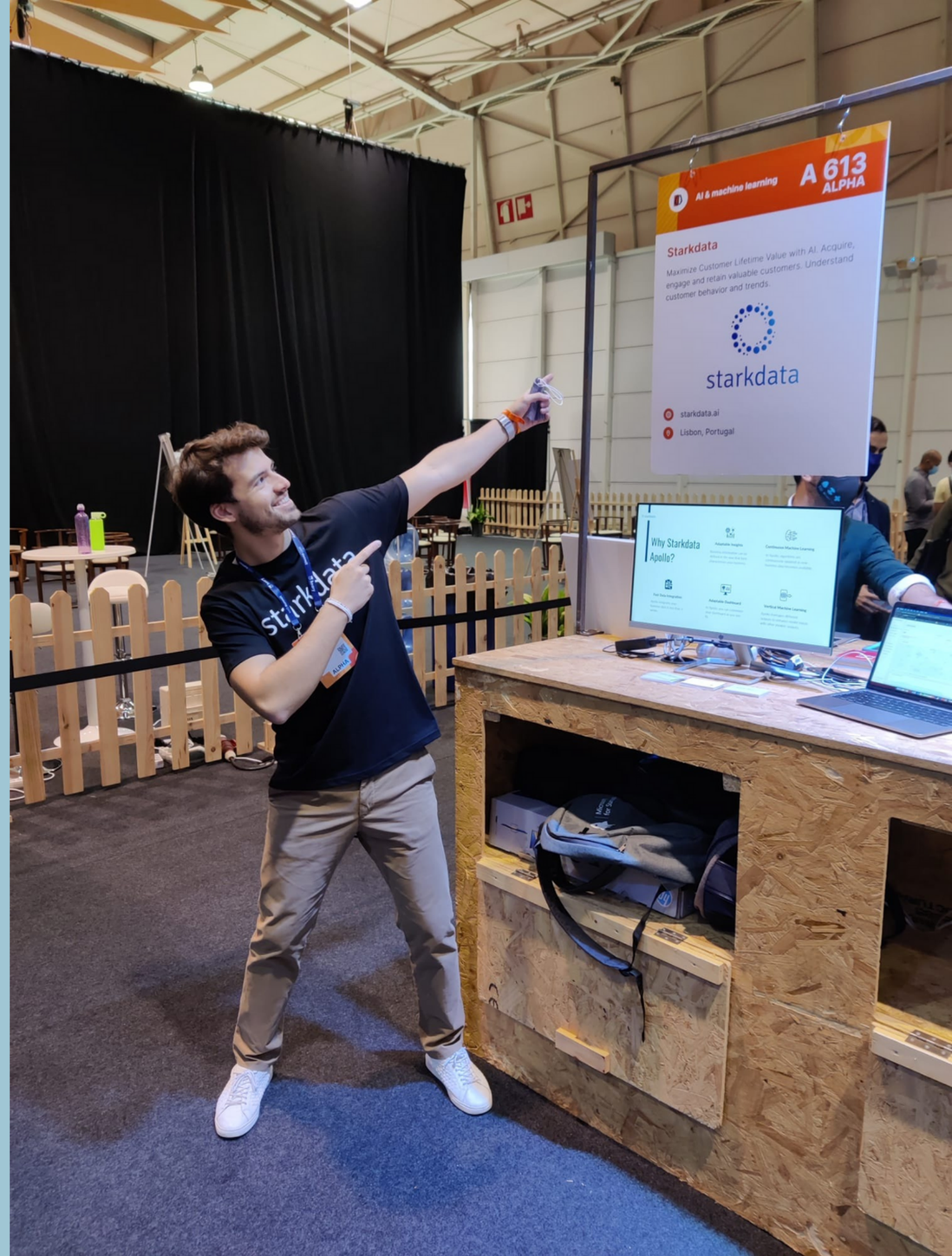
My Computer Vision Journey

(so far)

About me

João Ramos
Based in Lisbon
MSc Computer Science at IST (AI specialization)
Co-Founder at Starkdata



My journy

Project Timeline

1

Autonomous Flying

2

Hand Gesture Controler

3

Body Language Decoder

4

AI Gym buddy

5

Space Managment

Project Guide

01

Approach

A quick approach with the steps I followed

02

Example

An example created by me (video)

03

Challenges and Solutions

Challenges I faced and possible solutions

04

Real-world example

Real-world examples and use cases

How it started

DJI Tello

- Programable drone (official SDK)
- Affordable
- Ideal for indoor

How can I make the smartest drone possible?

Autonomous flight?

Space awareness?

Building mapping?

Obstacle avoidance?

Follow me while doing snowboard?

Learn to fly with reinforcement learning in a Unity environment?

etc. etc. etc. etc.

Autonomous flight?

Space awareness?

Building mapping?

Obstacle avoidance?

Follow me while doing snowboard?

Learn to fly with reinforcement learning in a Unity environment?

etc. etc. etc. etc.

A Simple autonomous driving

01

Identification of the nearest face

DJI camera plus a real-time object detection model

02

Track that face

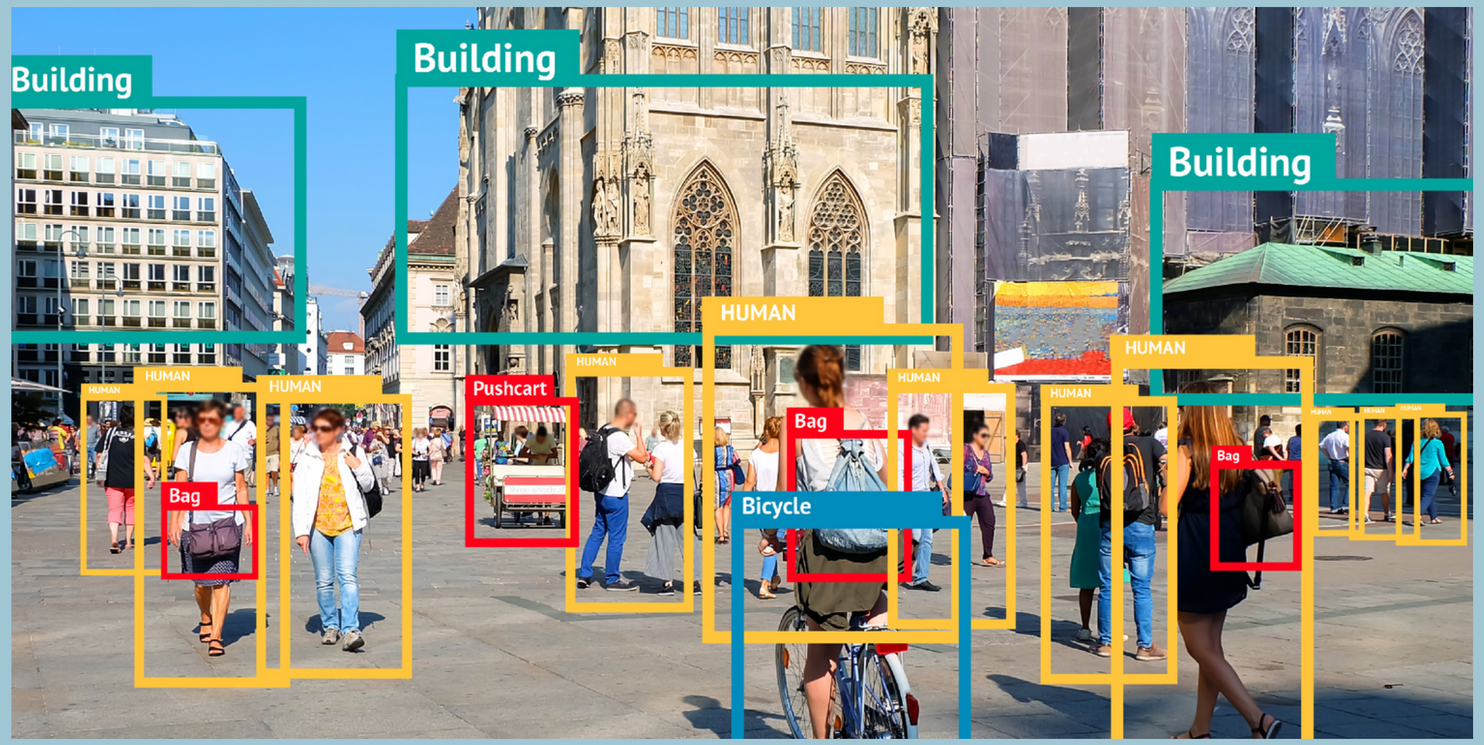
Adjust controls based on the face in the frame

Object Detection Model

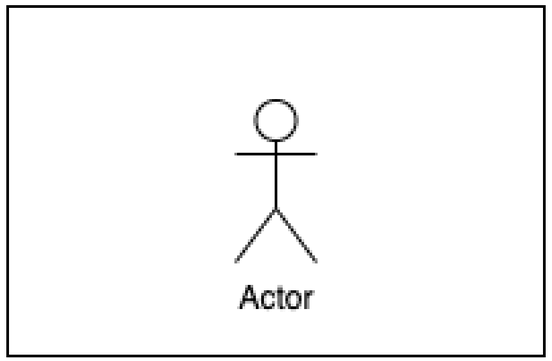
Model Used
Haar Cascade Classifier from OpenCV

