Recent Research

Prof. Maria De Marsico and her
«distributed» group

Handwriter recognition (also in forensics!)

Comparison of two architectures for text-independent verification after character-unaware text segmentation

Graduation work by Mohammadreza Shabani

CONTRIBUTIONS

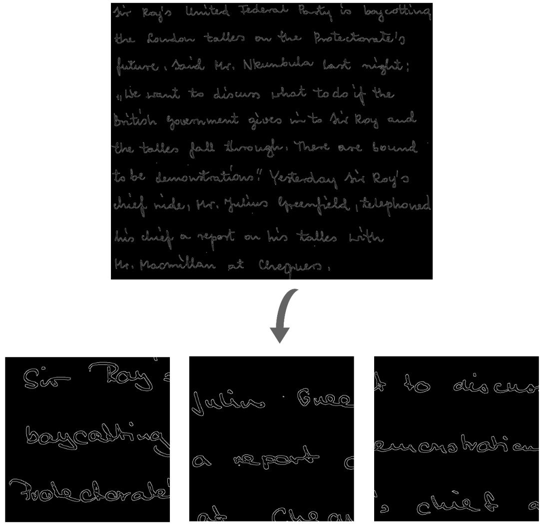
- **Character-unaware text segmentation**: faster and reflecting both horizontal and vertical co-articulation of the writing sign
- **Train test split adapted to forensic medium-long term writer verification**: experiments presented so far often train and test on writing patches taken from the same documents, which is not suited for behavioral biometric traits that lack permanence
- **Comparison of two fine-tuned deep architectures:** different complexity for possible use on mobile devices

FUTURE

- **More data for better evaluation of generalization**
- **Experiments with different writing systems**

Handwriter recognition (also in forensics!)

1)



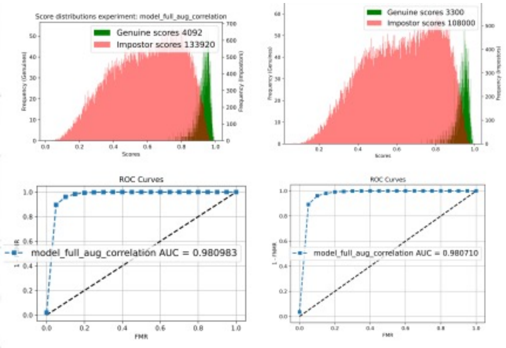
2)

- **Train test split adapted to forensic medium-long term writer verification**: takes into account that behavioral biometric traits lack permanence
  - training and testing sets contain patches coming from rigorously different documents
  - gallery and probe sets contain patches coming from rigorously different documents
    - ✓ pros: allows evaluating verification accuracy versus time variability characteristic of behavioral traits
    - ✓ cons: dramatic decrease of useful data (only 31 writers with at least 9 forms split in training and testing)
  - two strategies for a preliminary generalizability test: 31 training vs 31 testing – 25 training vs 31 testing

	Dataset	#ID	Forms per ID	Cropped Samples	Total
A	Train	31	5	8 (overlap)	1240
	Test	31	4	3 (no overlap)	372
B	Train	25	5	8 (overlap)	1000
	Test	31	4	3 (no overlap)	372

3)

MobileNetV2

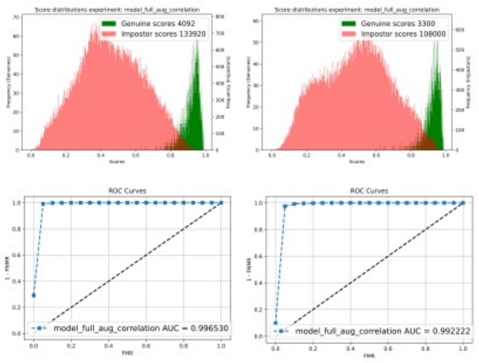


Training set with all 31 subjects

Training set with 25 subjects

	Similarity	GMEAN	GSTD	IMEAN	ISTD	AUC	EER	ZeroFMR	FMR1000	FMR100
A	correl.	0.93	0.04	0.59	0.20	0.98	0.07	0.98	0.75	0.39
	cosine	0.95	0.02	0.74	0.13	0.98	0.07	0.98	0.75	0.40
B	correl.	0.93	0.03	0.62	0.19	0.98	0.068	0.96	0.77	0.40
	cosine	0.96	0.02	0.75	0.13	0.98	0.07	0.97	0.78	0.40

ResNet-50



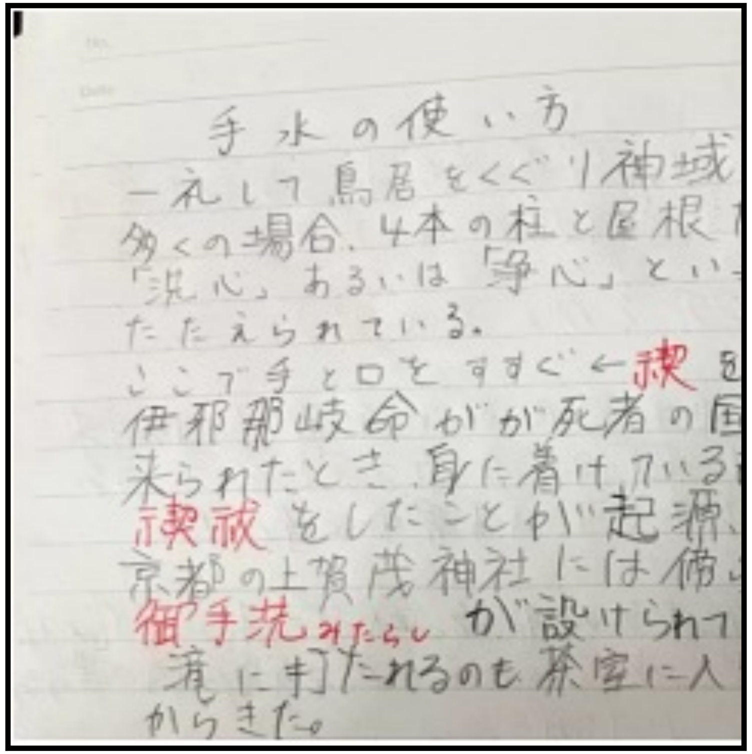
Training set with all 31 subjects

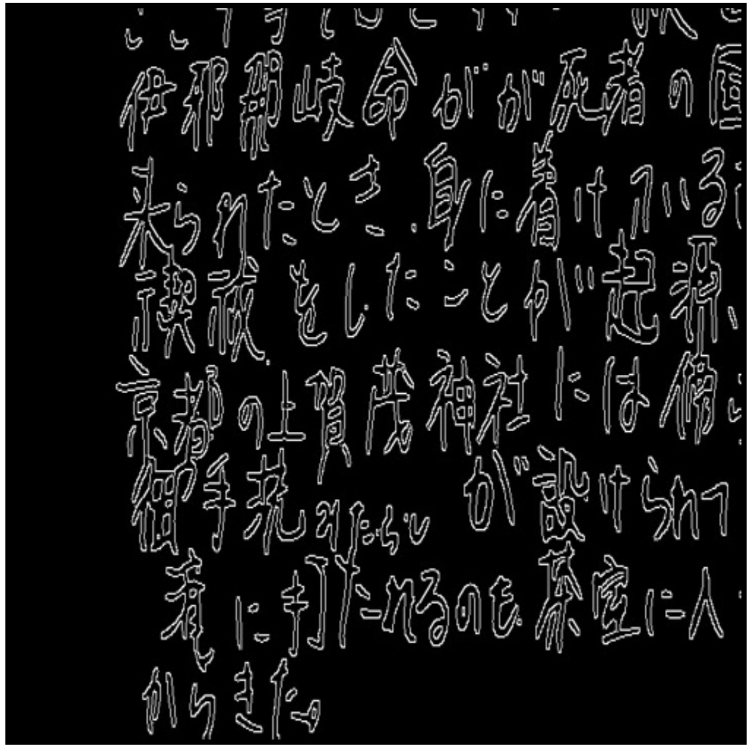
Training set with 25 subjects

	Similarity	GMEAN	GSTD	IMEAN	ISTD	AUC	EER	ZeroFMR	FMR1000	FMR100
A	correl.	0.92	0.05	0.46	0.18	0.99	0.03	0.71	0.33	0.09
	cosine	0.95	0.03	0.63	0.13	0.99	0.03	0.71	0.34	0.09
B	correl.	0.92	0.04	0.48	0.20	0.99	0.04	0.90	0.61	0.20
	cosine	0.95	0.03	0.66	0.13	0.99	0.04	0.89	0.61	0.20

Handwriter recognition (also in forensics!)

EXTENSION

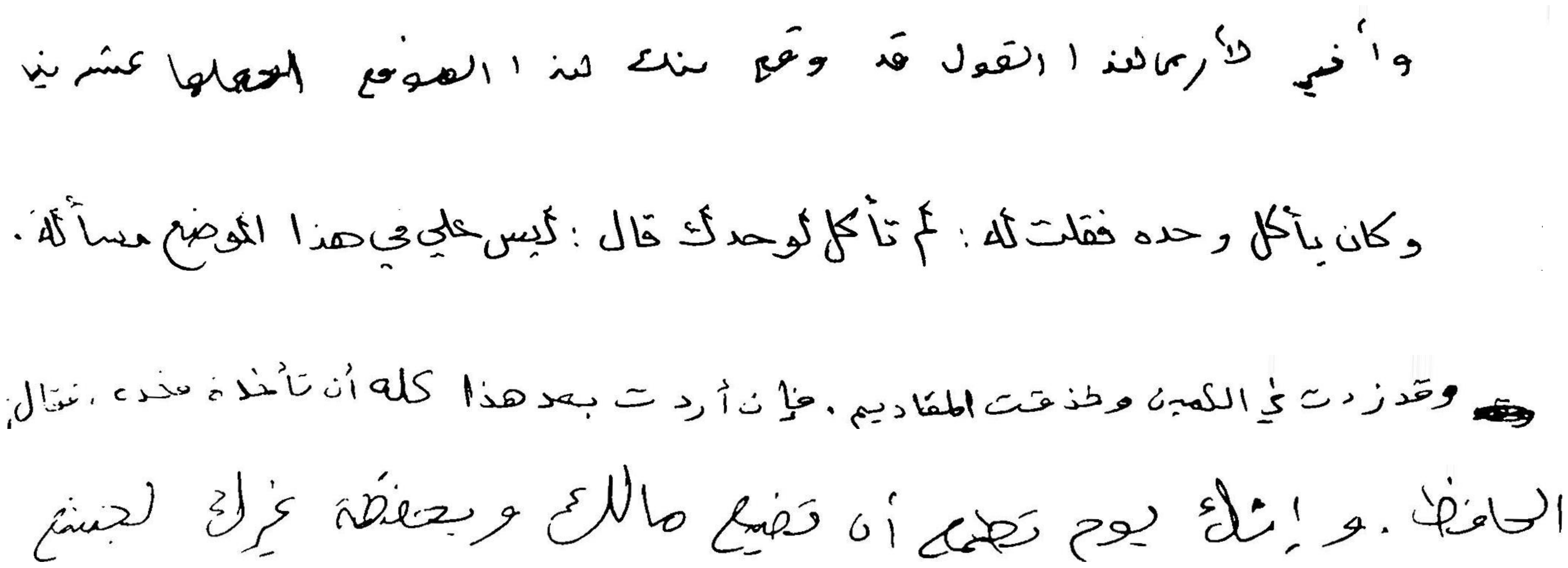




Mixed Kanji

Handwriter recognition (also in forensics!)

EXTENSION



Arabic

Handwriter recognition (also in forensics!)

