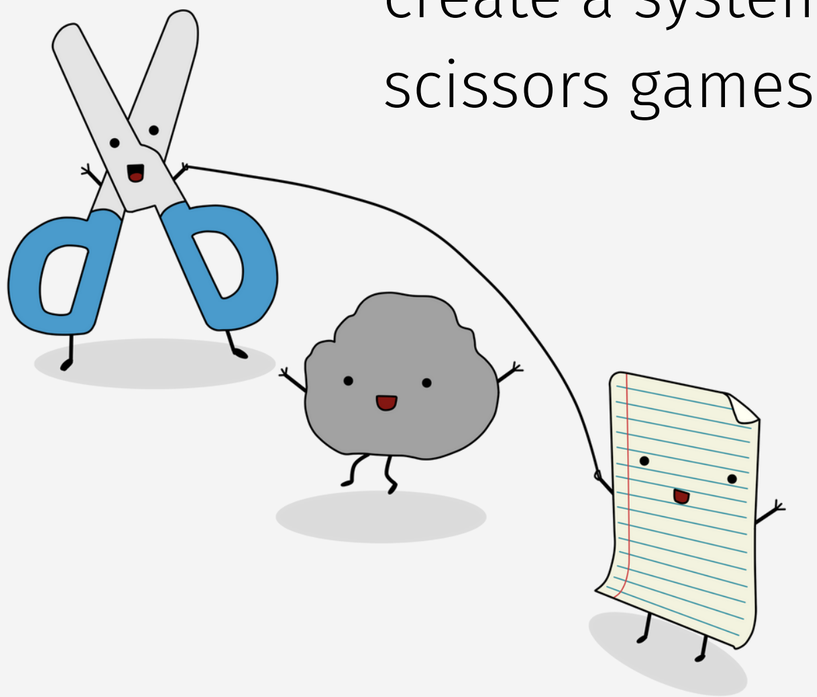
Davide Mor
Fabio Turchetta

Rock - Paper - Scissors

Combine gesture recognition and user identification to create a system capable of understand rock-paper-scissors games



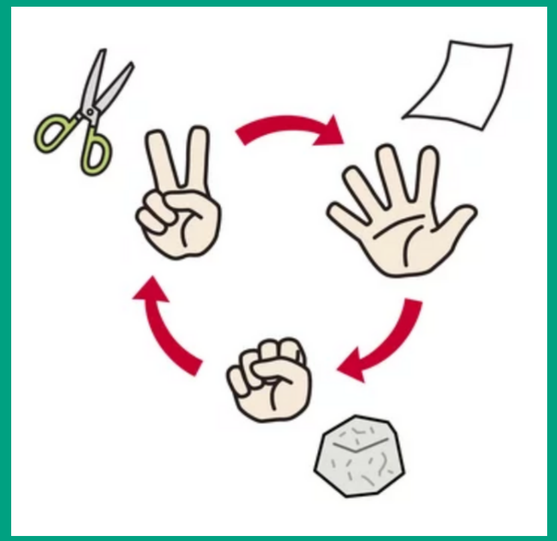
Intelligent Consumer Technology Project

Introduction

Our goal is to create a system able to classify hand’s positions in the gestures of a rock - paper - scissors game.

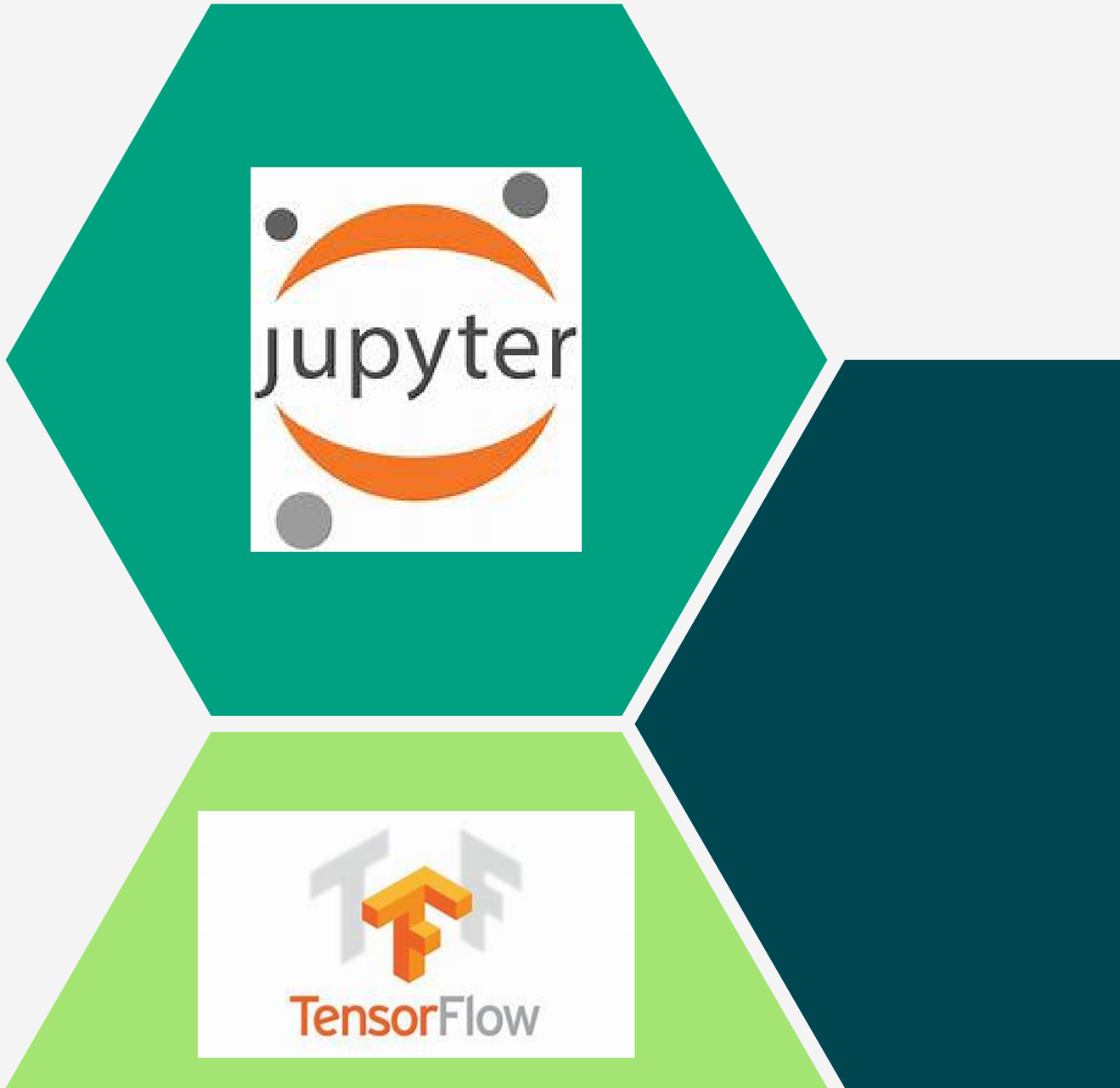
The system should be able to understand which user is playing and who is winning.

