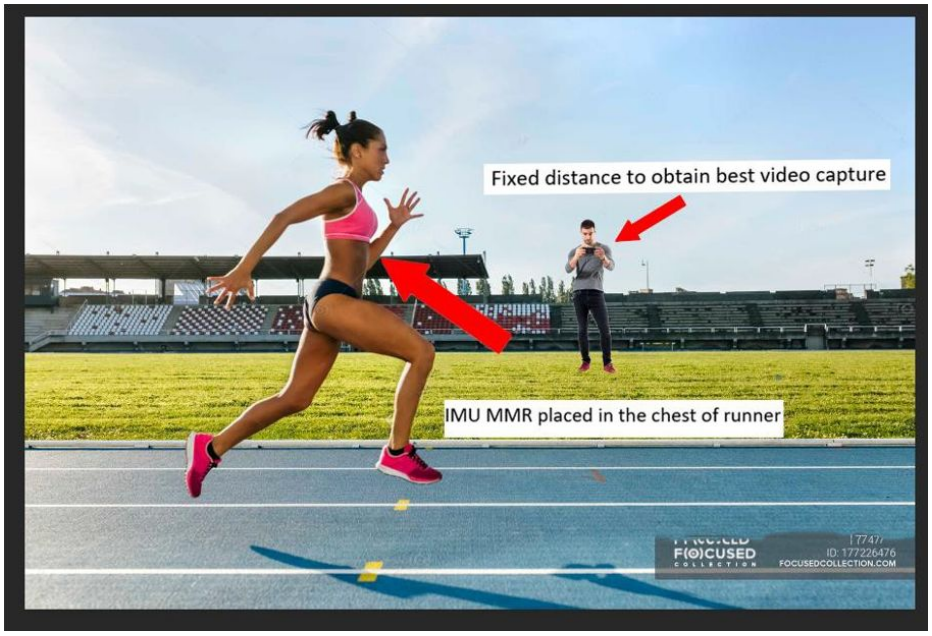
1. **Operation Manual**

1.1 Project Overview

The purpose of our design is to provide a product that will collect information for runners, sprinters, and coaches to improve sprinting performance. Currently, there are many devices that help with collecting data during runs, but they either cost too much or provide only a few factors. In order to meet this need, we designed our product to provide multiple factors, reliable data and be affordable. The factors that are considered important for the performance of sprinters are: speed, acceleration, timing, distance, and stride (the distance covered when you take two steps, one with each foot). Therefore, we have made these data points our focus.

Our program allows the user to upload videos taken on an iPhone camera to ensure its ease of use. This makes our design affordable, while still giving the customer info at the same accuracy as other data tracking devices. The Sprinter Data Collector gives the user full awareness of their performance when running.



1.2 Component/Module Description

Our program uses processed data from a camera and an accelerometer using the C++ programming language.

1.2.1 Video

The camera processing is done through a library extension known as OpenCV. The camera captures the footage necessary to calculate distance, time, speed, and stride length. In order for the user to gather the data from the video the user will need to run our software provided. This will read the video from the user's computer and then be able to give deliverable outputs to the user. Below are a given set of instructions to install some of the tools/libraries needed as well as download and run a python script.

The following commands will need to be entered into a Mac Computers Terminal:

-conda create -n Test1 python==3.6.9
-conda deactivate
-condo activate "*Name of project*"
-pip install imutils
-import imutils
-pip install numpy
-import numpy
-pip install opencv-python
-pip install jupyter
-jupyter notebook

After entering these commands a python script will open. Included there will be a file that will contain code to extract our outputs from the video that the user will run.

1.2.2 Accelerometer Data

IMU(Inertial Measurement Unit) is the accelerometer that will be used. The accelerometer, attached to the chest of the runner, gathers acceleration, speed, time, and stride frequency. The IMU will be attached to the chest of the sprinter. The data is accessed through the MetaBase app, exporting the data points through a CSV file. It will be running at sample rate of : 0.001Hz – 100Hz stream – 800Hz log

1.2.3 Program

Data acquired from these two devices provide accurate values of speed, acceleration, timing, distance, and stride. Speed, time and components of stride are measurements in both devices. Our program takes the data from the camera and the accelerometer as inputs. It calculates our data points and delivers them to the user as outputs.

1.3 Integration

With two methods of collecting data, coordinating the camera and accelerometer to give accurate measurements is essential. Both camera and accelerometer data can be acquired through phone. Top speed, stride frequency, and distance will primarily benefit from synching the data. In an attempt to improve accuracy of our outputs the different methods of extracting data should be compared.

1.4 Operation

1.4.1 Data Collection

1. Measure 20 meters on track. Try to pick an area with a stationary background.
2. Make marks with objects or chalk at 0 and 20 meters.
3. Measure 13.5 meters away from the 10-meter mark. Ensure that the field-of-view covers the 20 meters. (If not, measure the distance at where the 20 meters is covered and record/ measure from that distance)
4. Runner puts on the IMU and prepares to start the sprint.
5. Prior to starting sprint:
    a. Runner stands upright
    b. START IMU and Video recording
    c. Runner gets into start position and takes off when ready
    d. Ensure that sprint motion is continued through the 20 meters
6. Gather runner's height

1.4.2 Data Processing

1. Download all software needed (C++, Python, and OpenCV libraries and GNUPlot).
    a. C++ is done in Visual Studios Community 2019, which is free.
    b. Opencv library download is of 4.2.0.
    c. GNUPlot is a free download as well.

2. Download files composing of the Sprinter Data Collection program provided by Github.

3. Before running the program insert the video file and csv file from the accelerometer onto the program file folder.

4. When running the program input the runners weight and height.

5. Program will play a video file in correspondence to GNUPlot creating a graph of acceleration, speed, and stride marks.

Image processing :

Obtaining velocity of runner Accurate Start and Finish times of runner (20m)

Tracking individual feet of runner

Calculating stride length