Two approaches

FULL: all samples mixed together both in training and testing

SELECTIVE: script identification first and then subject recognition

ONGOING

Face expression analysis: emotions

How we can recognize emotions:

- facial expressions
- body and postural changes
- wearable sensors (blood pressure, heart rate, temperature)
- remote photoplethysmography to evaluate microvariations in blood flow

According to some approaches that study basic emotions, there are prototypes associated with specific emotions.

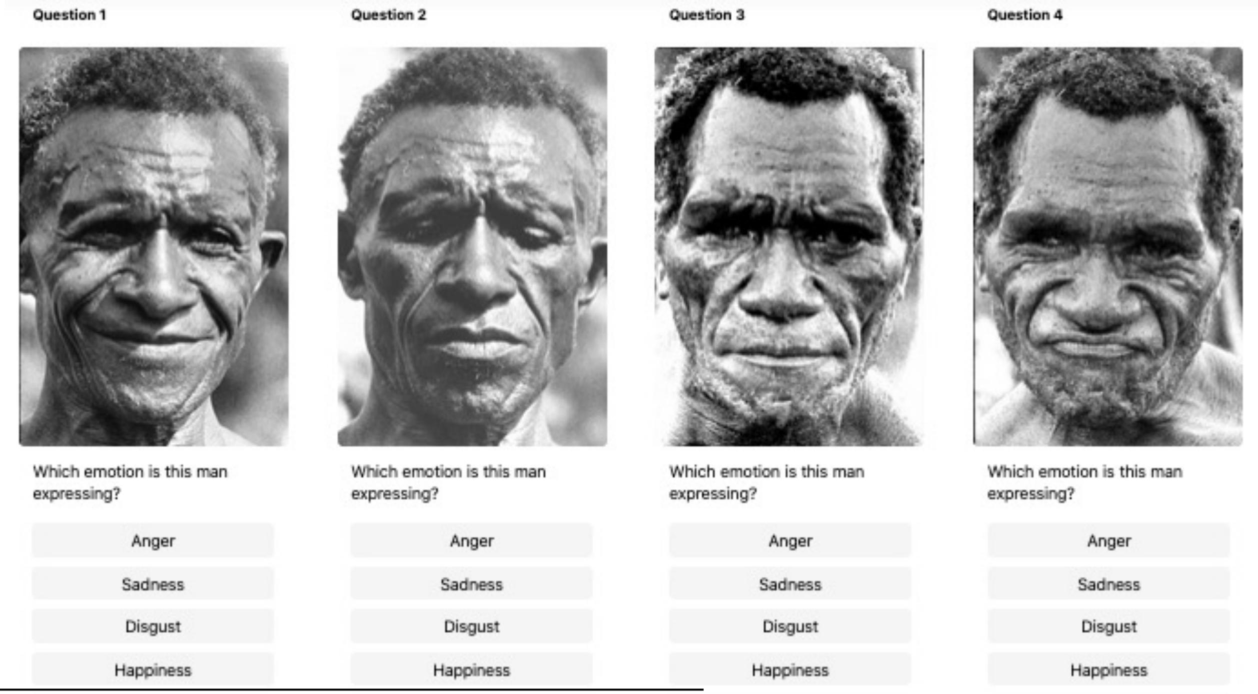
Face expression analysis: emotions

- Ekman, Woodworth, and Izard are among the main scholars of facial expressions as manifestations of emotions.

- In the experiments conducted, regardless of the culture of belonging, the facial expression of a well-defined emotion was universally recognized in its specificity.

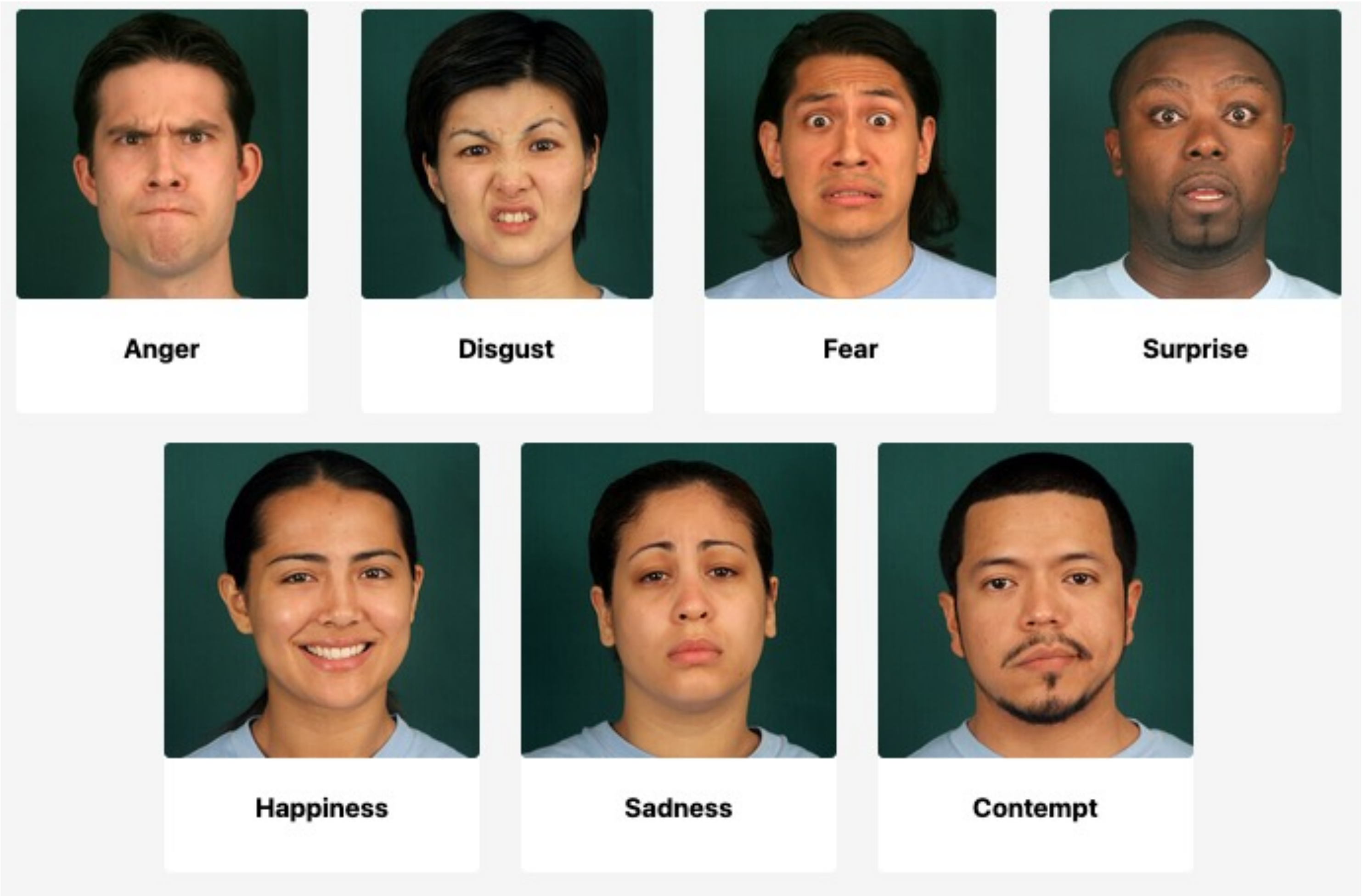
Face expression analysis: emotions

- This New Guinea man lived in an isolated, preliterate culture that used stone tools that no outsider had ever seen before. Dr. Paul Ekman asked him to show what his face would look like if: (1) friends had come. (2) his son had just died. (3) he was about to fight. (4) he had stepped on a dead, smelly pig.



From: https://www.paulekman.com/resources/universal-facial-expressions/

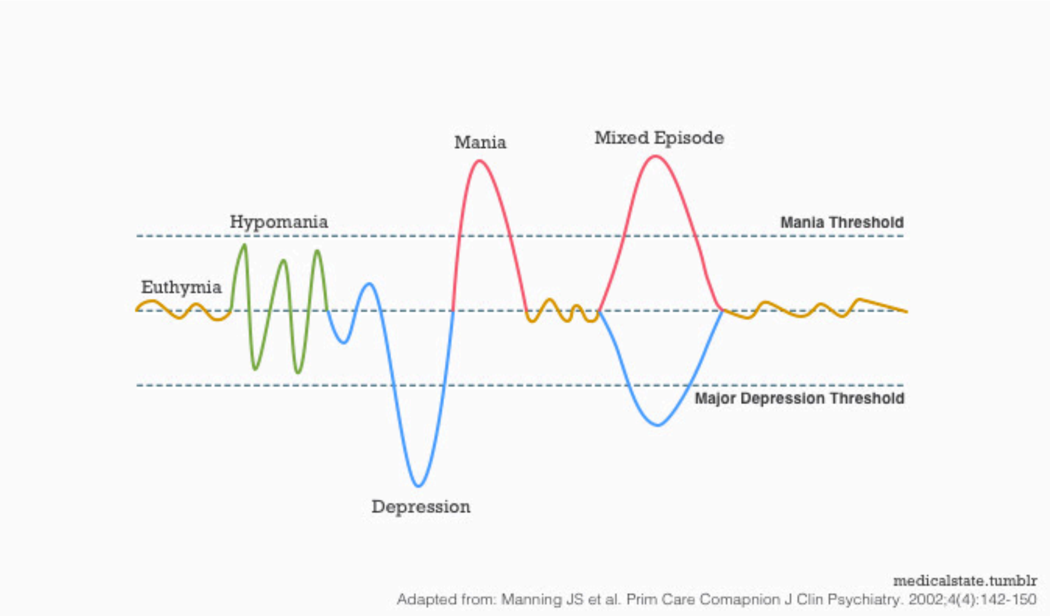
Face expression analysis: emotions



From: https://www.paulekman.com/resources/universal-facial-expressions/

Face expression analysis: emotions in healthcare

- **Mental Health Treatment.** Facial emotion recognition can help monitor the emotional state of patients, for example those undergoing mental health treatment, in cases of Serious Mental Illness (SMI) such as bipolar disorder or schizophrenia. It can be used to monitor mood changes and help identify patterns that may be relevant for treatment.

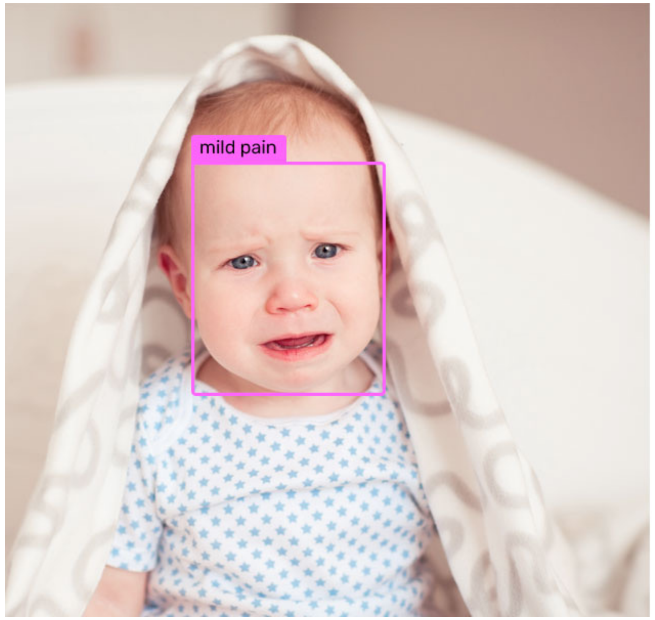


Face expression analysis: emotions in healthcare

- **Diagnosis and monitoring of illness.** Some illnesses and conditions (e.g., depression and anxiety or in cases of Major Depressive Disorder (MDD) can manifest with changes in facial expressions and body language. Recognition of facial emotions can help detect and identify these changes and aid in diagnosis and monitoring the effectiveness of treatment.