All of that must be done real time using webcam images.



Environment

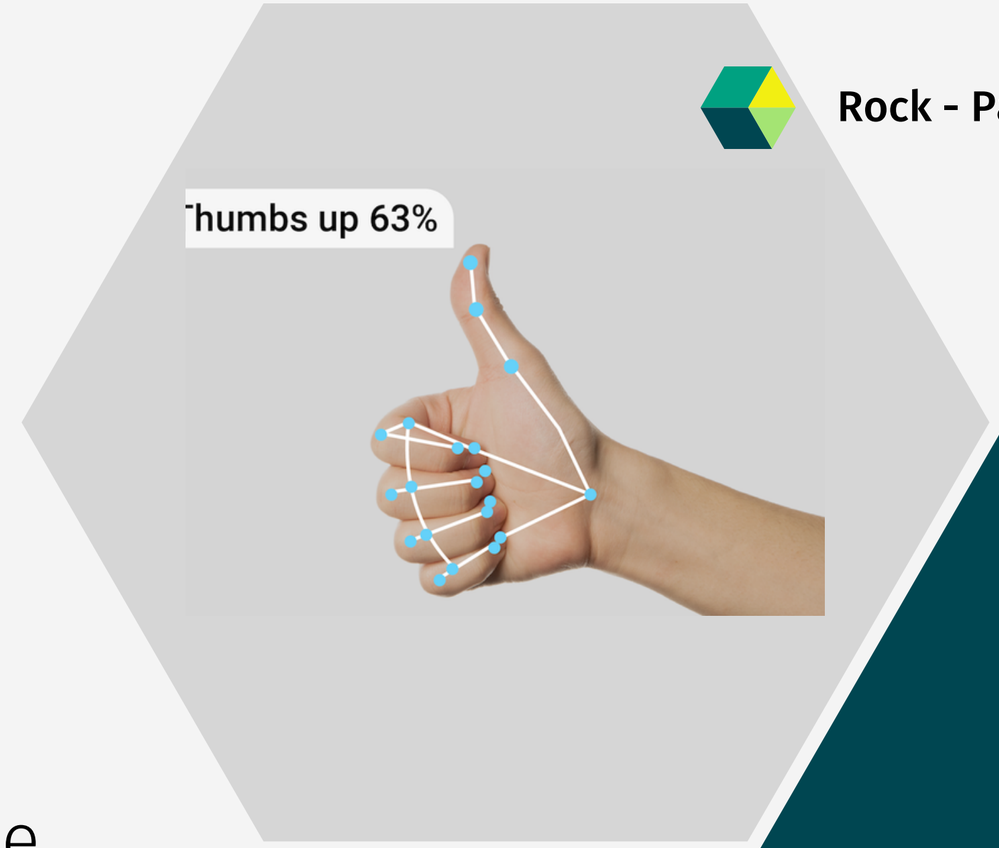
- Workspace environment used: Jupyter Notebook by Anaconda
- Device used for computation: CPU
- Library for developing machine learning models: TensorFlow
- Device for real-time information acquisition (images): computer camera.