Advantages
Light, runs without fancy hardware.
Simple to implement and use

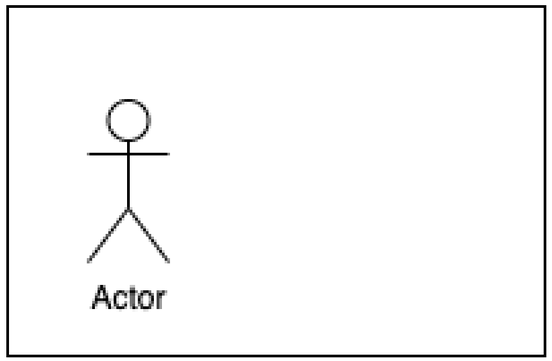
Disadvantages
Performance



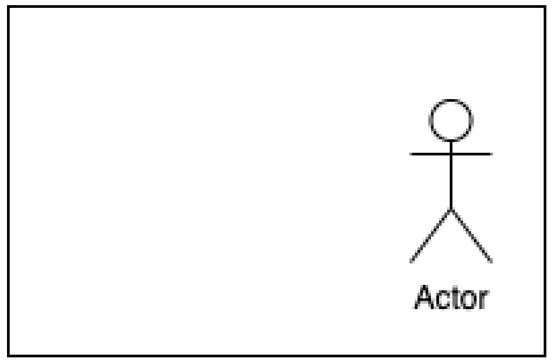
Tracking Face algorithm



Frame: Person Centered
Drone Command: Stay

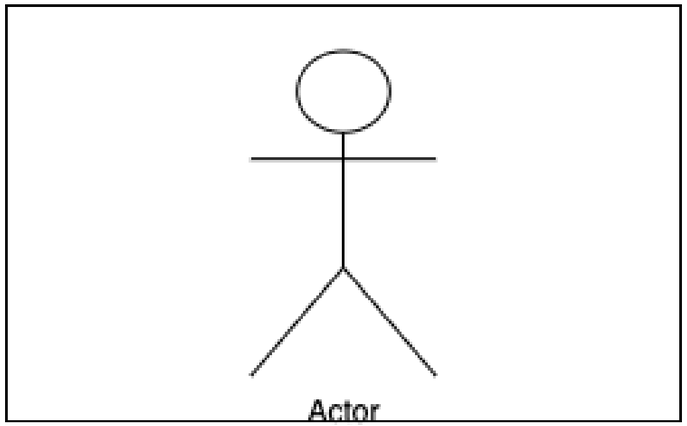


Frame: Person to the left
Drone Command: rotate left

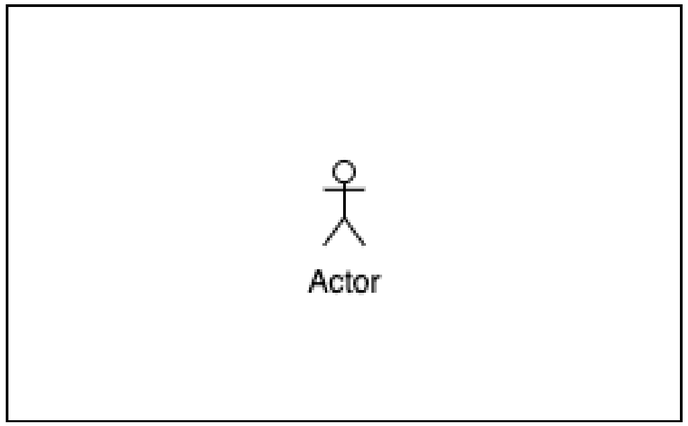


Frame: Person to the right
Drone Command: rotate right

Tracking Face algorithm



Frame: Too close to the drone
Drone Command: Move backward



Frame: Too far from the drone
Drone Command: Move forward

First Project

DJI Tello with autonomous driving

- Tello turned on and moved up
- Detects a face and moves forward
- Rotates to the left
- Moves backward
- Moves forward
- Finally moves backward and to the left simultaneously

Main challenges

Low performance Model

Couldn't recognise people from the side or back

PID Tunning

Hard to have a good PID vector with a low background in robotics

Solutions

Low performance Model

YoloV4 with Darknet
YoloV5 with Pytorch

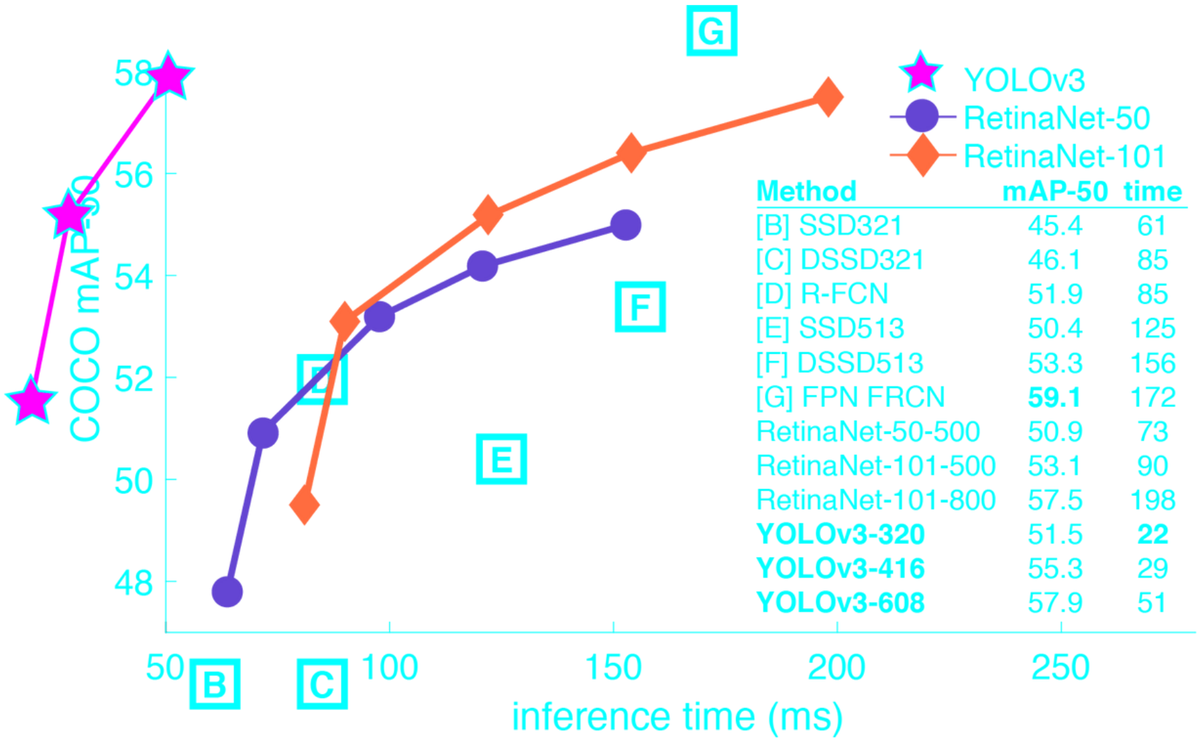
PID Tunning

time, patience and experiment

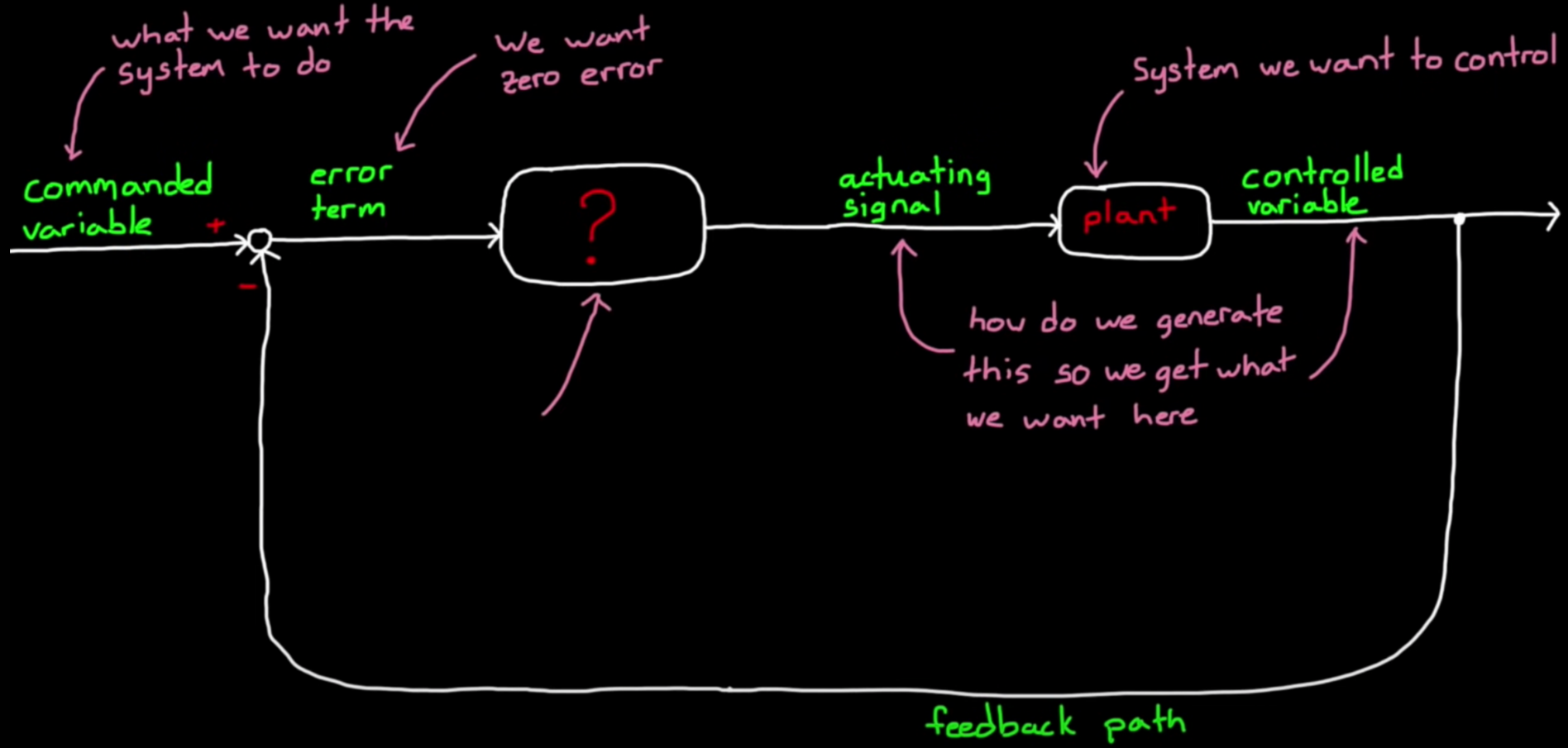
What is Yolo?