Face expression analysis: emotions in healthcare

- **Detection and management of pain or discomfort.** Detection and interpretation of facial expression can be of great help in diagnosing pain in patients, especially those unable to communicate verbally (newborns, patients in coma?)



Approaches to facial expression recognition

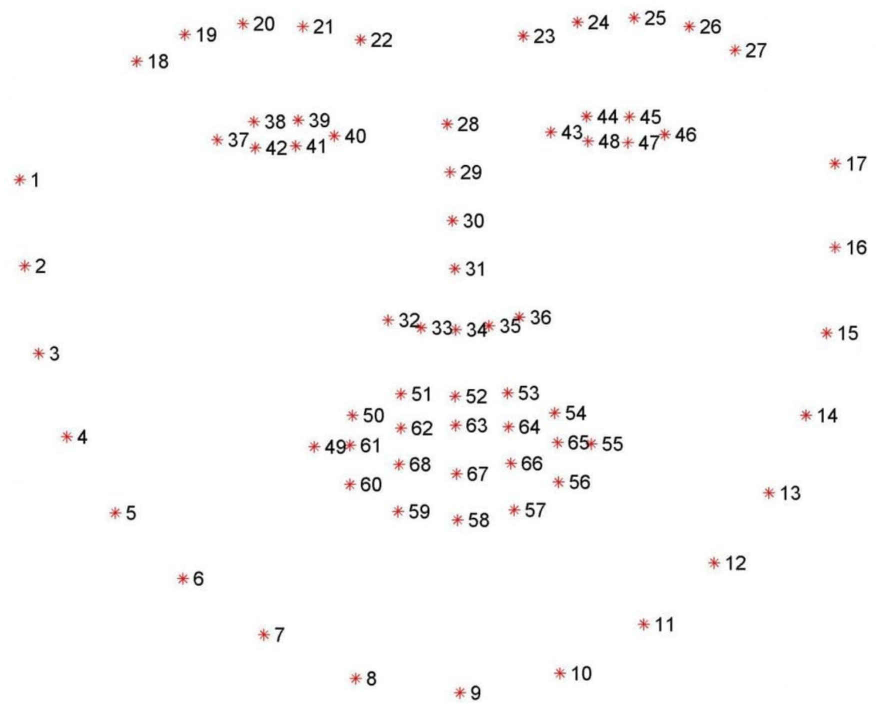
Based on the type of classifier

- Based on “hand-crafted” features (often called shallow machine learning models) - require careful engineering of the features to be extracted
- Based on deep networks (deep machine learning models) - require a large amount of data for training

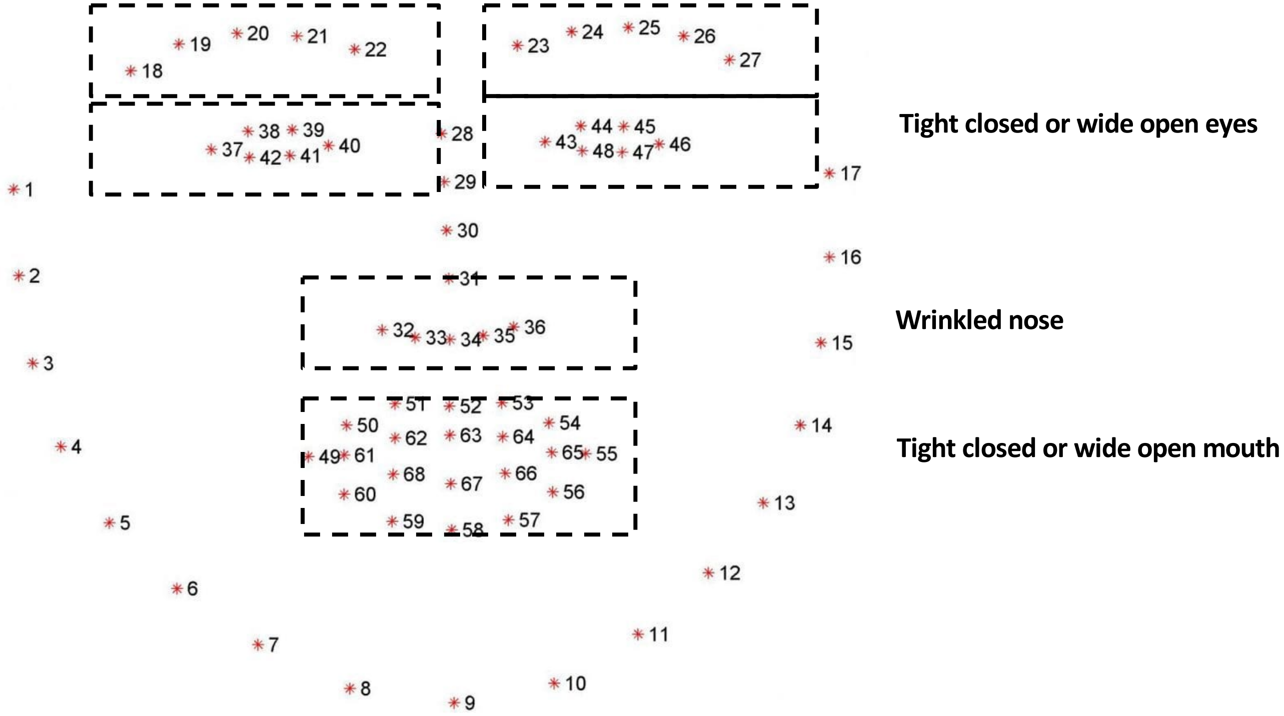
Based on the data processed

- Static approaches: single images are analyzed - work if the expression is particularly marked
- Dynamic approaches: the variations that occur between the neutral expression and the expression in its maximum manifestation are taken into account - require video data and greater computational complexity, as well as being influenced by the speed of the variations if the camera frame rate is low

Example of hand-crafted features: landmarks



Example of hand-crafted features: landmarks



Example of hand-crafted features: Action Units

- Action Units are the movements of a facial muscle or muscle groups that configure the expression of an emotion, based on the Facial Action Coding System (FACS) developed by Paul Ekman and Wally Friesen to develop the FACS system

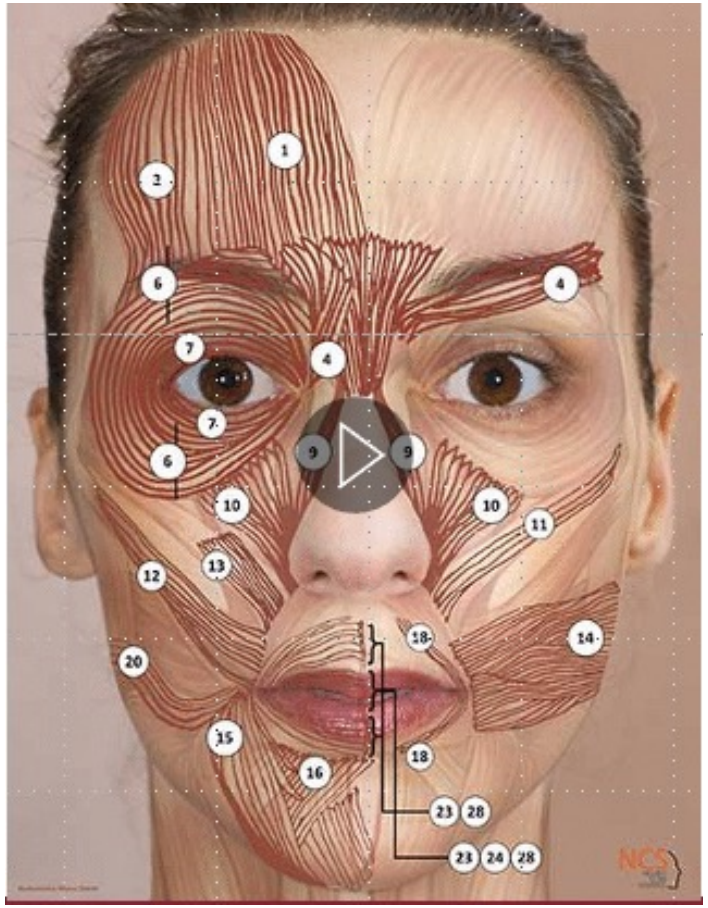
- Each AU is characterized by a number that indicates the type or muscle group involved, and a letter from A (minimum) to E (maximum) for the intensity.

From: https://lab-ncs.com/articoli/corso-facial-action-coding-system-facs/

Example of hand-crafted features: Action Units

- Action Units are the movements of a facial muscle or muscle groups that configure the expression of an emotion, based on the Facial Action Coding System (FACS) developed by Paul Ekman and Wally Friesen to develop the FACS system

- Each AU is characterized by a number that indicates the type or muscle group involved, and a letter from A (minimum) to E (maximum) for the intensity.



From: https://lab-ncs.com/articoli/corso-facial-action-coding-system-facs/

Example of shallow machine learning approach

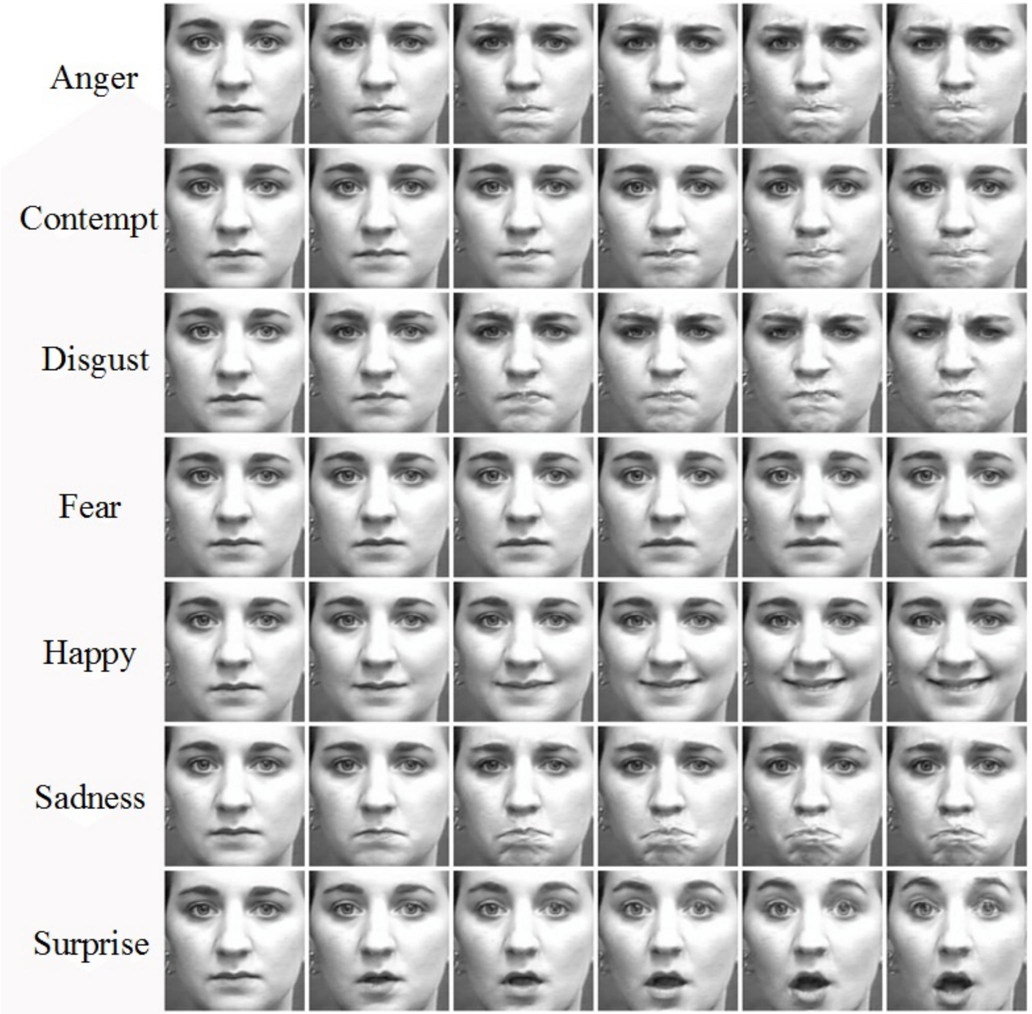
Single images: example dataset Feret 13
(https://www.kaggle.com/datasets/msambare/fer2013)

Only the peak of the expression is considered

The classification is affected by the emotion arousal

Example of shallow machine learning approach

Video (frame sequence): example dataset Extended Cohn-Kanade dataset (CK+) (https://www.kaggle.com/datasets/davilsena/ckdataset)



The overall expression dynamics is considered starting from the initial state, which is generally neutral.