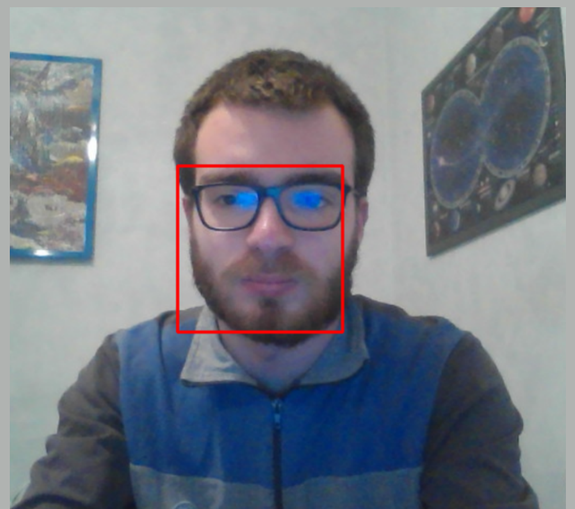


Work division

The project is divided in two smaller parts:

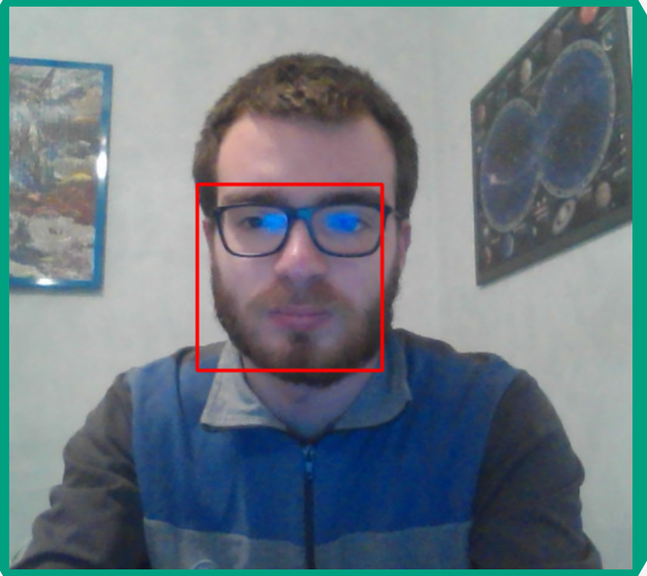
- the gesture must be recognized and the logic of the game must be implemented
- users must be detected and recognized





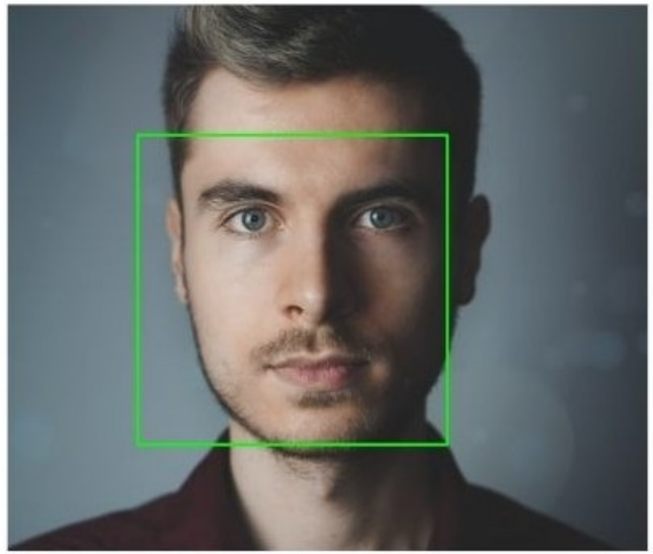
Face detection

- how to detect a face?
- how to track a face in a video stream?

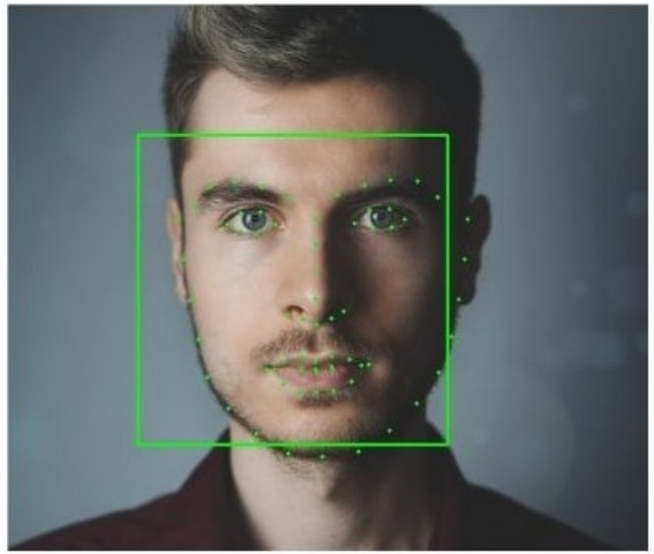


Face detection

We decided to use dlib’s frontal face detector to process images to automatically get face positions. Dlib library also provide a second model for landmark estimation.