YOLO: Real-Time Object Detection

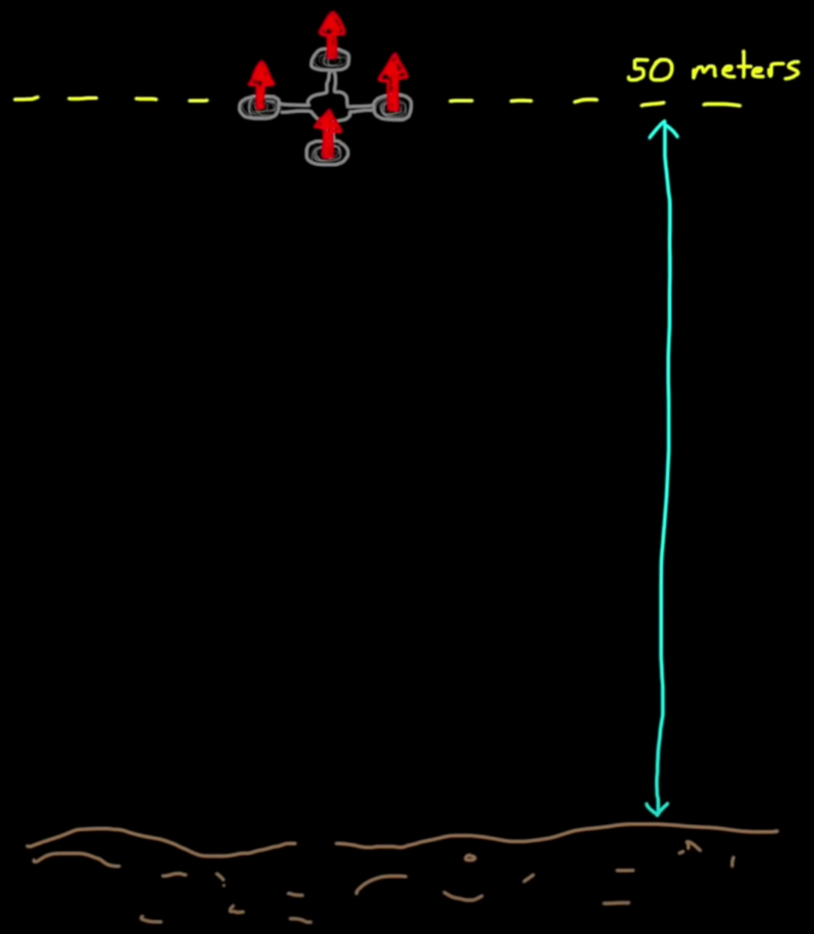
You only look once (YOLO) is a state-of-the-art, real-time object detection system.



Simple PID Explanation

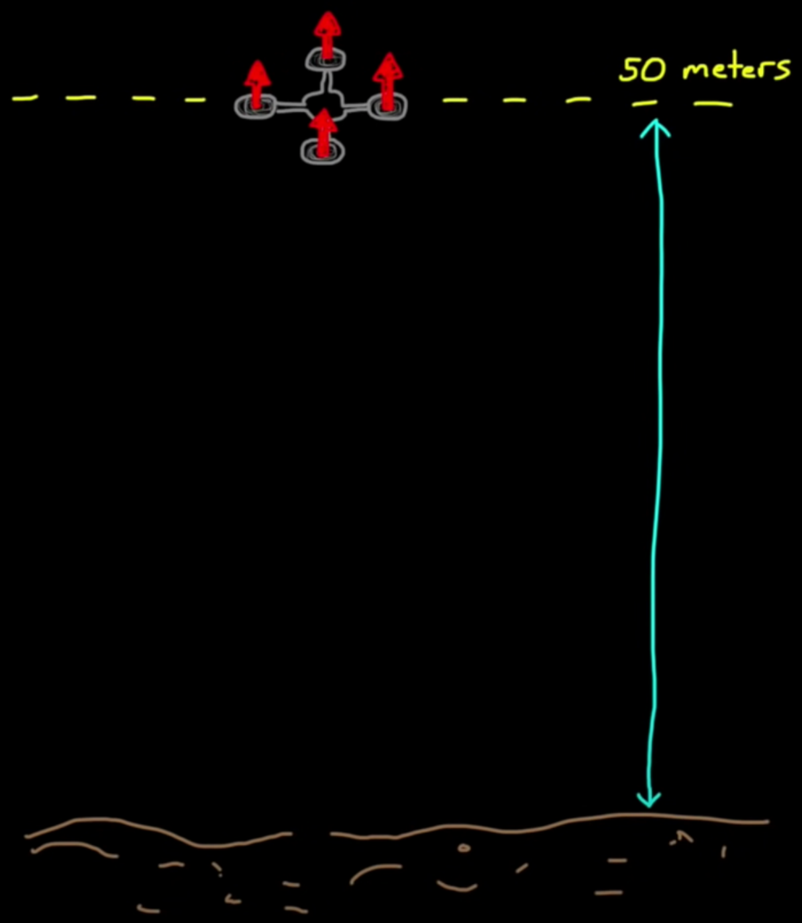


Simple PID Explanation

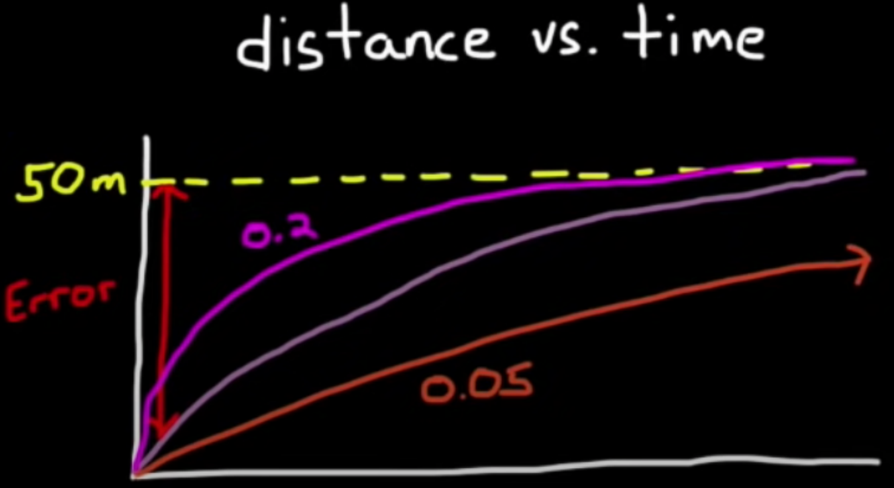


Error is 50 meters
propellers will spin up
drone will rise, reducing the error

Simple PID Explanation

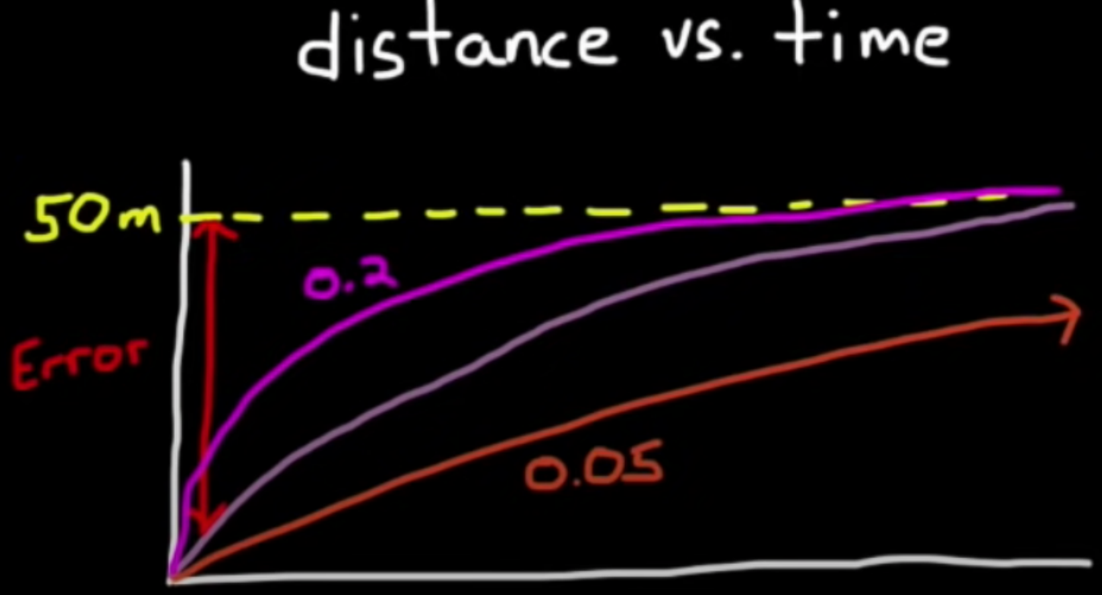


Error is zero meters



Error is 50 meters
propellers will spin up
drone will rise, reducing the error

Simple PID Explanation



Faster -> risk of passing the point
Slower -> stable

Real World Examples

Fast Food Delivery

Very futuristic but it will be possible in the future

Agriculture

inspects and control for a variety of crops

Equipment inspection

More frequent and accurate monitoring of large-scale or dangerous equipment.