Also in this case arousal affects classification accuracy

Proposed pre-processing

- After isolating the first 20% of the frame sequence (stage 0), which has been empirically observed to represent the neutral expression, the remaining frames are classified into:
- 0–33% of the expression (stage 1)
- 33–66% of the expression (stage 2)
- 66–100% of the expression (stage 3)

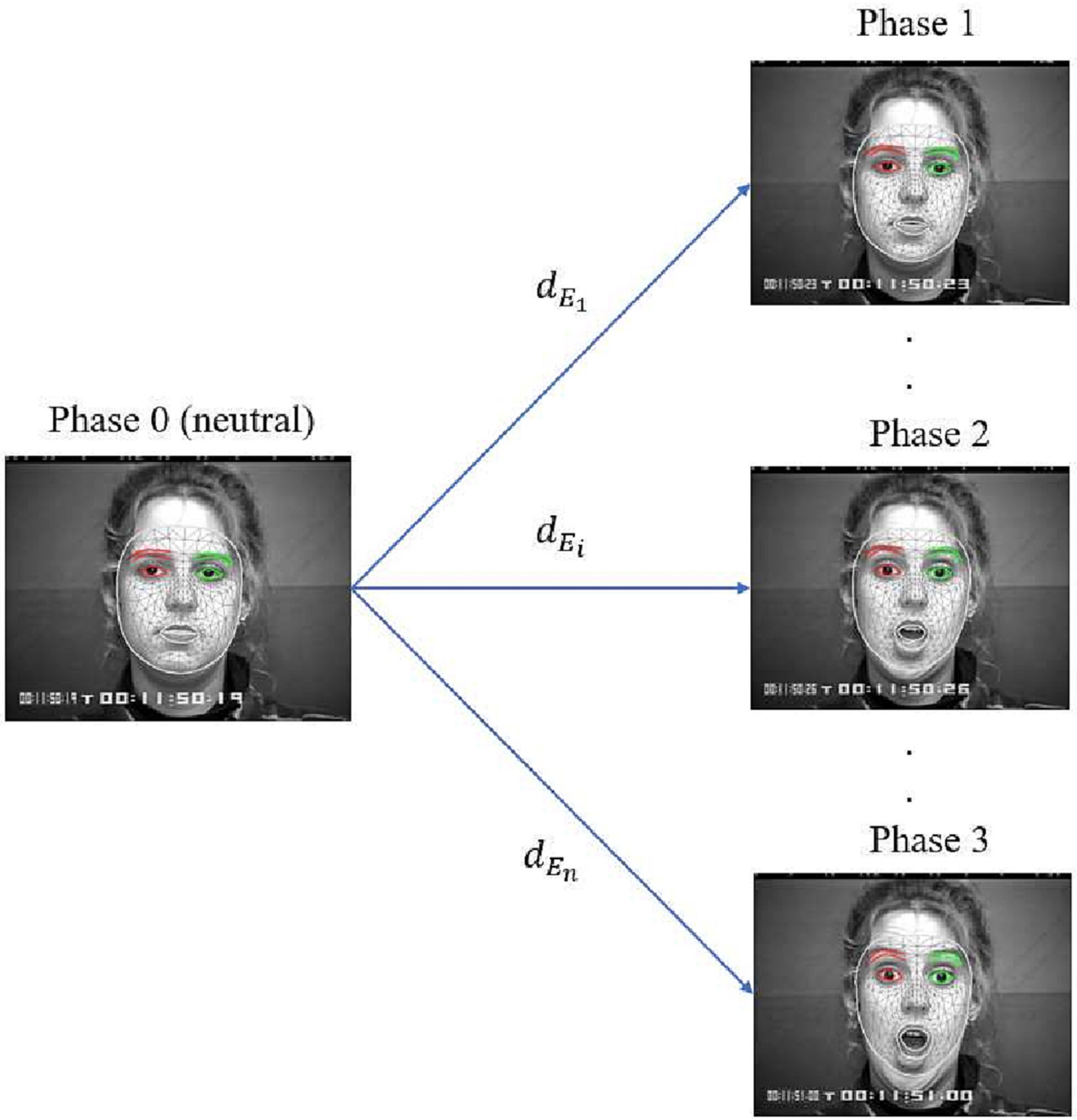
CK+ contains videos representing anger, contempt, disgust, fear, happiness, sadness and surprise

With preprocessing we go from 7 to 22 classes: a single neutral class that shows common features for all expressions, and a class for each of the 3 stages of each expression

Chosen features

- MediaPipe Face Mesh extracts 468 3D landmarks for the face (real x,y, virtual z)
- For each frame/image of the sequence related to a subject starting from the first (neutral expression) the extracted points are organized in a matrix
- The adopted method then calculates the Euclidean distances between the coordinates of the landmark point in the first matrix and those in each of the following matrices calculated from the second to the last frame of the same sequence
- Each image in the video sequence is compared with the neutral expression at the starting point of the sequence using the displacement of each of the detected landmarks with respect to the initial position.
- Each of these comparisons produces a series of 468 distances, one for each landmark point.
- The 468-element vectors are used to train a classifier for each of the 22 classes

Chosen features



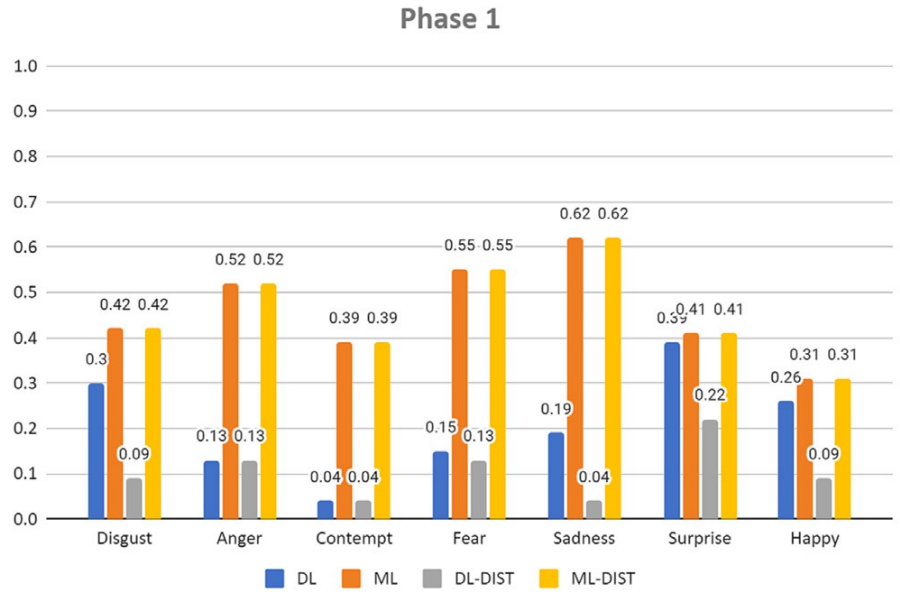
Chosen classifier

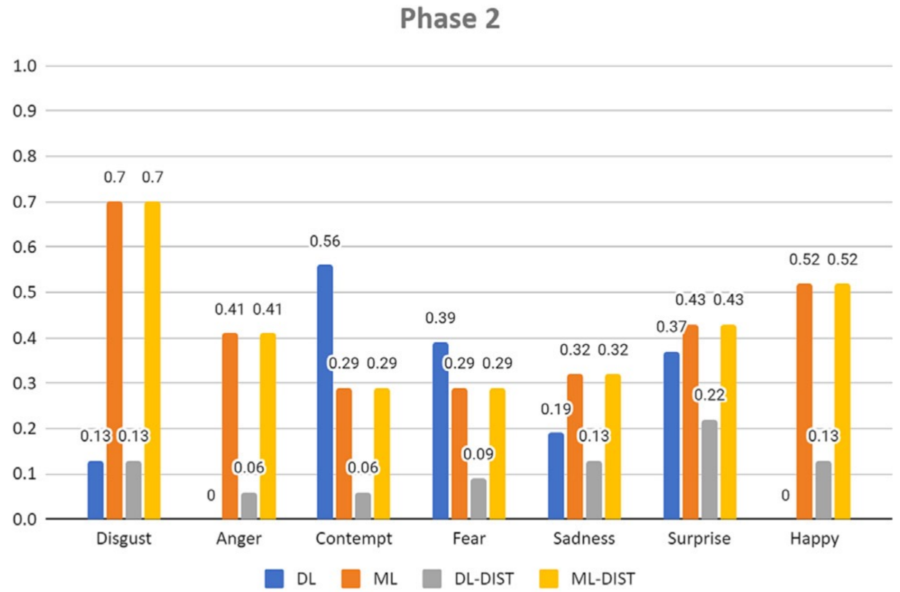
- We chose to use the SVM (Support Vector Machine) classifier as a basis
- The multiclass problem is approached as a series of binary problems (One-to-One strategy) The total number of classifiers built is n(n-1)/2 where n is the number of classes. In our case, we created 231 SVMs.

Experiments

- The method was compared with a concatenation of VGG-Face, (a popular CNN specialized for faces) to extract the embeddings to be used as features and LSTM (Long Short Term Memory) a recurrent neural network able to take into account the dynamics present in the input sequence for classification.
- Reduced size images were used to test the possibility of remote processing