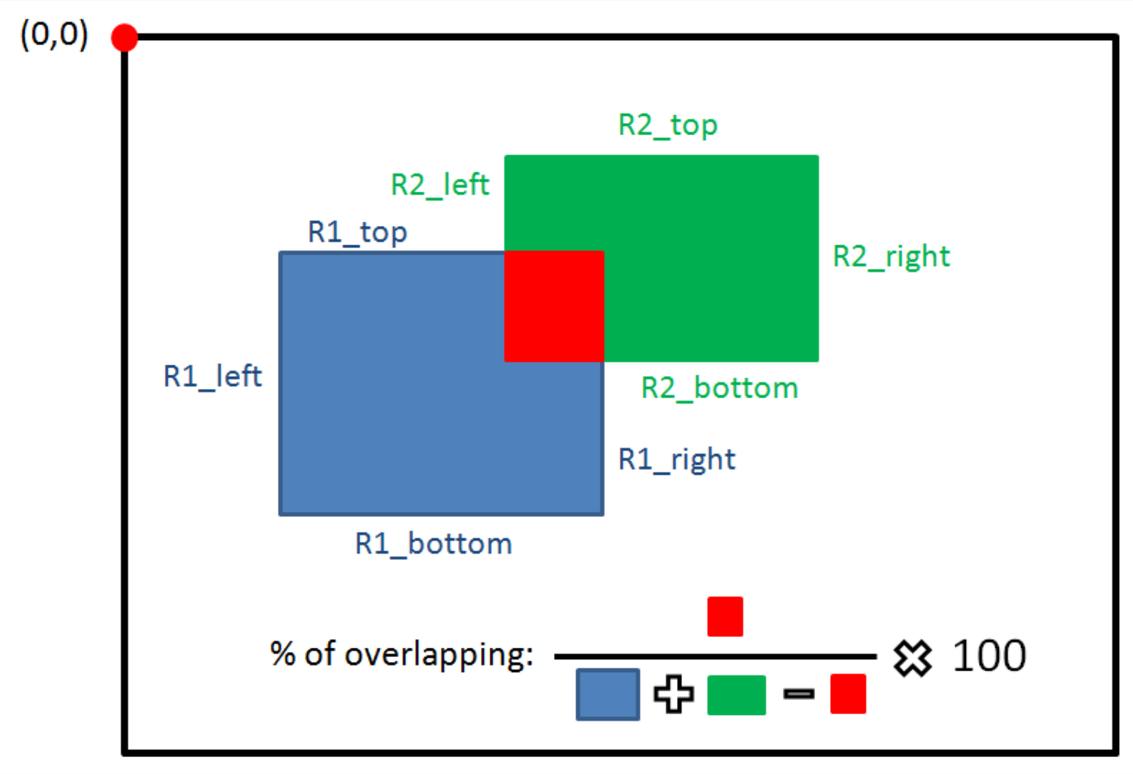


Step 1
Face Detection



Step 2
Landmark Detection

Face tracking



If we find a face in the same position or close enough to one detected in the previous frame we can give as granted that the two faces belong to the same user.

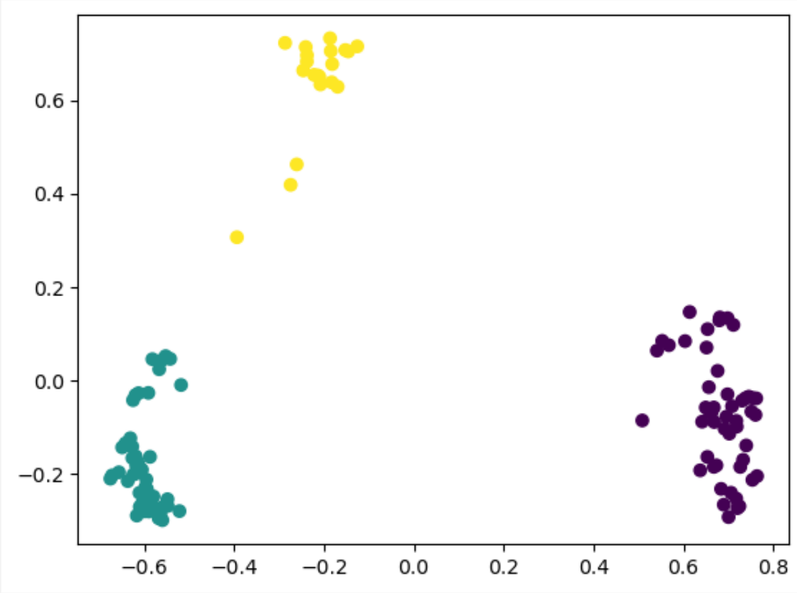
To compute how much two faces are close each other we used the percentage of overlapping of the bounding boxes.

Alignement

- must be performed for each face for which we want compute the embeddings
- we must compute the 5 landmark needed starting from the 68 given by dlib’s model

Feature extraction

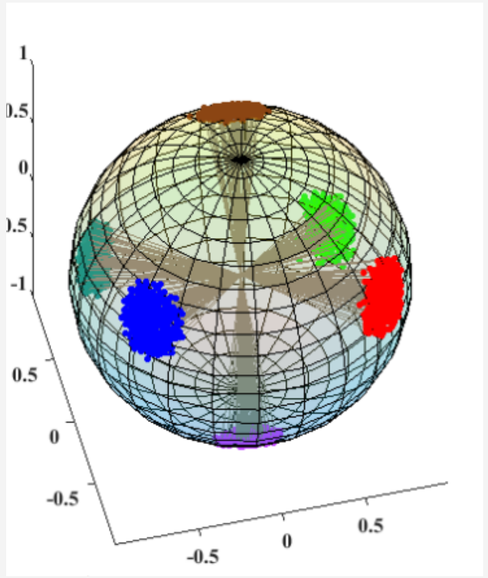
The aligned face is used as input for the spheereface model to obtain an high dimensional representation of the face.



principal component analysis of normalized multidimensional representation of faces of 3 users

User recognition

Then we compute the cosine similarity between the face embeddings and the mean of embeddings associated to each registered user.