Different ways to control a drone

01

Smartphone

Native DJI App

02

Keyboard

A simple API

03

Hand gesture

ML model for hand gesture recognition

04

Voice

ML model for voice control

Different ways to control a drone

01

Smartphone

Native DJI App

02

Keyboard

A simple API

03

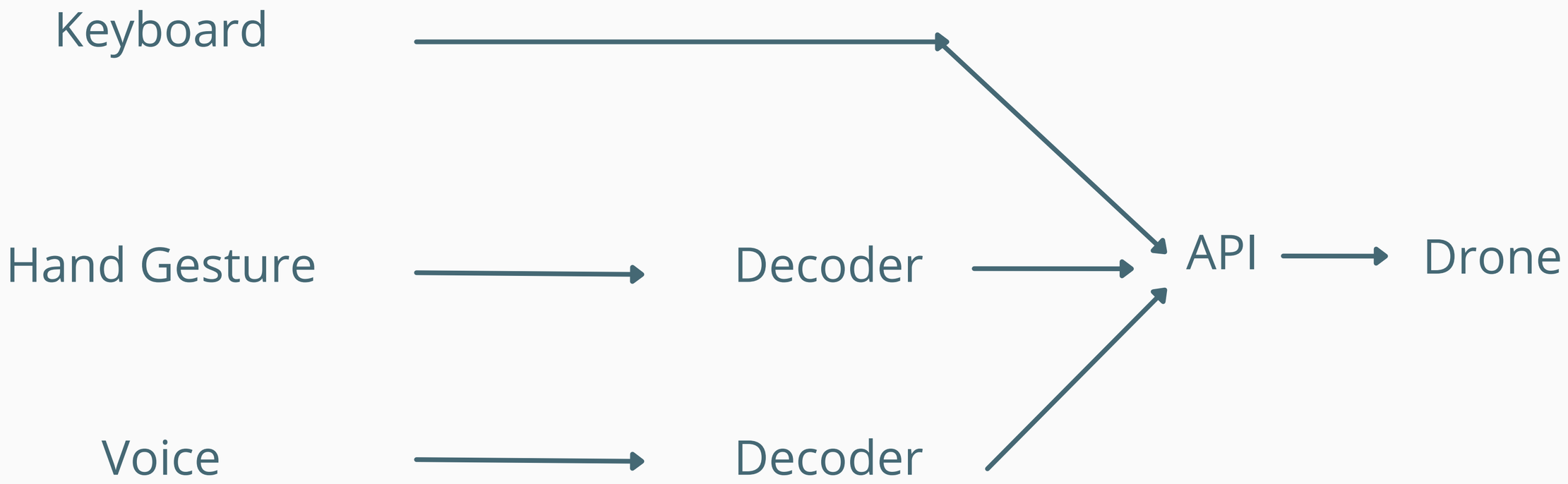
Hand gesture

ML model for hand gesture recognition

04

Voice

ML model for voice control


Keyboard
Hand Gesture
Decoder
API
Drone
Voice
Decoder

Hand gesture control

01

MediaPipe model

Detect hand and fingers in real-time

02

Custom neural network

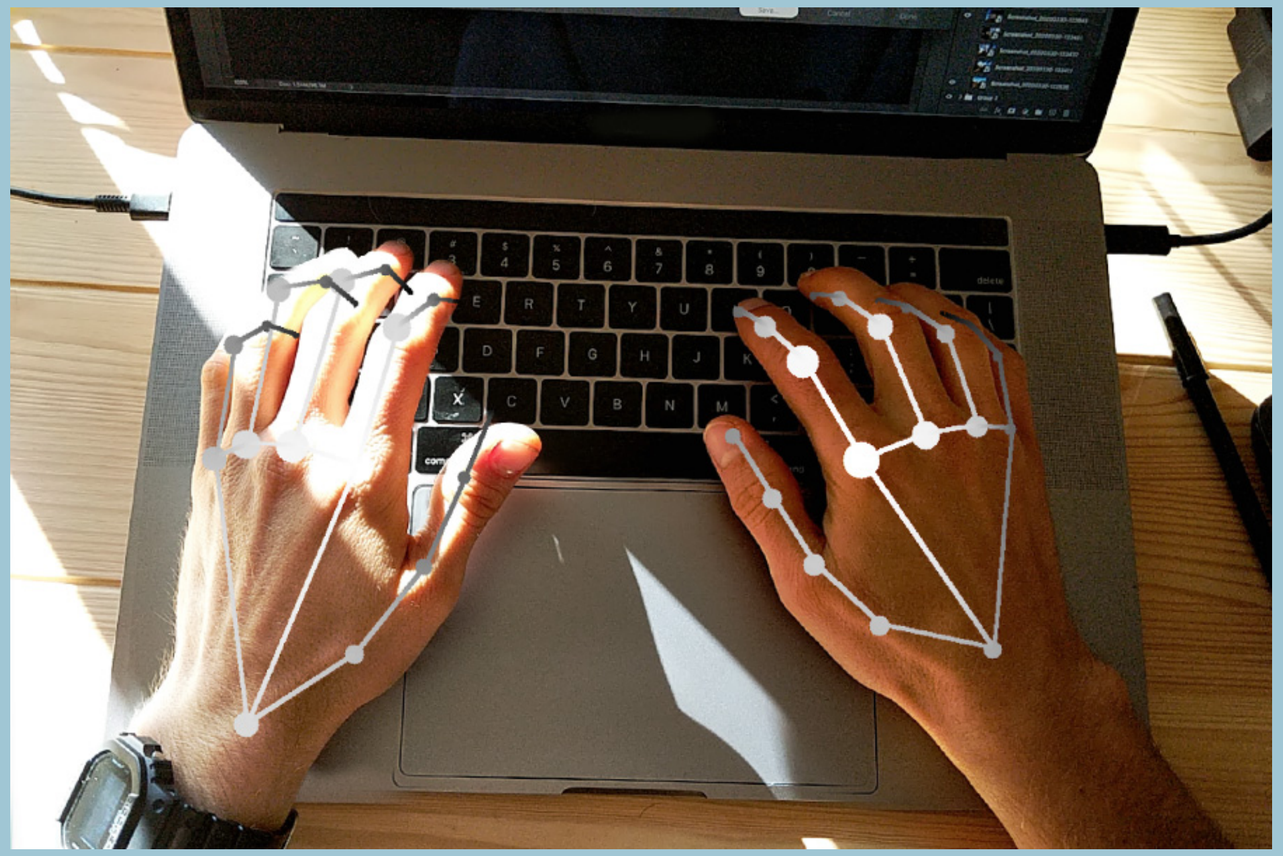
Hand gesture and finger movement classification

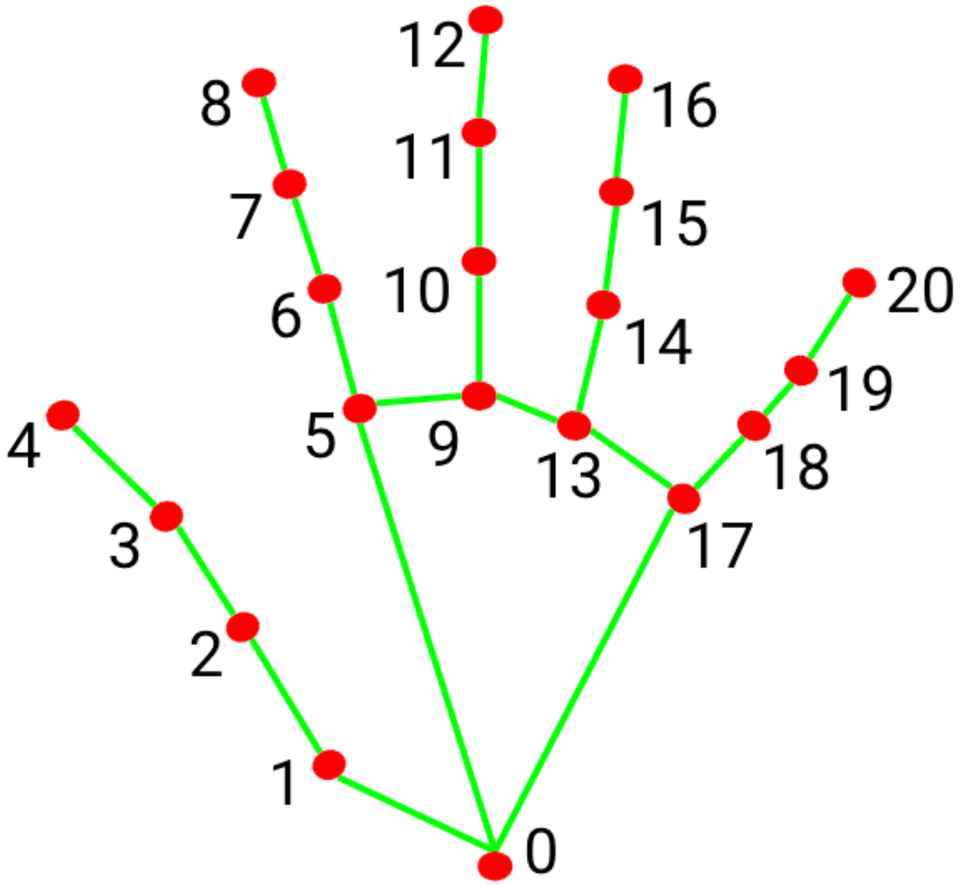
Hand detection

Model Used
MediaPipe Hand model

Advantages
Good documentation
Simple to implement
Works with almost any hardware (very light)

Disadvantages
None





0. WRIST
1. THUMB_CMC
2. THUMB_MCP
3. THUMB_IP
4. THUMB_TIP
5. INDEX_FINGER_MCP
6. INDEX_FINGER_PIP
7. INDEX_FINGER_DIP
8. INDEX_FINGER_TIP
9. MIDDLE_FINGER_MCP
10. MIDDLE_FINGER_PIP
11. MIDDLE_FINGER_DIP
12. MIDDLE_FINGER_TIP
13. RING_FINGER_MCP
14. RING_FINGER_PIP
15. RING_FINGER_DIP
16. RING_FINGER_TIP
17. PINKY_MCP
18. PINKY_PIP
19. PINKY_DIP
20. PINKY_TIP