Results

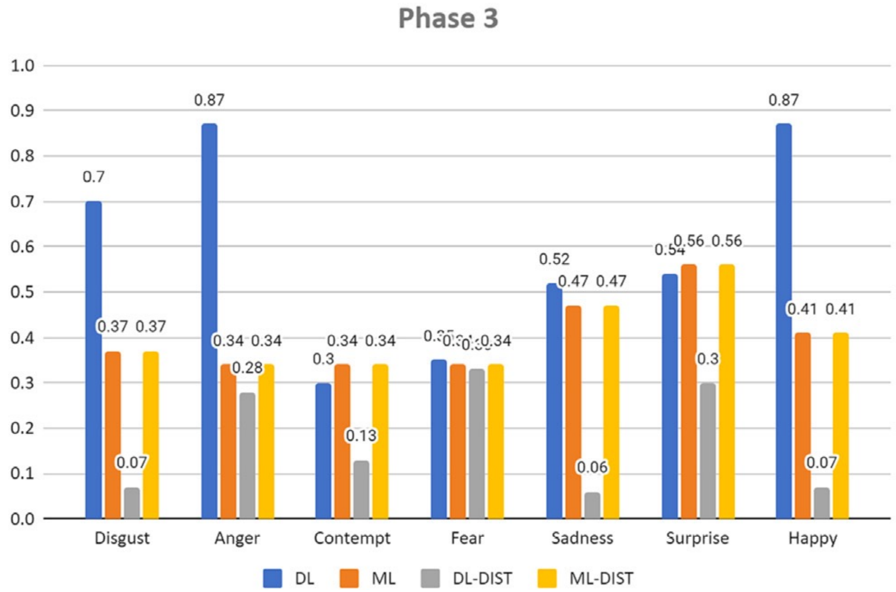




Shallow ML better in first phases

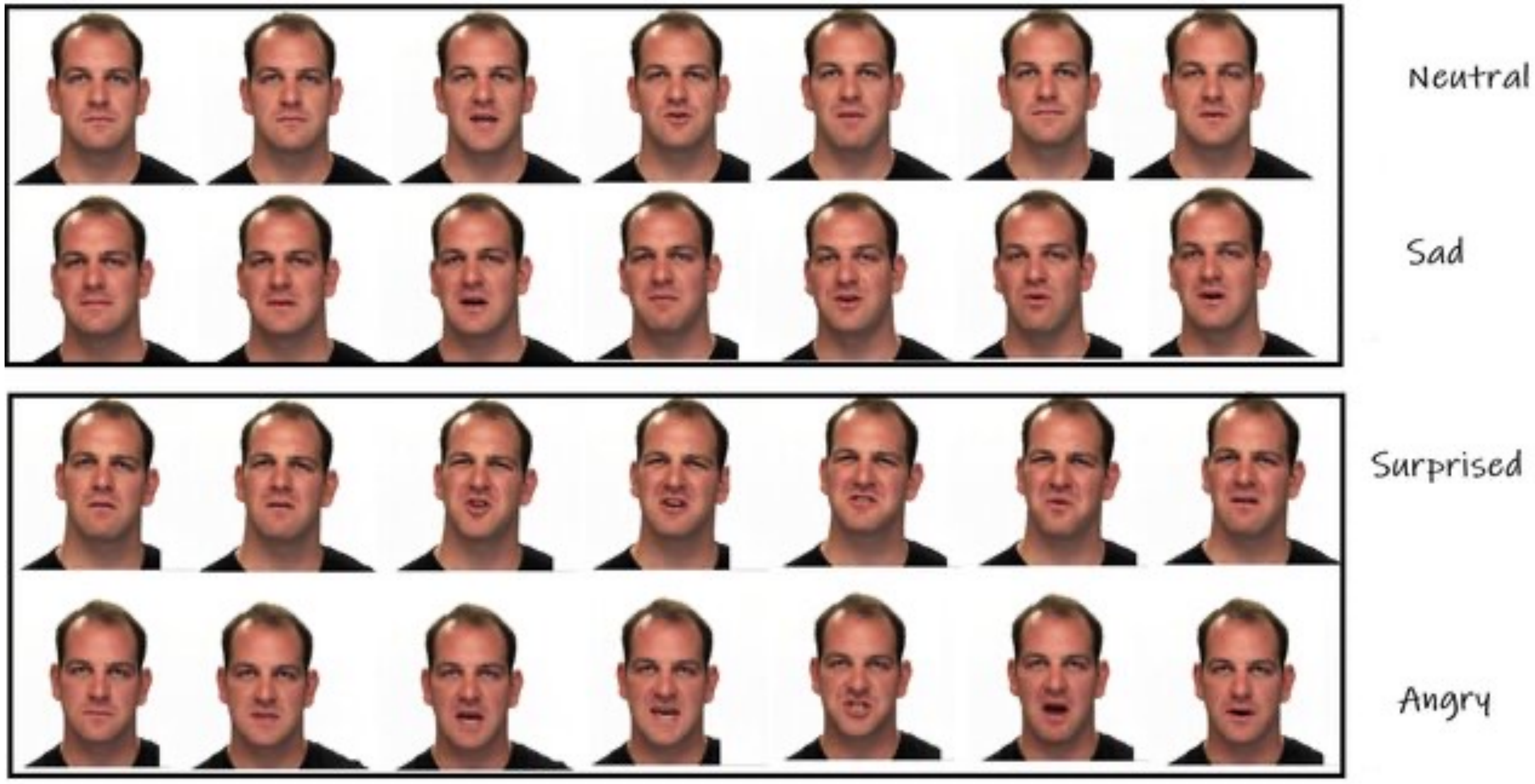
Shallow ML better at a distance

TODO: fuse phases



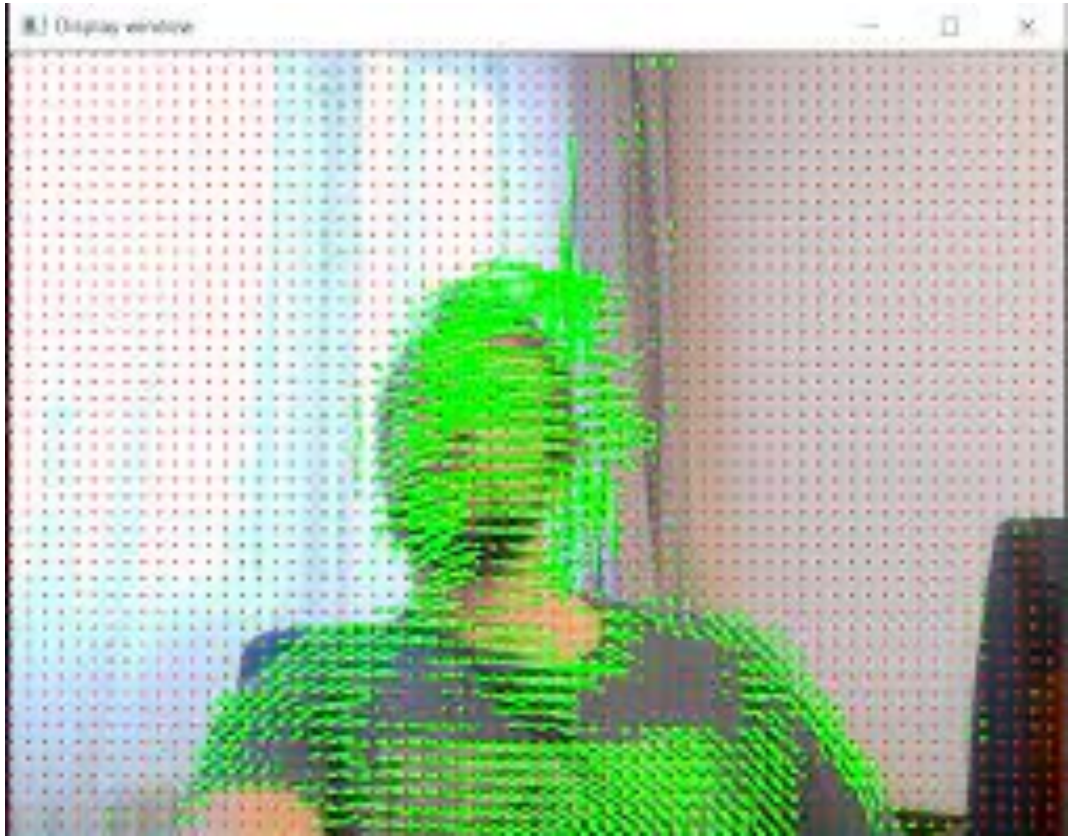
A deep learning approach: Multimodal Emotion Recognition with face, movement and voice

Face: video frames



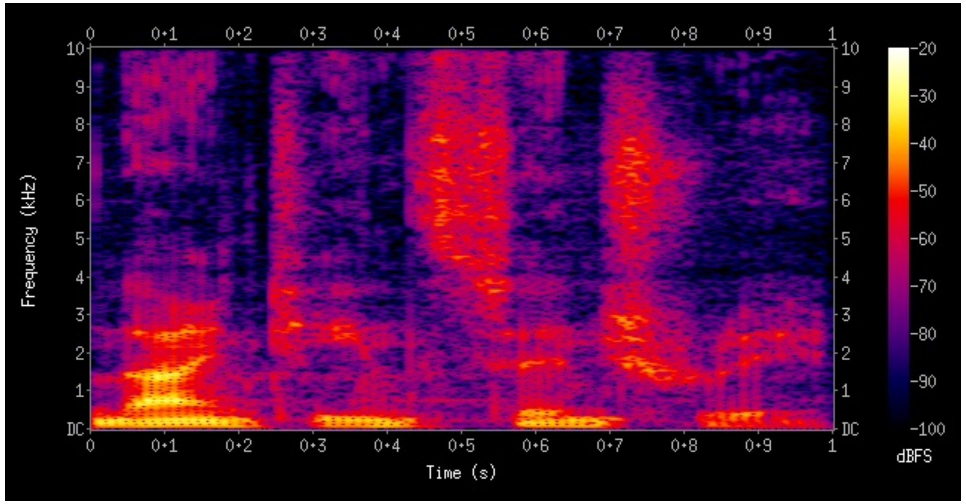
A deep learning approach: Multimodal Emotion Recognition with face, movement and voice

Face: Optical Flow to capture movement from pairs of subsequent frames



A deep learning approach: Multimodal Emotion Recognition with face, movement and voice

Voice: spectrograms



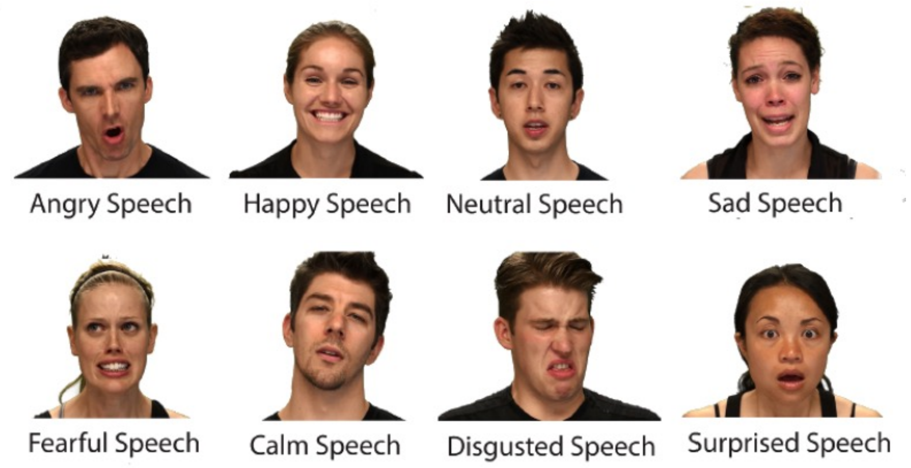
On a Cartesian diagram, frequency (f) is represented on the vertical axis and is measured in Hertz (Hz), or cycles per second, and in kilohertz (kHz), or thousands of cycles per second, while the time dimension (t) is represented on the horizontal axis. The graph can be created in different formats; generally, the intensity of the sound is expressed by the color.