If there is a user for which we have an high similarity (>0.7), we conclude that the face in the foto belong to that registered user.

Register a new user

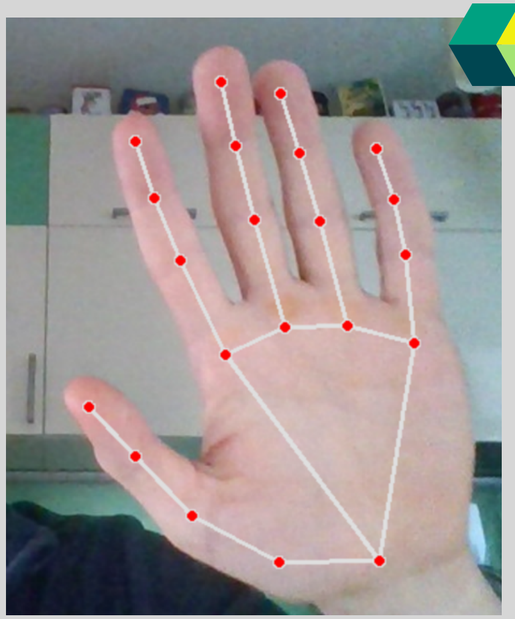
If there is no registered user with high similarity to the detected face we wait a few frames to collect more embeddings, if still there is no user that match the mean of the observation we proceed by adding a new user to the list.

The collected embeddings are used as data for future comparisons.

Gesture classification

Pipeline:

- Hand recognition
- Gesture recognition



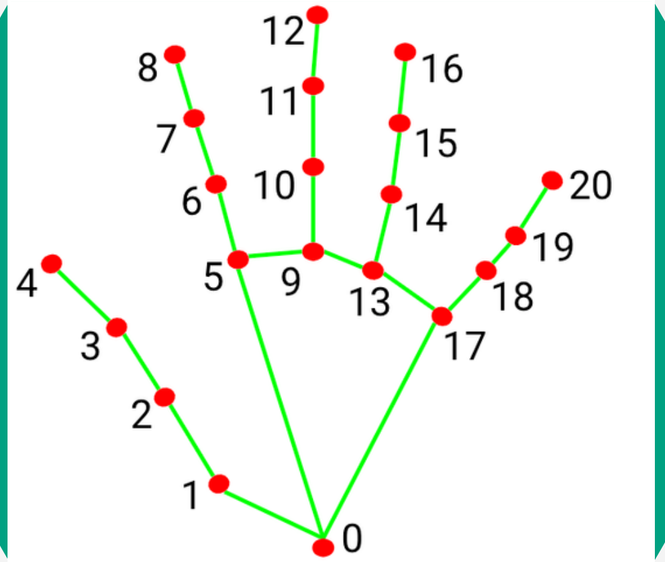
Hand recognition

Gesture recognition

Hand recognition

Refers to the identification and localization of hands in an image or video.

The library used is MediaPipe, which has a model that identifies hands and returns the positions of 21 landmarks.

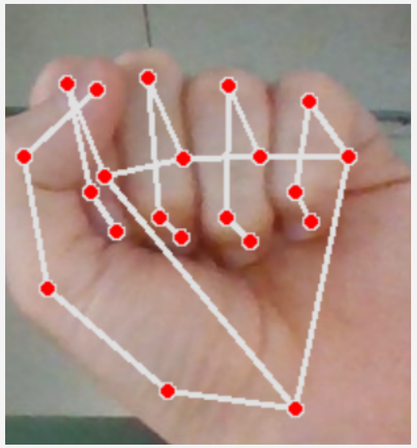


MediaPipe landmarks

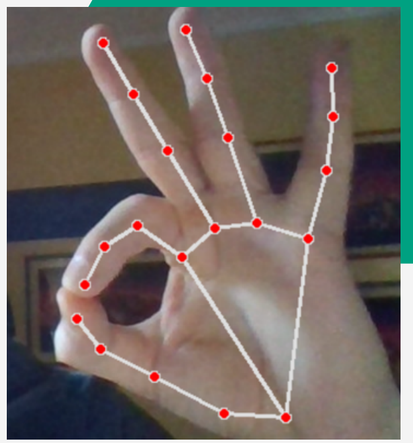
MediaPipe

Data collection

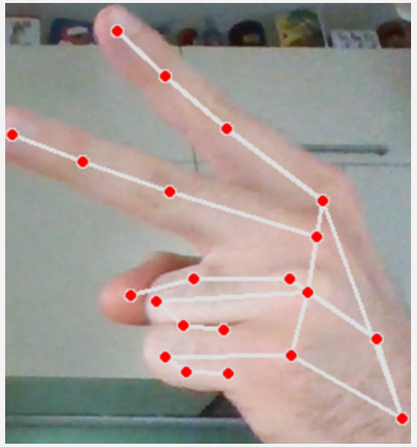
- Personally collected dataset
- 4 classes (rock, paper, scissors, ok)
- 400 hand landmarks for each class



