An example in the following image:



The output is a video showing the trajectory which the ball executed.

## 2D Map

The most challenging part of the … challenge … is to compute a top-view map containing the positions of the ball and of the camera, relative to the ground. You can output this top-view 2D Map in a video. You know you did this right if the position of the ball is not affected by the position of the camera. Make sure you include the trajectory of the ball in the 2D top-view Map.

## General recommendations

The following general recommendations and guidelines are given:

* You can use any available software packages you can find; the internet is your friend here.
* For extra points, we want to see the code well documented.
* Again, for extra points, the following chapter is left free for you to write a “documentation” of each step, how it was tackled and implemented. Please include the used formulas.
* You have maximum 10 hours to complete as much as possible from the task. Not everything is required, but the more, the better.
* At the end, please send all videos, code, documentation to [octavian@dotlumen.com](mailto:octavian@dotlumen.com) and [cornel@dotlumen.com](mailto:cornel@dotlumen.com) and [diana@dotlumen.com](mailto:diana@dotlumen.com) using a file transfer service such as WeTransfer.

# Implementation

// Your documentation here

# Bibliography

|  |  |
| --- | --- |
| [1] | Intel RealSense, "Intel RealSense D435i Camera," [Online]. Available: https://www.intelrealsense.com/depth-camera-d435i/. |
| [2] | Intel RealSense, "Beginner's Guide to depth," [Online]. Available: https://www.intelrealsense.com/beginners-guide-to-depth/. |