A spectrogram is usually obtained by dividing the total time interval (i.e. the one relating to the entire waveform to be analyzed) into equal sub-intervals (time windows) of 5 to 10 ms duration and calculating the Fourier transform (or the fast Fourier transform, FFT) of the part of the waveform contained in each window, which provides the intensity of the sound as a function of the frequency. The Fourier transforms, relating to the different time windows, are then assembled to form the spectrogram

A deep learning approach: Multimodal Emotion Recognition

Multimodal Emotion Recognition via Convolutional Neural Networks: comparison of different strategies on two multimodal datasets

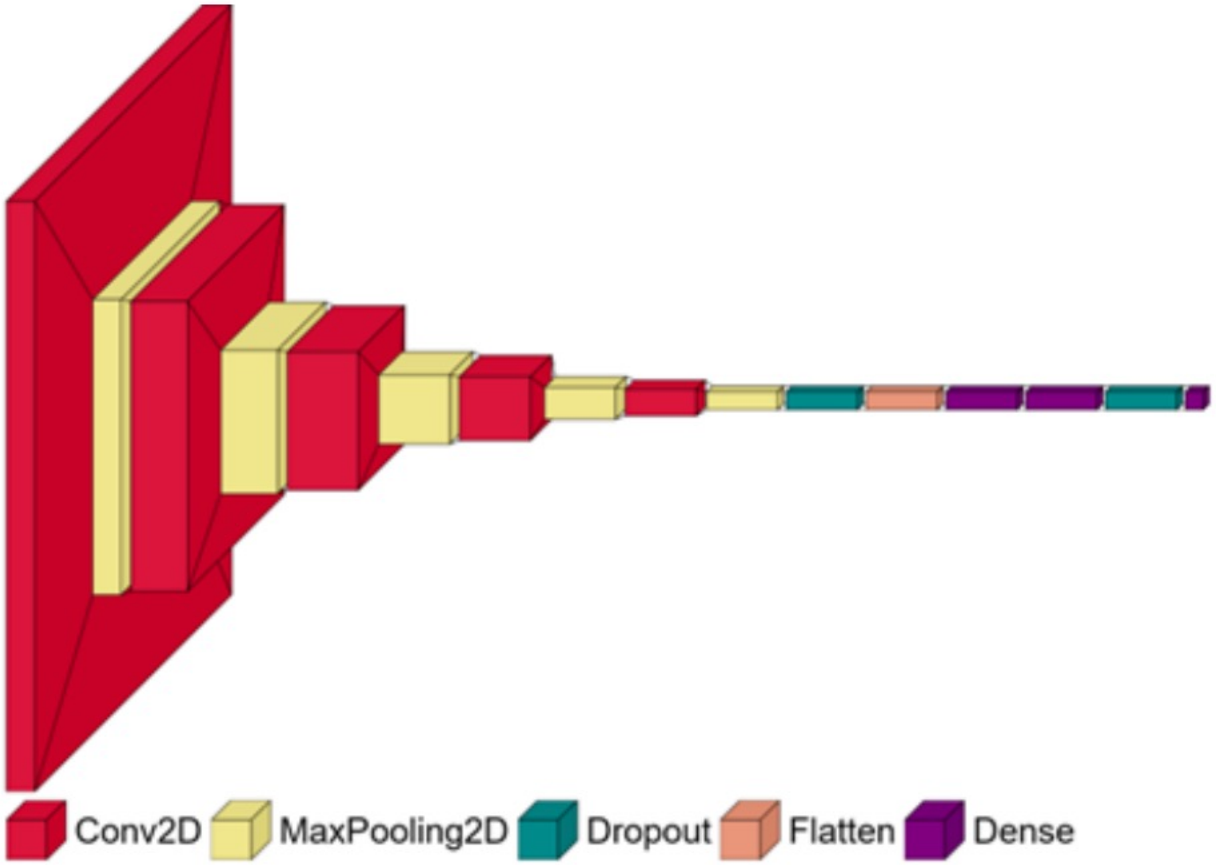
Graduation work by Sara Tramonte



Samples from RAVDESS dataset

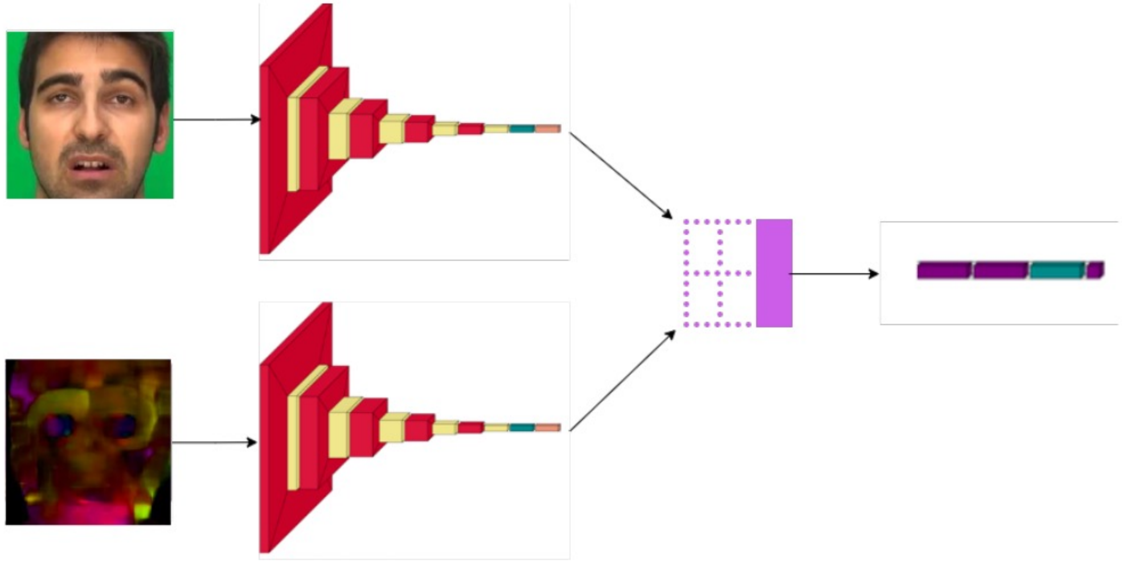
Subjects participating in BAUM-1 dataset

Multimodal Emotion Recognition

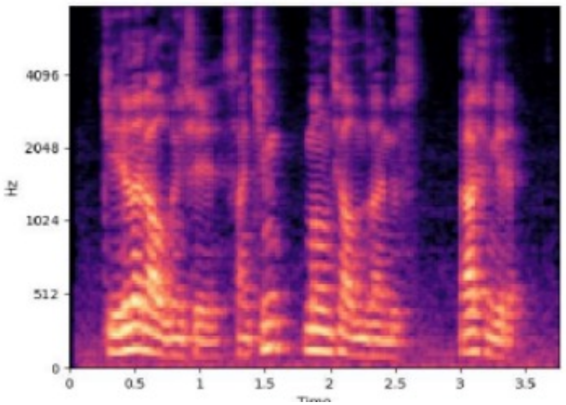


Proposed model for emotion recognition by frames and/or MEL-spectograms

Multimodal Emotion Recognition



Our proposed 2-input CNN model



Multimodal Emotion Recognition

Model type \Epochs	**10**	**20**	**30**	**40**
F & Opt	0.6187	0.8224	0.8412	0.8477
Opt & Mel	0.7558	0.8564	0.8803	0.8667
F & Mel	**0.7558**	**0.9416**	**0.9667**	**0.95**
3-input	0.8374	0.9212	0.9264	0.9392

The comparison of results with different 2-input models and a 3-input model on BAUM-1.

Type of input \Epochs	**10**	**20**	**30**	**40**
Frames	**0.8922**	**0.91**	**0.926**	**9377**
Opt	0.6642	0.6739	0.6851	0.6953
Mel	0.6642	0.8375	0.833	0.8623
2-input : F & Mel	0.7914	0.8378	0.8468	0.9014
3-input	0.7955	0.8934	0.9146	0,9127

The comparison of results with different 2-input models and a 3-input model on RAVDESS - normal data.

Type of input \Epochs	**10**	**20**	**30**	**40**
F	0.933	0.9285	0.9316	0.9529
Opt	0.7038	0.7064	0.7021	0.7011
Mel	0.7423	0.9086	0.9306	0.9439
2-input : F & Mel	**0.9205**	**0.9344**	**0.952**	**0.9595**
3-input	0.9	0.9289	09366	0.9359

The comparison of results with different 2-input models and a 3-input model on RAVDESS - strong data.

Multimodal Emotion Recognition

Methods	**Year**	**Accuracy**
Zhalehpour et al. [19]	2017	51.29
Zhang et al. [34]	2018	54.57
J.Cornejo and H. Pedrini [42]	2019	56.01
F. Ma et al. [43]	2020	67.59
C. Guanghui and Z. Xiaoping [44]	2021	71.26
Kansizoglu et al. [45]	2022	56.01
I. Hina et al. [46]	2022	61.68
Our work	**2023**	**95.00**

Comparison on BAUM-1

Methods	**Year**	**Accuracy**
Ghaleb et al. [47]	2019	79.00
Su et al. [48]	2020	74.86
C. Luna-Jiménez et al. [23]	2021	80.08
A. Radoi et al. [26]	2021	78.70
B. Mocanu & R. Tapu [49]	2022	87.89
Middya et al. [28]	2022	86.00
Our work	**2023**	**95.95**

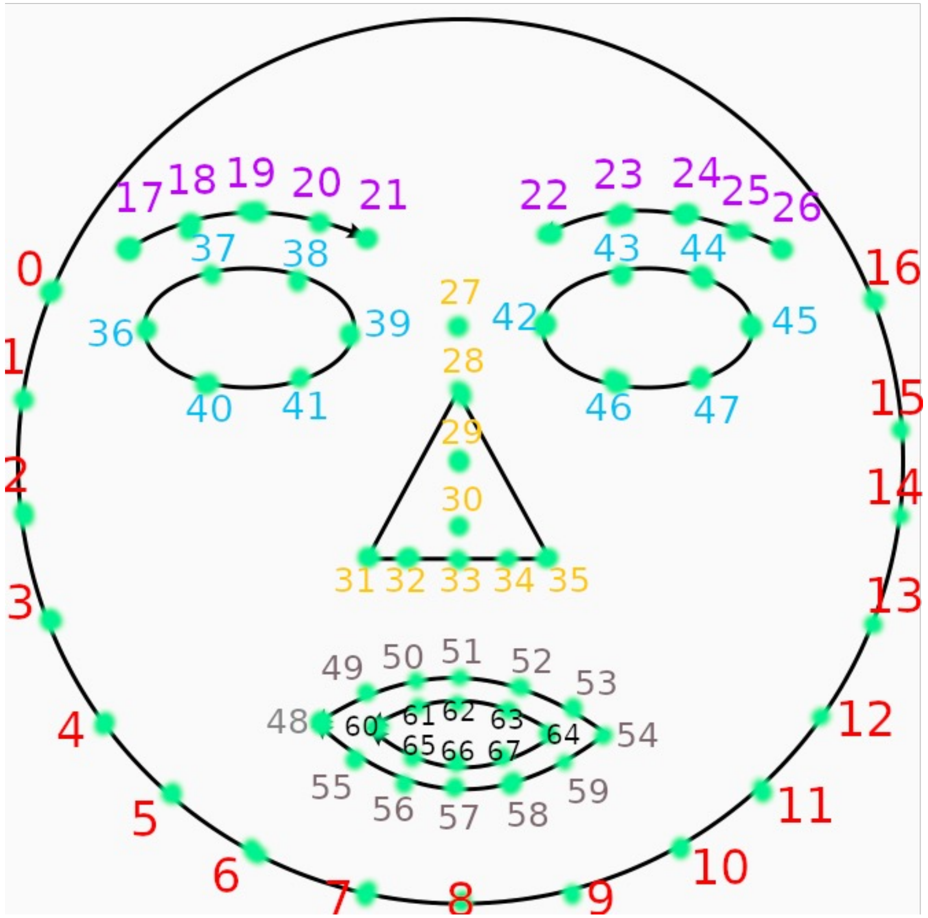
Comparison on RAVDESS

Lie Detection

Your face may say the truth when you lie

Graduation work by Giordano Dionisi
Graduation work by Donato Francesco Pio Stanco