MediaPipe Hand model

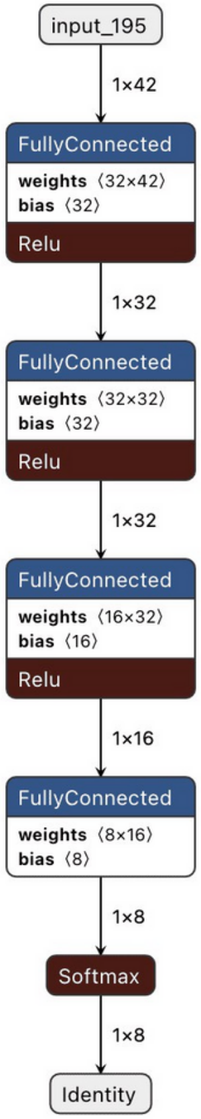
https://google.github.io/mediapipe/solutions/hands

Custom neural network

Model Used
4 Fully-connected layers and 1
Softmax layer for classification

Advantages
Simple
Gets the job done

Disadvantages
Not reliable enough for production

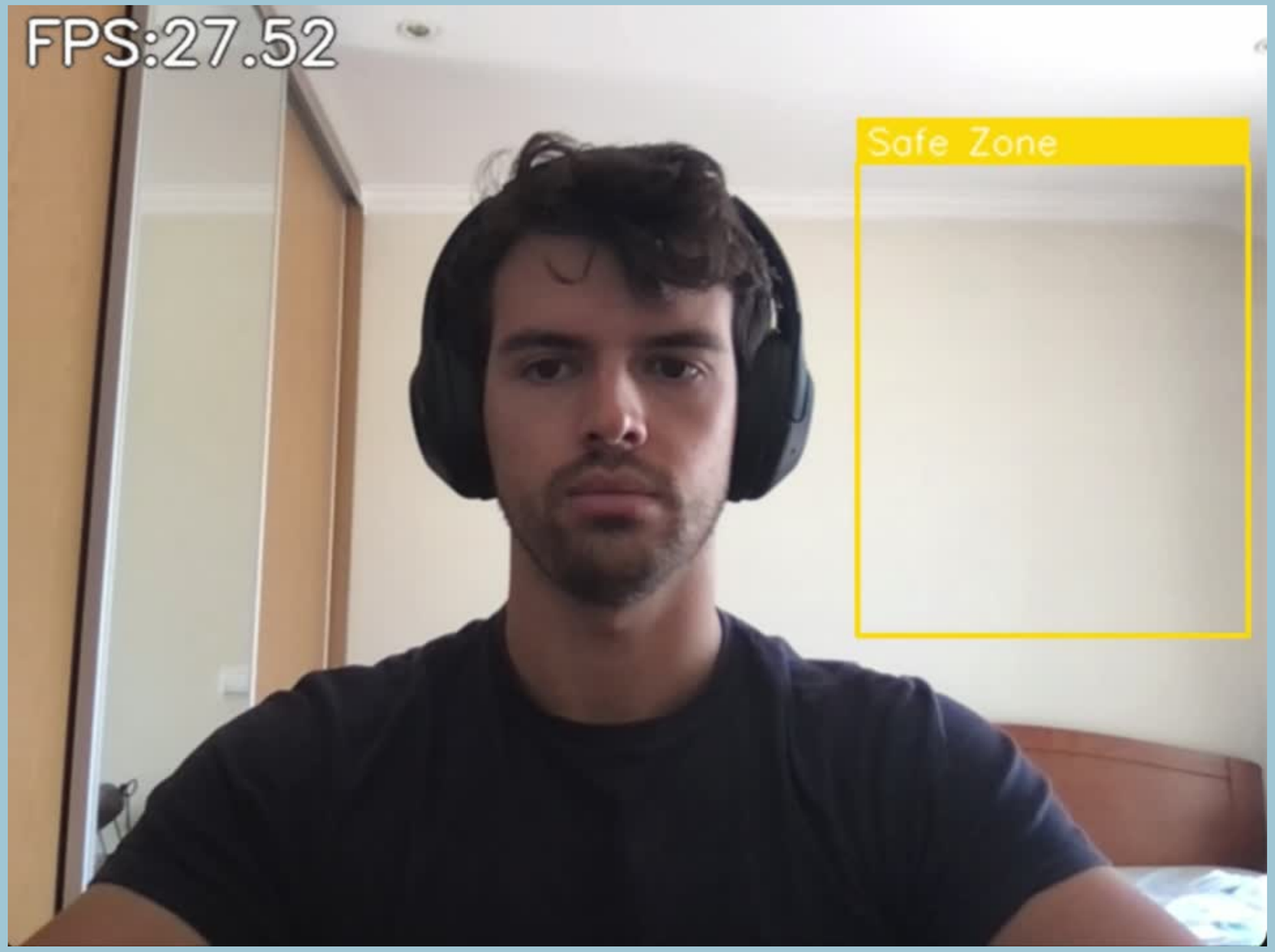


https://github.com/kinivi/tello-gesture-control

Second Project

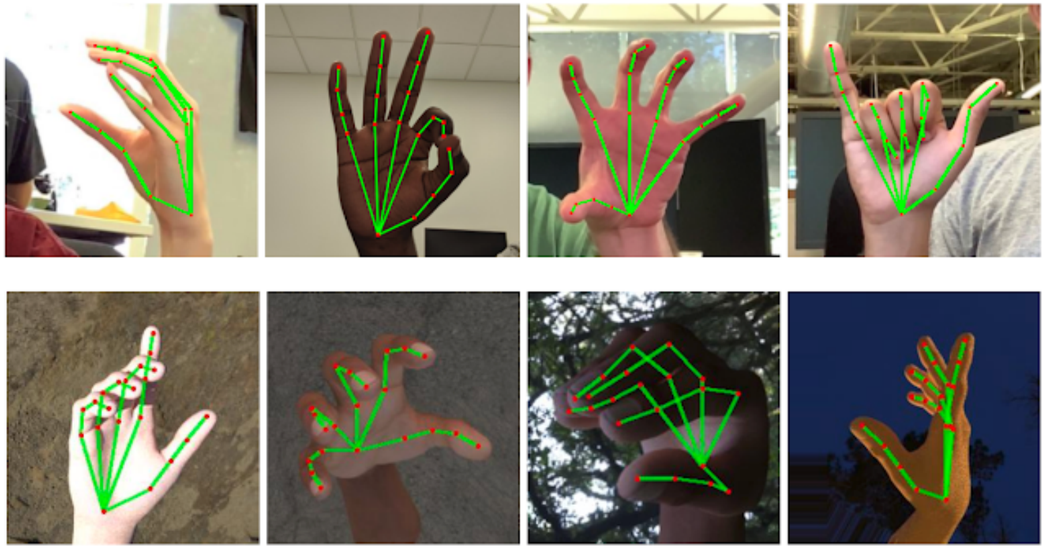
Hand Gesture Control

- SafeZone: Only gestures inside SafeZone will be detected as commands
- Detects different gestures
- Detects finger movements (Clockwise and Counter Clockwise)



Project Review

- I had to create my own dataset
- The model was able to distinguish easily different hand gestures
- Relatively easy to create a model for classification
- No major complication or difficulty



Real World Examples

Hand Tracking for VR

Interact without needing VR controllers

Virtual Piano

Play virtual piano or any other instrument

Virtual Desktop

device control through hand gestures

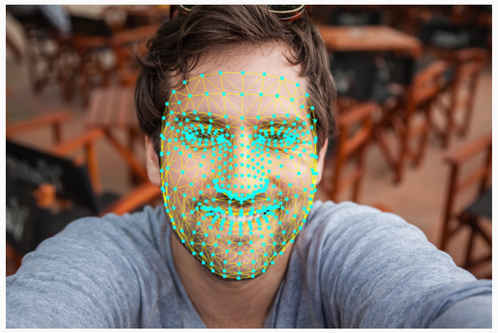
sign language understanding

Understand sign language instantly

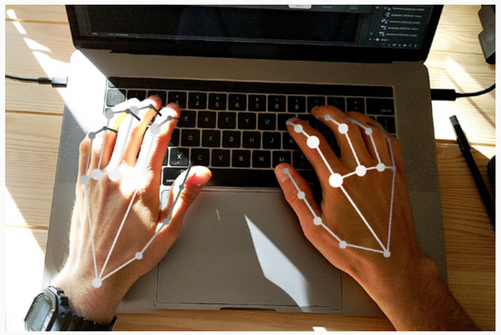
MediaPipe is really Powerful

What else can I create?