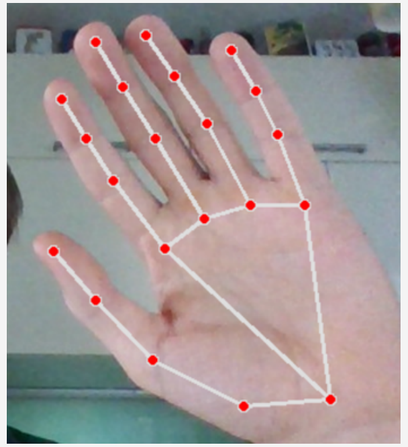
Rock



Ok



Scissors



Paper

Data preprocessing

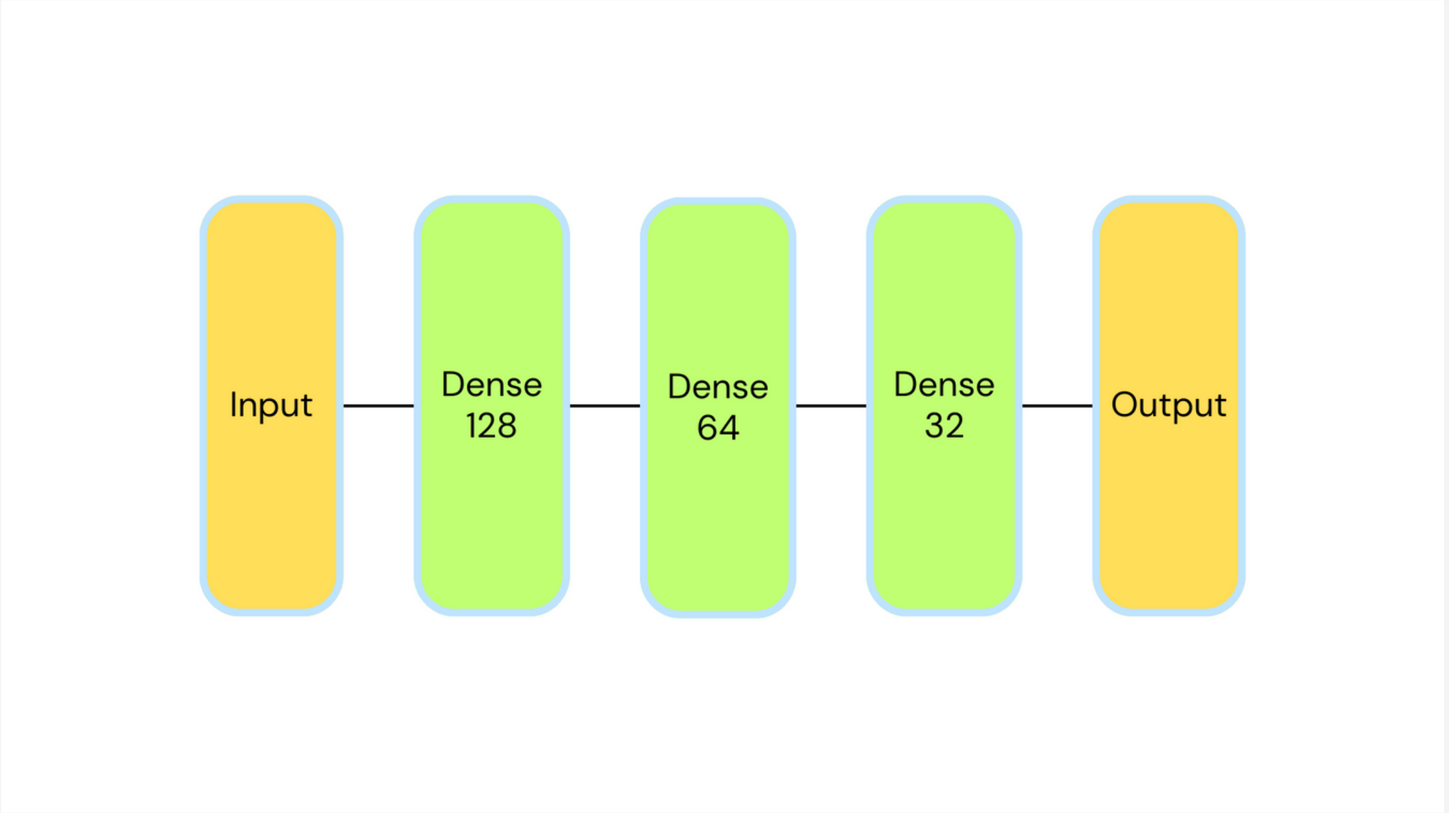
Transformation of the landmark list into an array of size 210 corresponding to the pairwise distances of each landmark (normalized).