How To Spot Features?
LANDMARKS

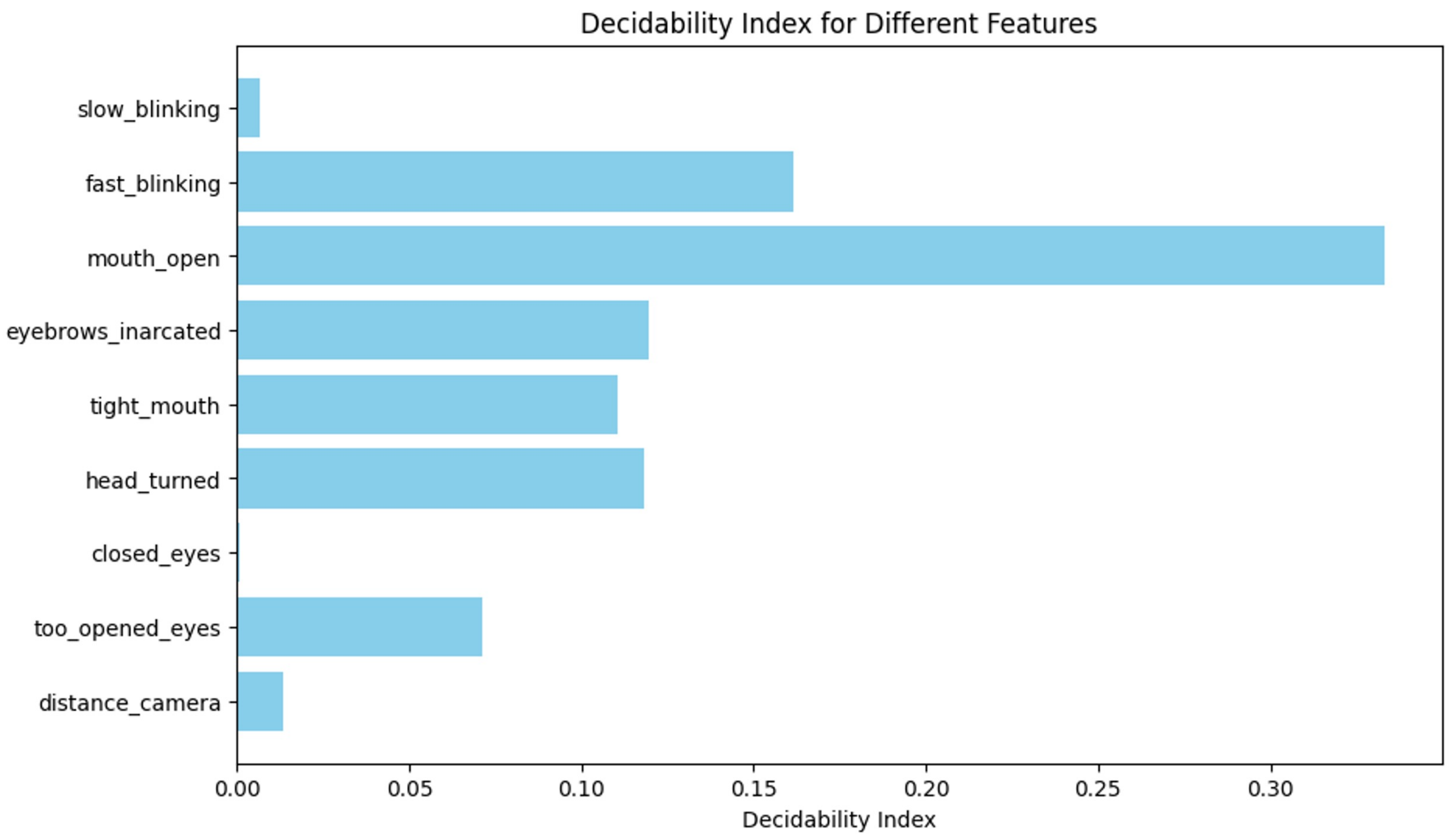


Chosen microexpressions

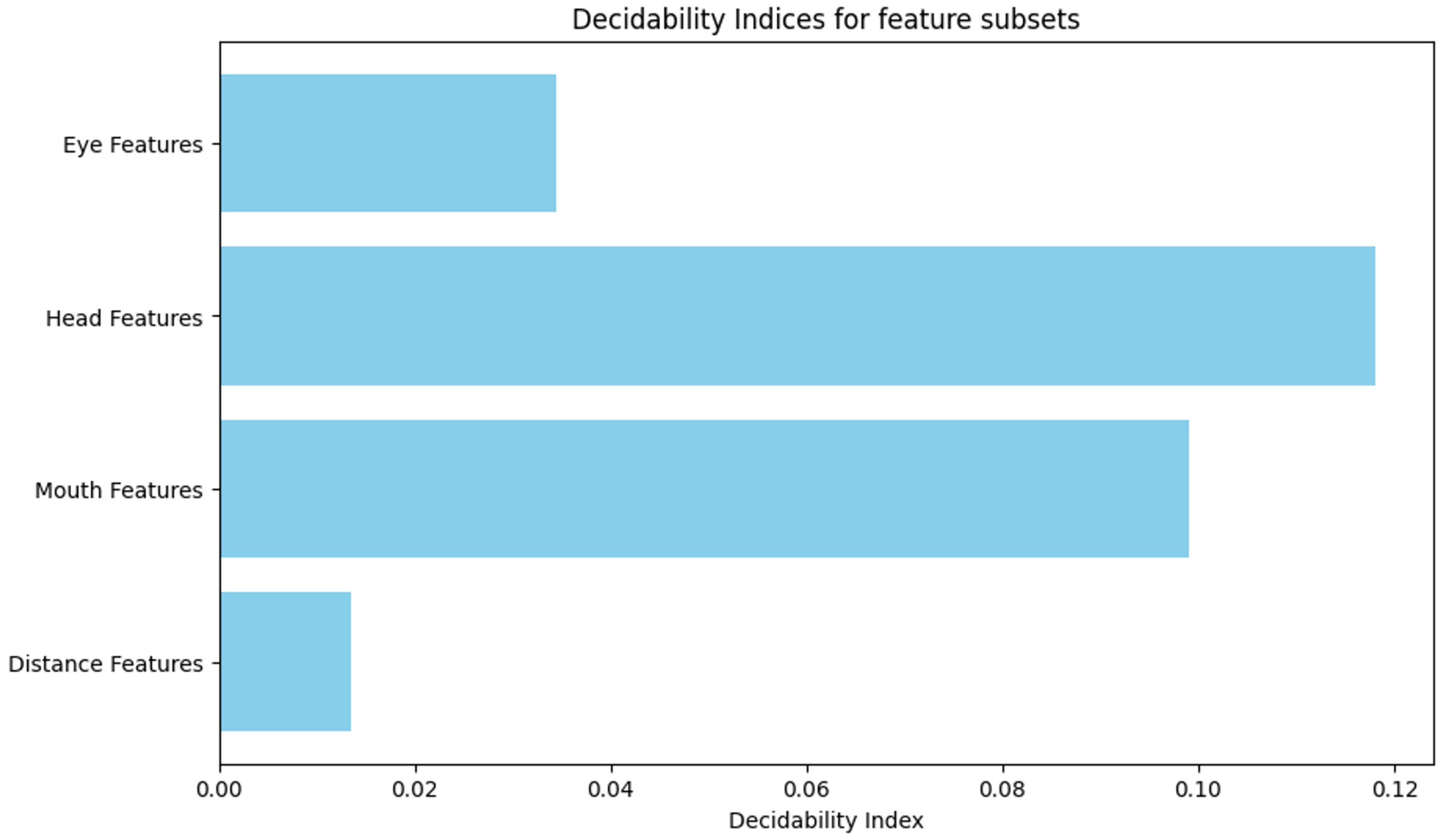
For facial detection, γ's facial landmarks are captured and the following features are computed:

- **Eye Blinking**, that is subdivided into:
  - Fast Blinking, Slow Blinking, Closed Eyes, Too Opened Eyes
- **Mouth Dynamics** divided into:
  - Mouth Open, Tight Mouth, Raised Eyebrows
- **Head Position**:
  - Left-Right Movement, Bottom Left-Right Movement, Turned Head
- **Distance From Camera**

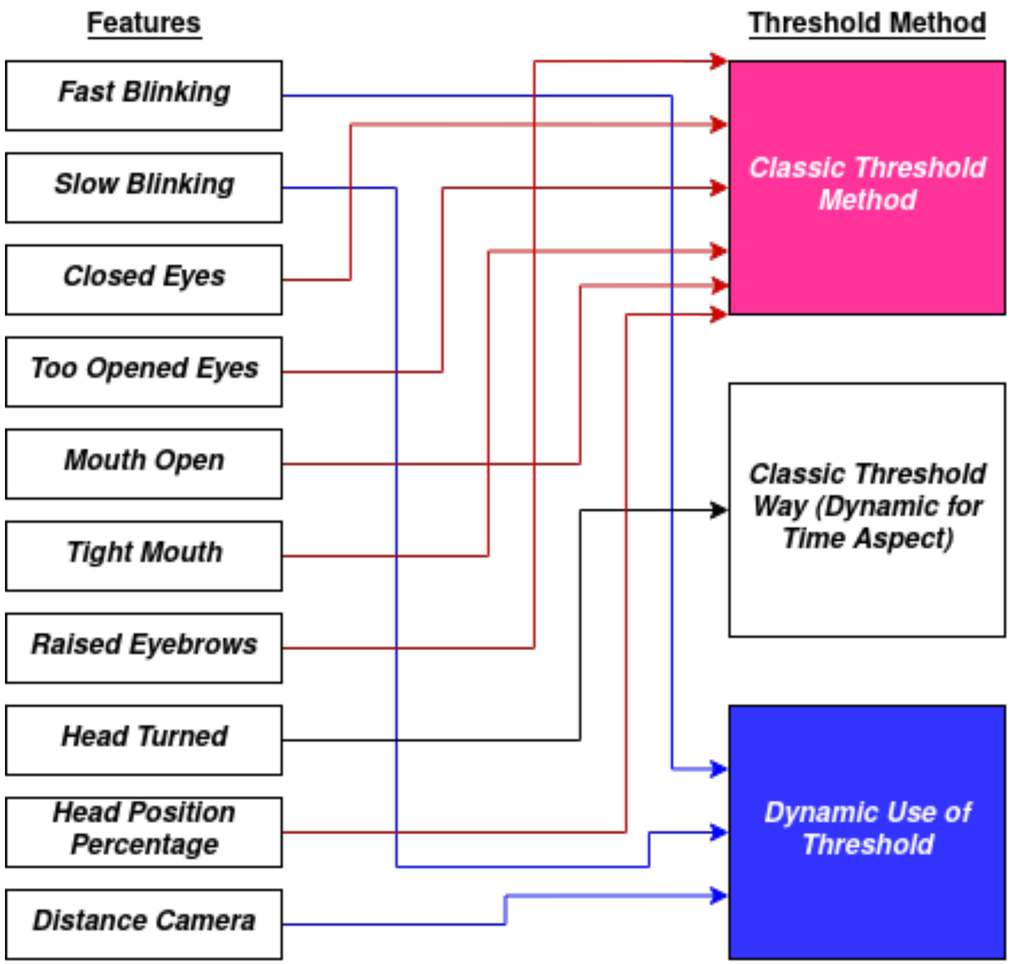
Decidability of chosen features



Decidability of subsets of features



See Displacement …



Dynamic thresholds

To manage critical thresholds we decided to use the **Exponential Moving Average** or **EMA**.