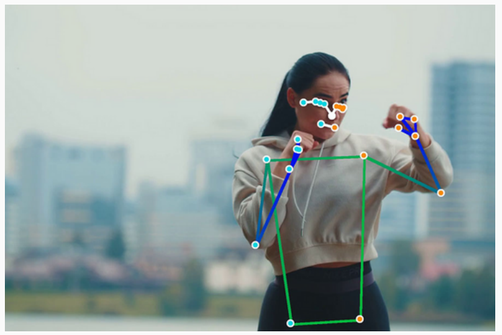
MediaPipe Models



Face Mesh

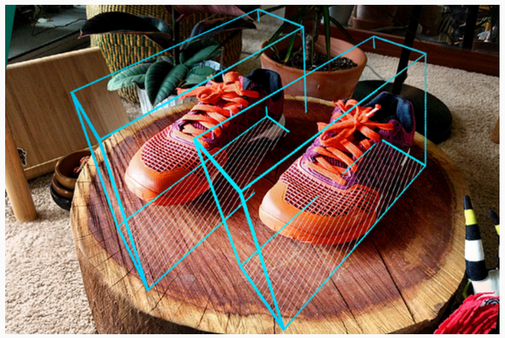


Hand Tracking



Pose Detection and Tracking

Object Detection



3D Object Detection

Hair Segmentation

Next projects

Body language decoder

A model that understands poses, reactions and body language

'AI Gym buddy'

A pure algorithm model that detects body-parts and joints and calculates their anguls

01

MediaPipe model

Detect body parts in real-time

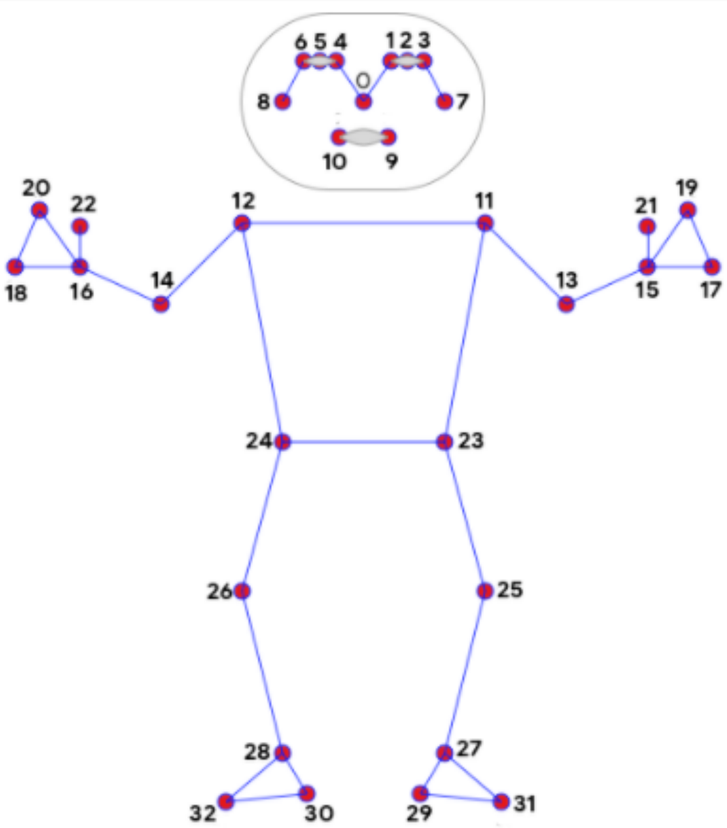
02

Coding

An algorithmic that calculates angles in body parts

'AI Gym buddy'

MediaPipe Pose model



0. nose
1. left_eye_inner
2. left_eye
3. left_eye_outer
4. right_eye_inner
5. right_eye
6. right_eye_outer
7. left_ear
8. right_ear
9. mouth_left
10. mouth_right
11. left_shoulder
12. right_shoulder
13. left_elbow
14. right_elbow
15. left_wrist
16. right_wrist
17. left_pinky
18. right_pinky
19. left_index
20. right_index
21. left_thumb
22. right_thumb
23. left_hip
24. right_hip
25. left_knee
26. right_knee
27. left_ankle
28. right_ankle
29. left_heel
30. right_heel
31. left_foot_index
32. right_foot_index

Disadvantages

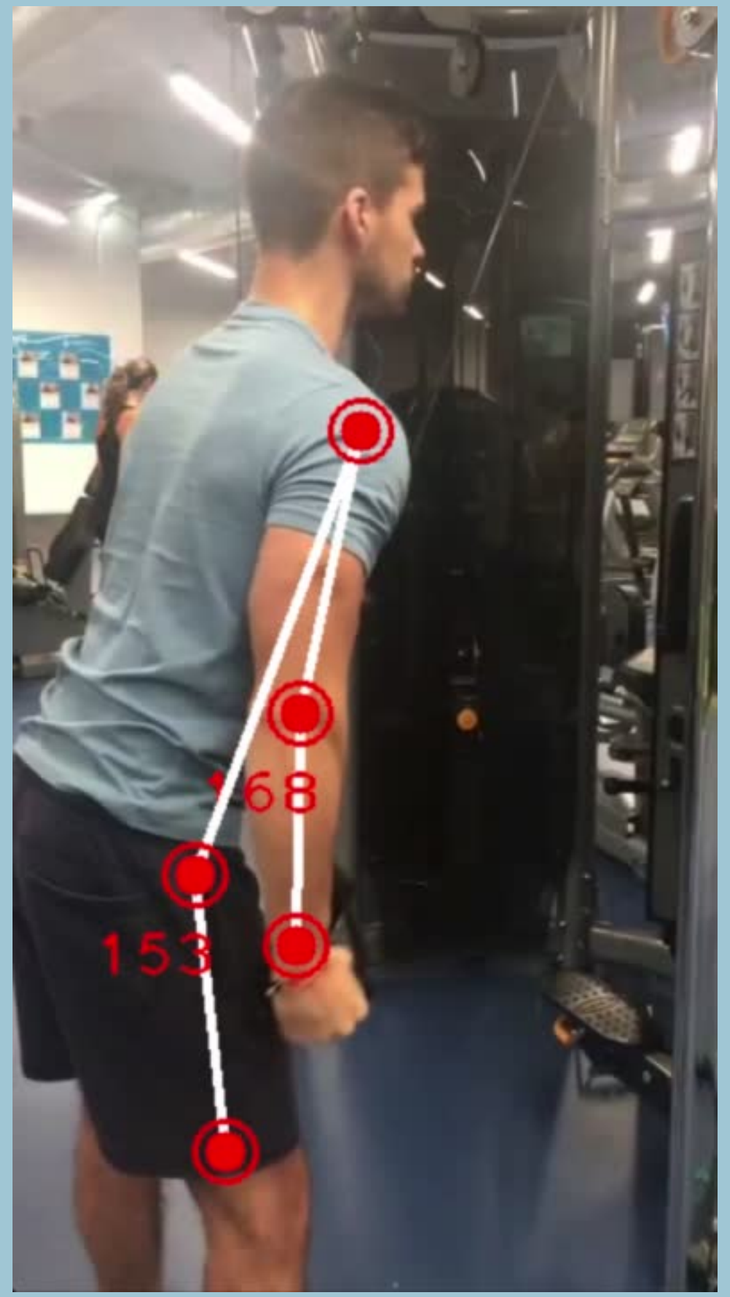
- Low accuracy when parts of the body overlap
- Low accuracy when a person uses baggy cloths

https://google.github.io/mediapipe/solutions/pose

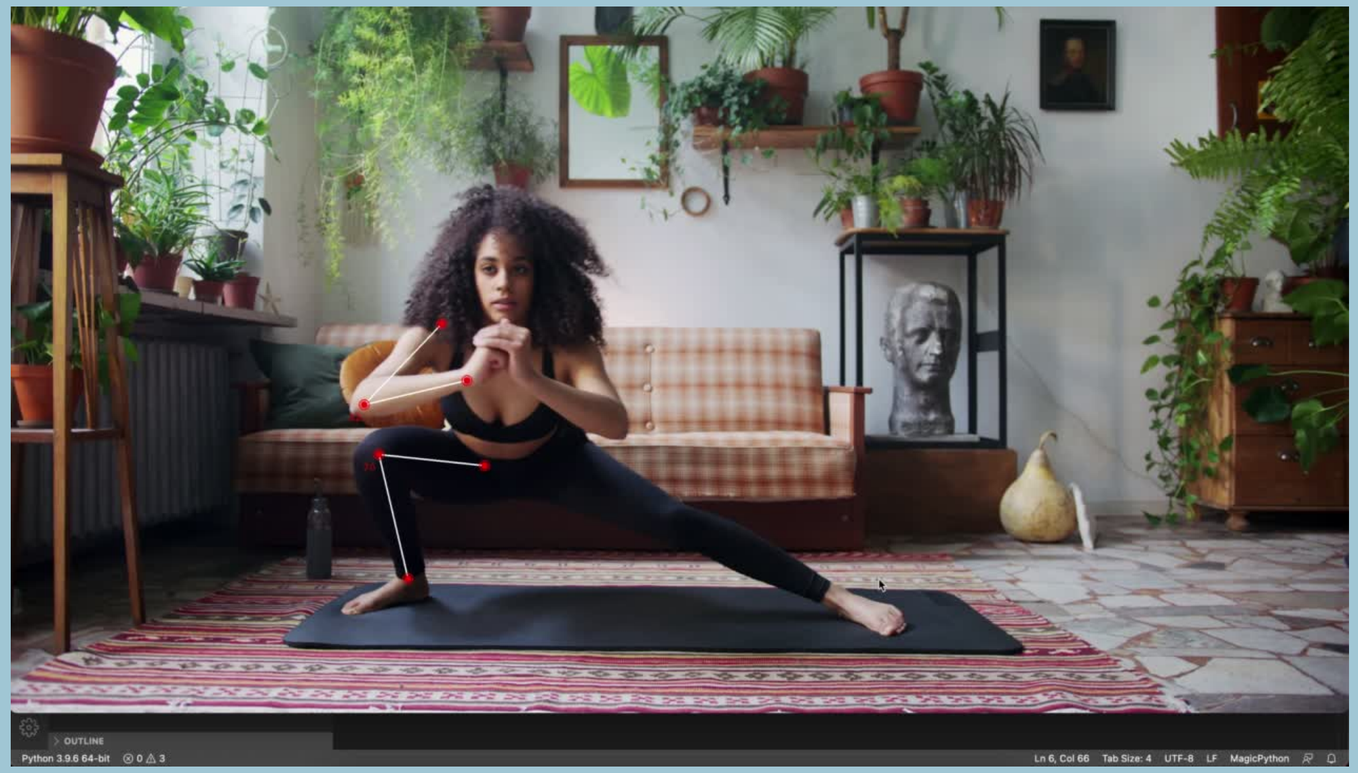
First Example

No overlays, easy and
accurate example

- Personal example
- Joints detected properly
- No major errors



Second Example High resolution



- Example with high resolution
- Detections at 30 FPS
- Filmed from the front
- A few errors started to show

Third Example

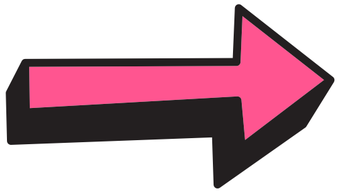
With muscle overlay,
messy example

- Also a personal example
- Filmed with an angle similar to the first example
- Not a study shot
- Muscle overlay
- Important body parts may not show properly