Train/Test/Evaluation split: respectivley 80%/10%/10%

Gesture recognition

- Through an MLP model
- Input: processed data
- Output: scores for each class

Gesture recognition model

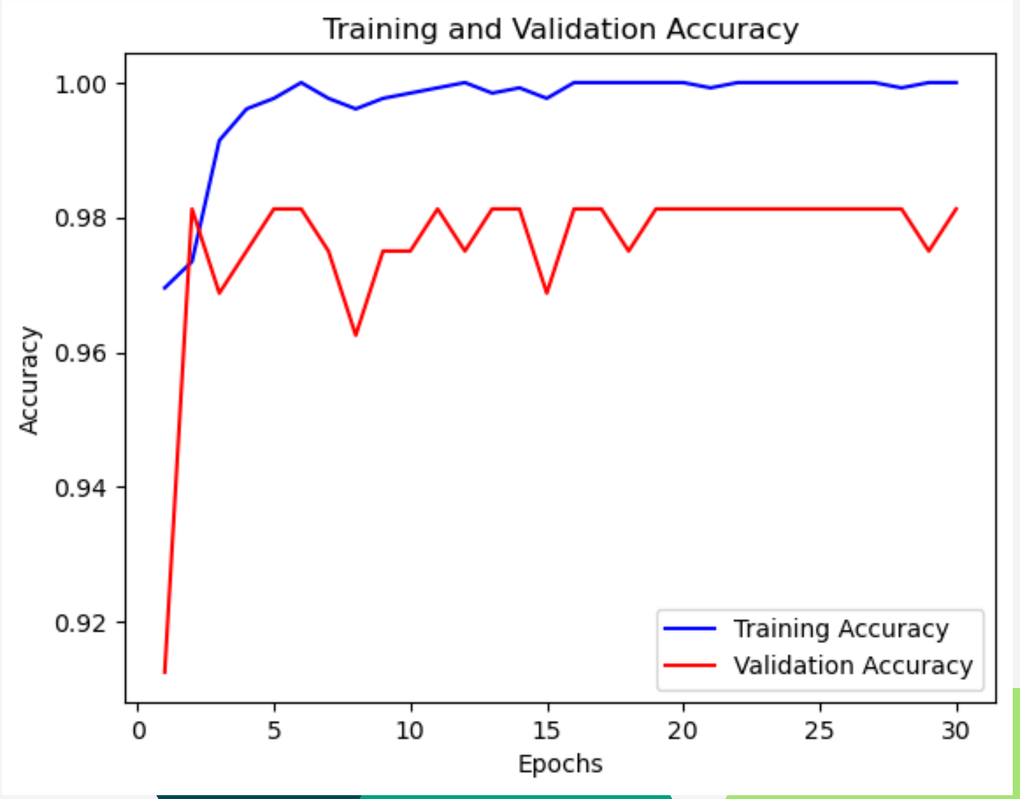


Activation functions:

- Layer 1/2/3: ReLu
- Classification Layer: Softmax

Model Training

- Optimizer: Adam
- Loss: Sparse categorical crossentropy
- Metric: Accuracy
- Batch size: 128
- Learning rate: 0.001
- Epochs: 30