Main problem and solution

Low performance Model

Low accuracy with muscle overlap.
When the joint isn't properly detected, the angle is completely wrong



Low performance Model

There isn't a solution in this case. It's almost impossible to create a better model.
Different cases may not need such high accuracy

Oportunities

Physiotherapy

Help patients in recovery

Apps and Programs

Software to help with training

Posture correction

Posture monitoring and observation

Body Language

body language detection

Body language decoder

01

MediaPipe models

- Detect body
- Face Mesh
- Hands

02

Dataset

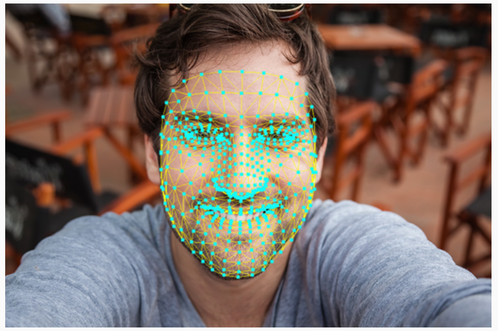
Build a dataset from scratch

03

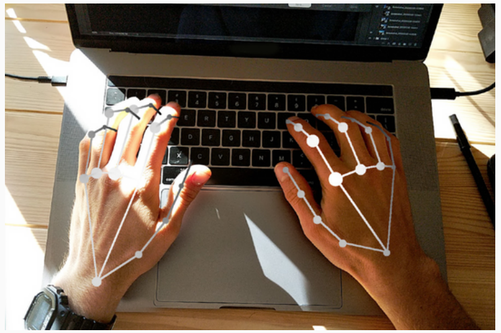
Custom ML model

Body Language classification model

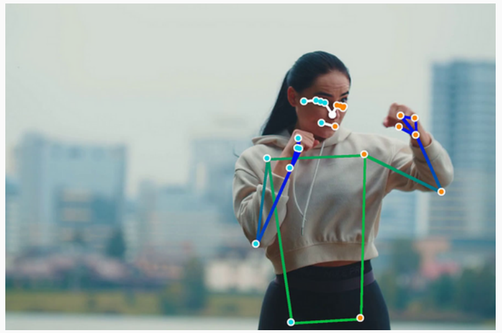
Detection Models



Face Mesh



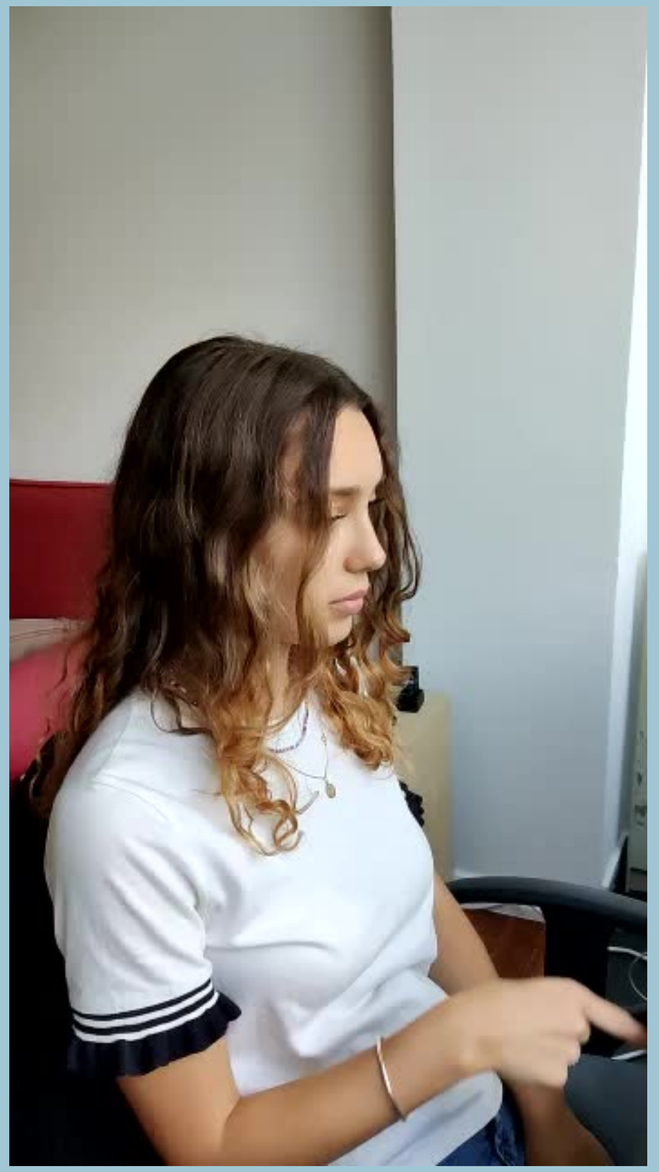
Hand Tracking



Pose Detection and Tracking

Building a Dataset

- Used a notebook
- Opencv to open a stream from the webcam to the computer
- While it's recording, it runs all detection models in background and those coordinates plus the class on a csv.saves each frame

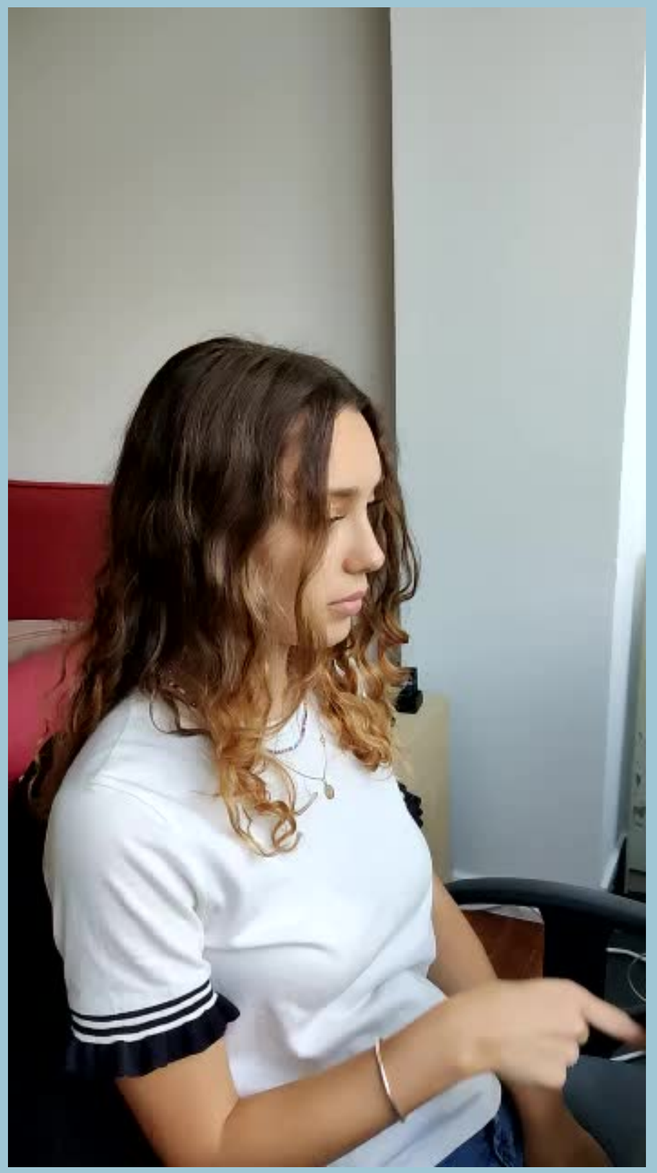


Building a Dataset

- Joana moves around the camera, changes positions but keeps the same posture.
- Trying to create diversification

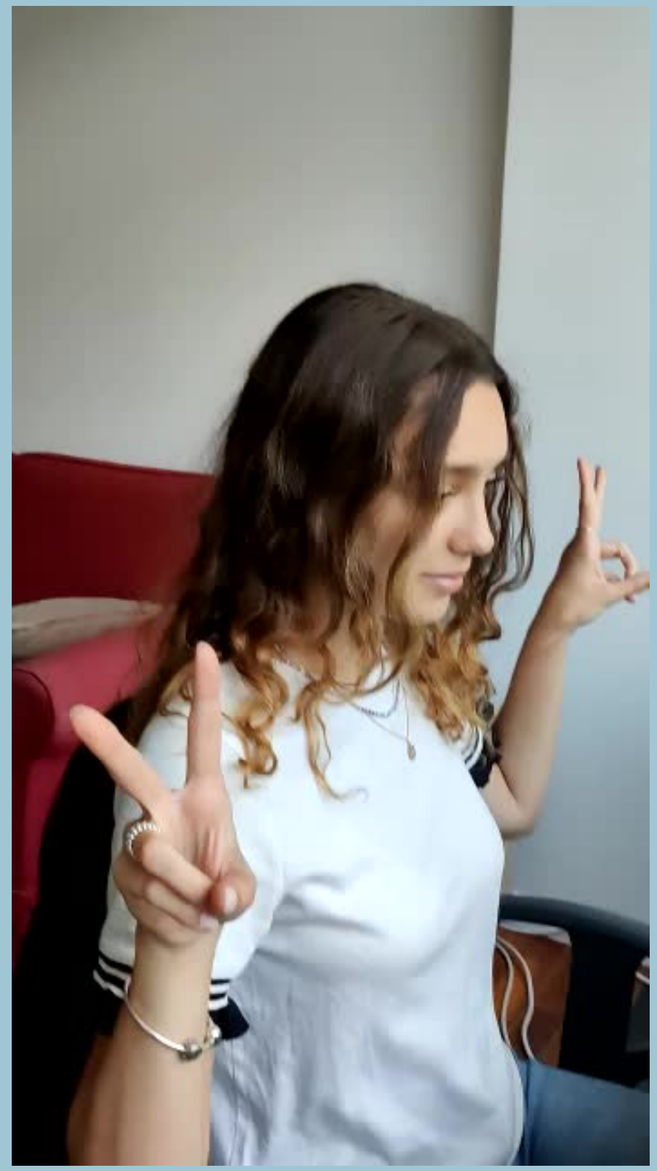
To improve diversification:

- Standing up
- Different people
- Move more



Building a Dataset

- Very difficult to create an accurate dataset with thousands or millions of records with a low percentage of outliers



Classification model

Baseline Models Experimented

- RidgeClassifier
- RandomForestClassifier
- GrandientBoostingClassifier

How to improve accuracy

- Convolutional Neural Network (CNN)

Advantages
Simple baseline

Disadvantages
Hard to distinguish different postures due to a high number of imputs

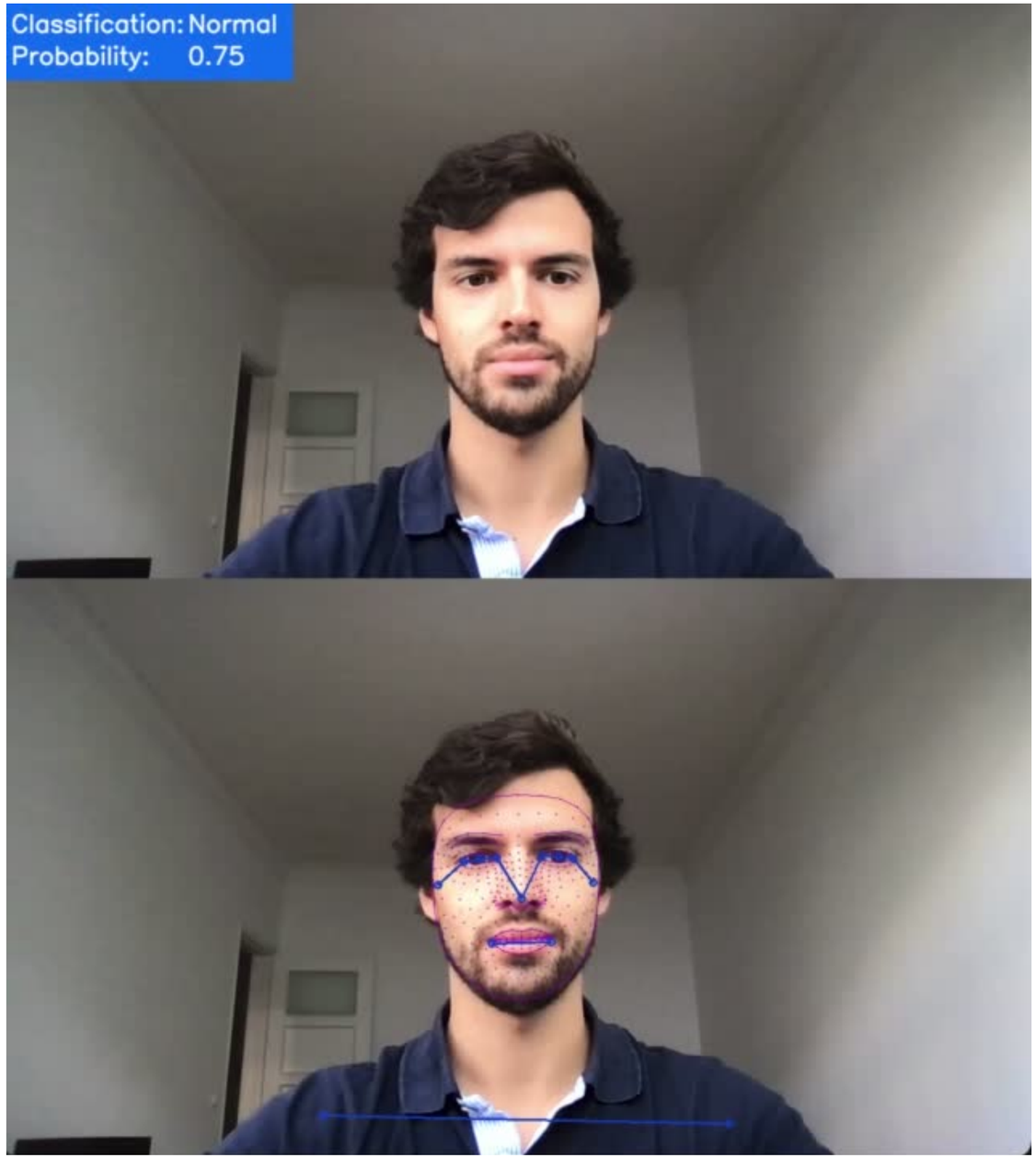
First Example

Classes

- Normal, Happy, Hello, Schocked, Be Quiet, Flex, Peace

Results

- Able to predict 'accurately' in a controlled environment
- Difficulty in distinguishing 'Normal' from 'Happy'
- Easy to identify very different classes (Flex with both arms up, Hello with a single hand up, etc)



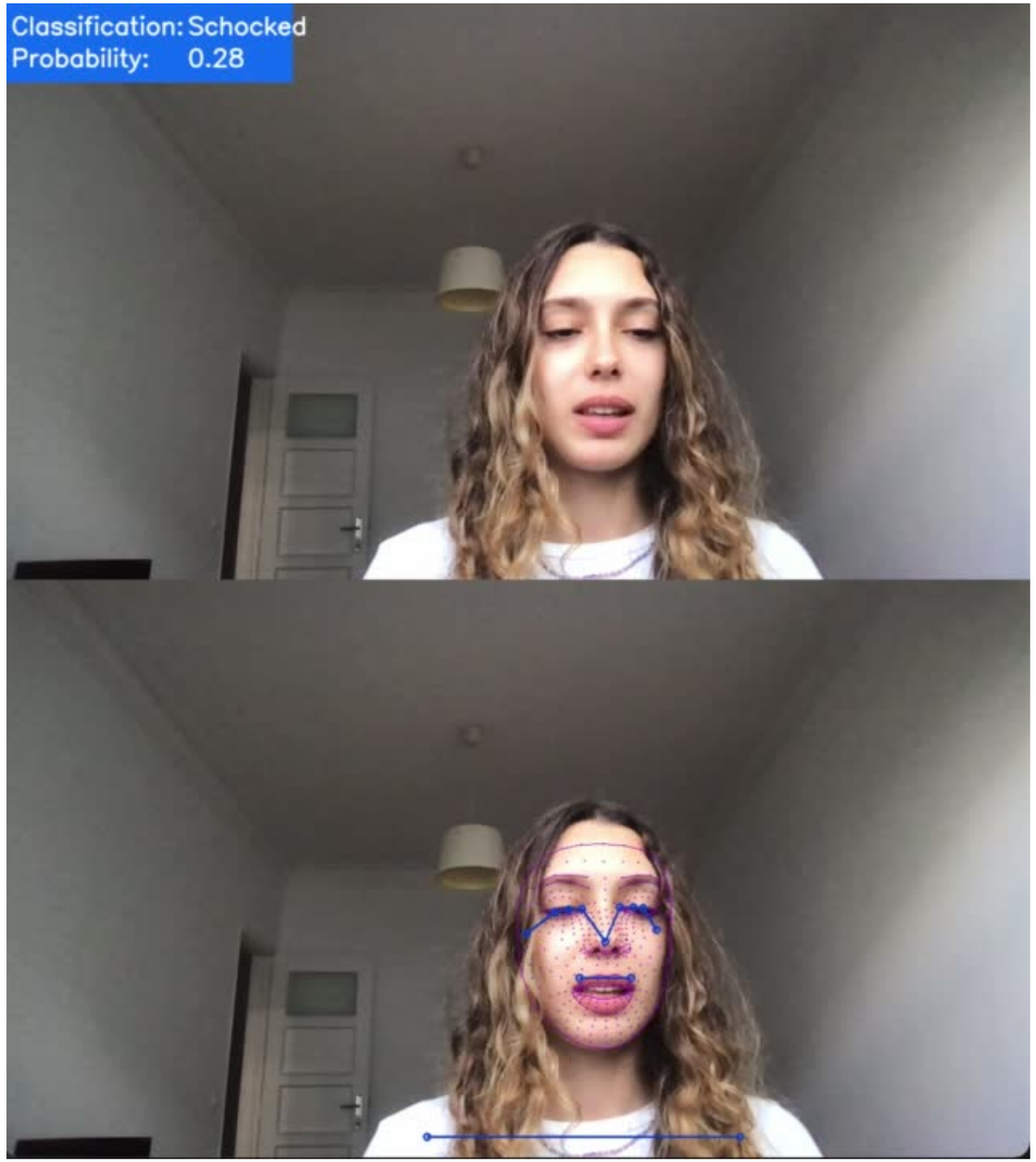
Second Example

Classes

- Normal, Happy, Hello, Schocked, Be Quiet, Flex, Peace

Results

- Poor dataset quality
- Completly random results



Main obstacles

01

Dataset

Small
Low-quality
Few examples
Low diversification

02

Model

Used a simple (baseline) model

03

Inputs

Face landmarks (468)
Hand landmarks (21)
Pose landmarks (33)
1566 points (x,y,z)

04

Complex Problem

Overall too many difficulties

Possible Solutions

01

Dataset

Time = Quality
Bigger dataset
Be more careful with outliers
More diversification

02

Model

A robust neural network

03

Inputs

Use fewer points from Face Mesh

04

Complex Problem

Devide and conquer

Space Management

Control and supervision physical spaces

People Counting

Challenge

- Know exactly how many people are inside a space. And how many entered and exited

Customer Journy

Challenge

- Know where people went, where they stoped, heat maps, etc.

Space Managment

People Counting

Number of people in and out, current ocupation

People Segmentation

Gender, age etc

Customer insights

Customer journey, customer density etc

Cashierless Check-out

Amazon Sensei

Thank you!

joao.ramos@starkdata.ai