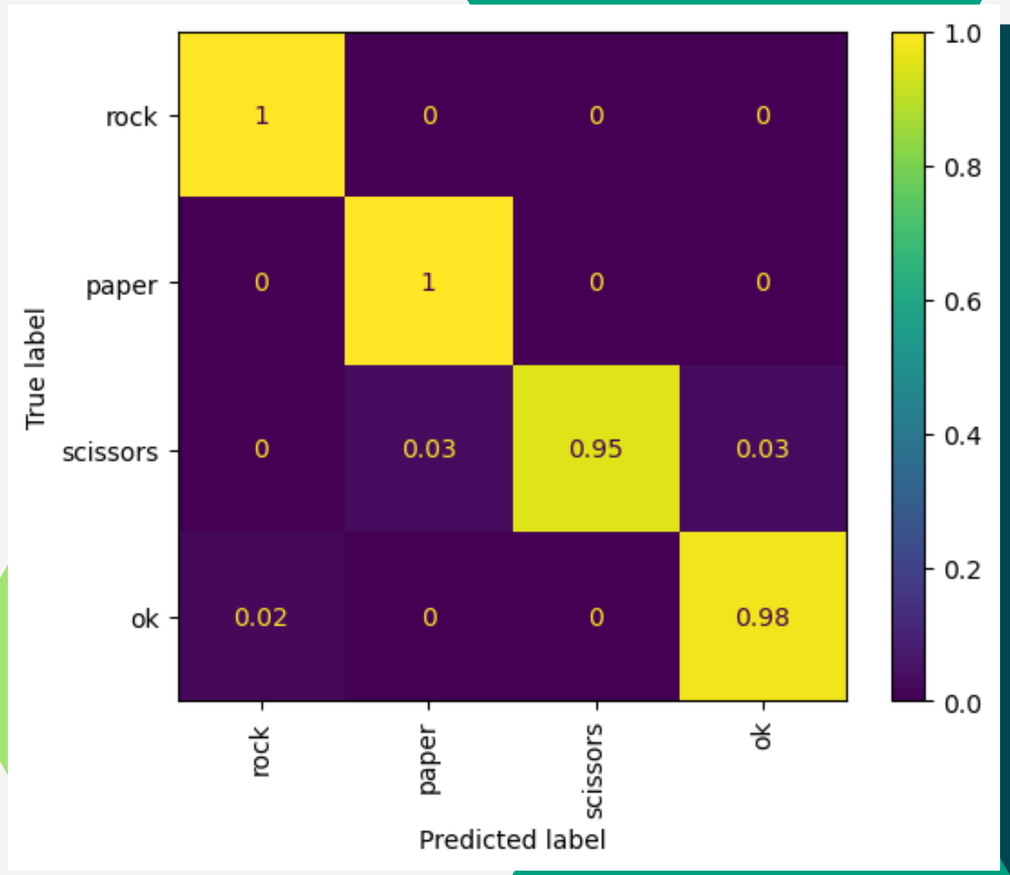


Training results

Test accuracy: 0.981

	precision	recall	f1-score	support
rock	0.98	1.00	0.99	45
paper	0.97	1.00	0.99	36
scissors	1.00	0.95	0.97	37
ok	0.98	0.98	0.98	42
accuracy			0.98	160
macro avg	0.98	0.98	0.98	160
weighted avg	0.98	0.98	0.98	160

Classification report



Confusion matrix

Experiments:

Model2: MLP model that receives as input angles between selected triplets of landmarks

Model3: Mixed MLP model that receives 2 inputs (distances and angles between landmarks) and after passing them through linear layers, concatenates them before passing them to additional linear layers.

Comparison of model performance (accuracy)

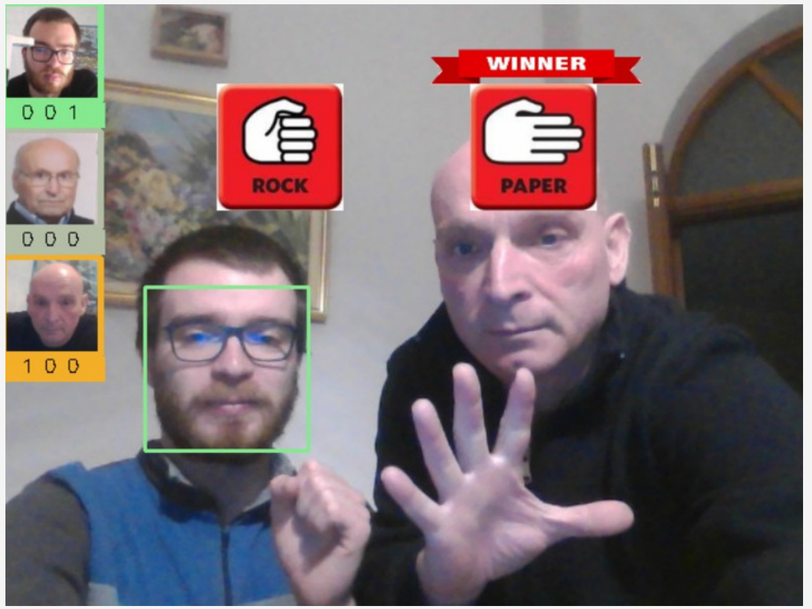
	Model	Model2	Model3
3 Layers	0.975	0.894	0.968
4 Layers	**0.981**	0.906	**0.981**
5 Layers	0.962	**0.919**	0.956

Game

Phase 0: Wait for both players to be ready (OK position), and save the hand (right or left) used to play.

Game

Phase 1: The players make their own move, and the game understands the move made.



Phase 2: Determines the winner, and updates the score.

Phase 3: Players can redo the start signal (Ok) to play a new game.

Problems encountered

- no acces to a GPU
- best and more robust models are also the slowest
- in the screen may appear more or less hand than expected
- old user data may not be reliable

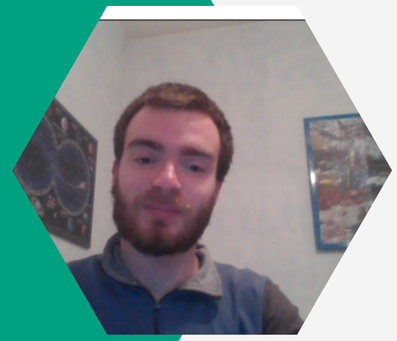
Conclusions:

- The hand classification pipeline offers excellent performance. The use of reciprocal distances from landmarks provides advantages because it allows for generalization regarding the hand's position in the image and the background (such as the face recognition model) or environmental conditions such as lighting.
- The front-facing camera as the sole point of view occasionally presents limitations since requires staying within the frame and the face must front the camera; it also and hands and fingers may not be fully visible.

References:

- A Computational Approach to Hand Pose Recognition in Early Modern Paintings; Valentine Bernasconi, Eva Cetini and Leonardo Impett
- MediaPipe Hands: On-device Real-time Hand Tracking; Fan Zhang, Valentin Bazarevsky, Andrey Vakunov, Andrei Tkachenka, George Sung, Chuo-Ling Chang, Matthias Grundmann
- SphereFace: Deep Hypersphere Embedding for Face Recognition. Authors:Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, Le Song.
- Joint Face Detection and Alignment using Multi-task Cascaded Convolutional Networks, Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, Yu Qiao

Thank you for attention



Davide Mor
d.mor1@campus.unimib.it

Fabio Turchetta
f.turchetta@campus.unimib.it