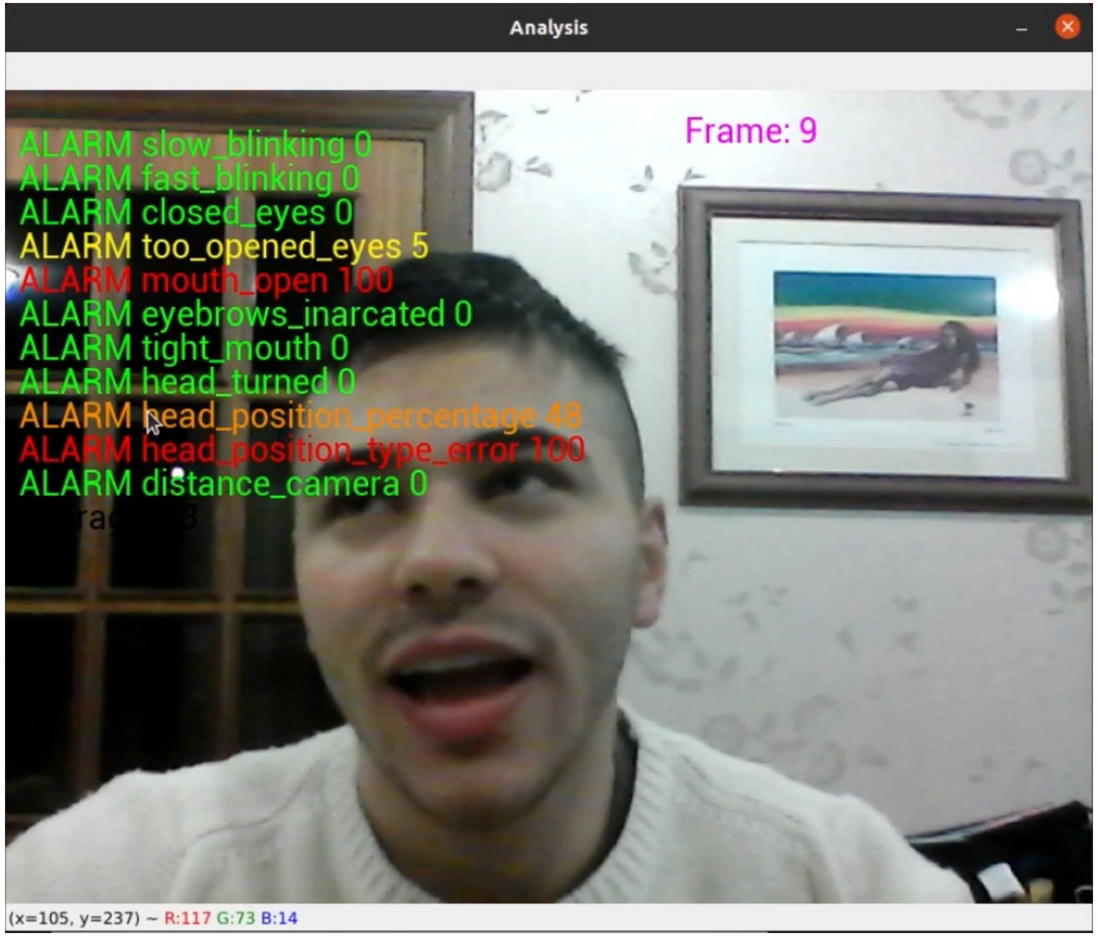
EMA is used in the following way:

$$\text{DynamicThreshold} = (\alpha * (C - P)) + P$$

where:

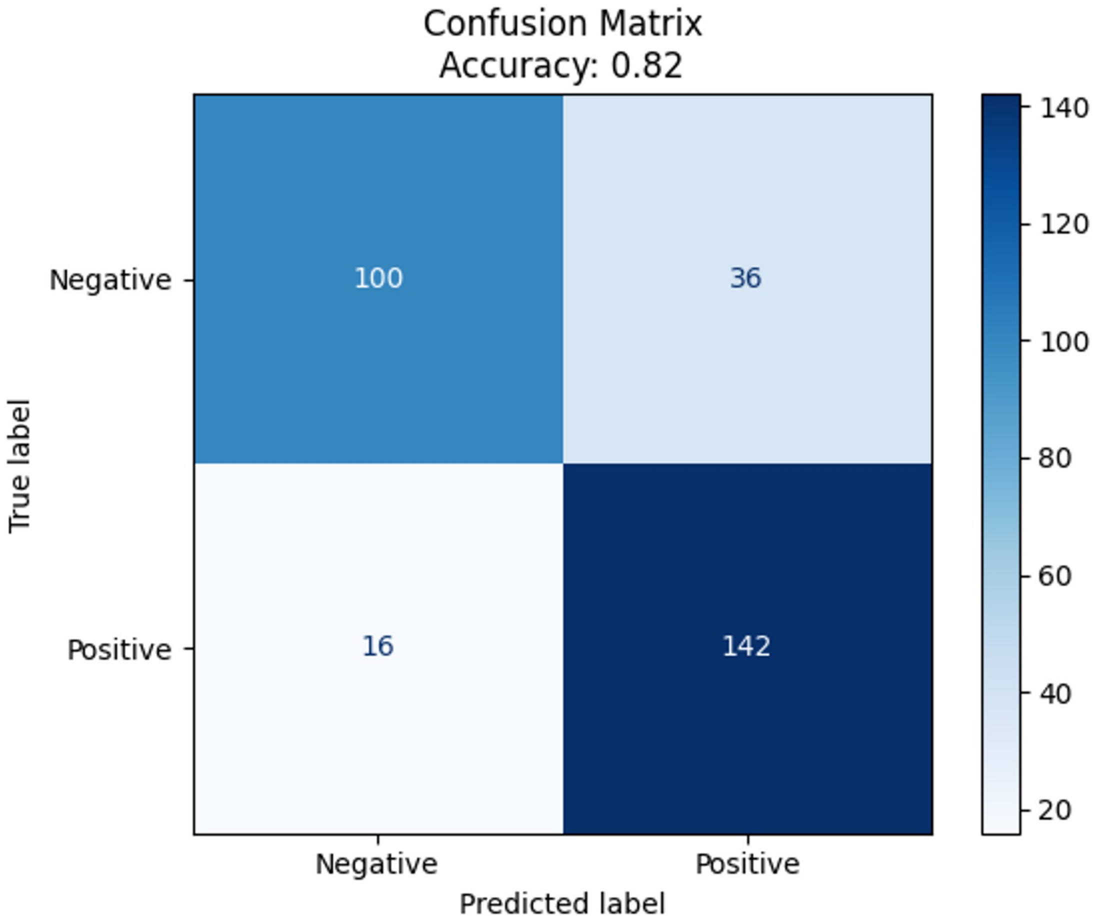
- α: represents the smoothing factor
- C: represents the current value of the feature
- P: represents the previous value of the threshold at that time

Real-Time Evaluation



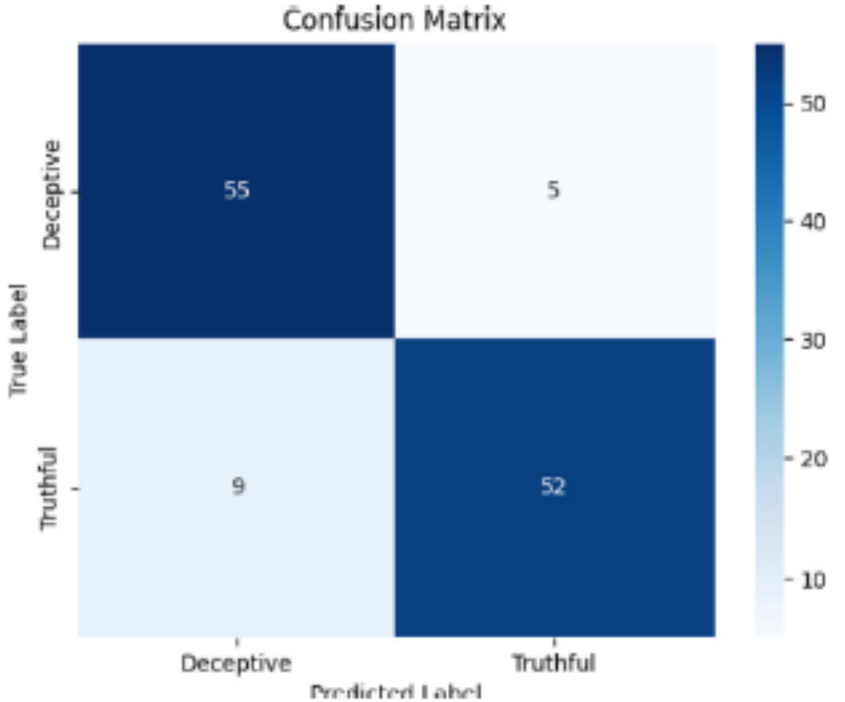
Offline evaluation on two datasets

Bag-of-Lies



Offline evaluation on two datasets

Real-life trial dataset



Comparison with SOTA

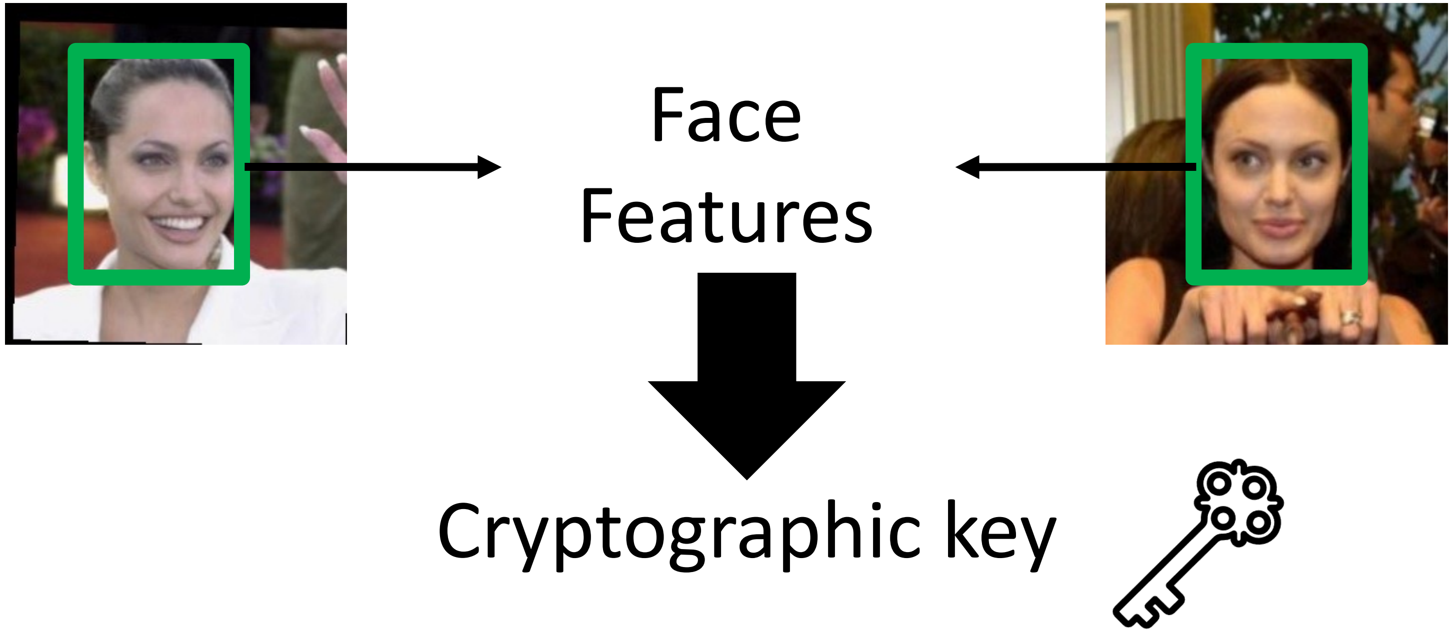
Comparison with other methods in the state of the art working on micro-expressions. When tested on multiple modalities, the comparison is with the visual one. For completeness, also methods using the entire video frames are reported. The best results for deception detection from micro-expressions are in bold, and the second-best results are underlined.

PAPER	Dataset	Accuracy for single trait	AUC for single trait
Pérez-Rosas et al. (2015)	Real-Life Trial Dataset	76.03 (RF) visual data	–
Wu et al. (2018)	Real-Life Trial Dataset	–	0.8064 (RF)
Krishnamurthy et al. (2018)	Real-Life Trial Dataset	76.19 (3D-CNN) visual data	0.7512
Avola et al. (2019)	Real-Life Trial Dataset	76.84 (SVM-RBF)	–
Gupta et al. (2019)	Bag-of-Lies	55.26 (LBP+RF) visual data	–
Kamati et al. (2021)	Real-Life Trial Dataset	96.00 (Deep architecture) visual data	–
Kamati et al. (2021)	Bag-of-Lies	97.35 (Deep architecture) visual data	–
Proposed FTM	Bag-of-Lies	<u>82.00</u>	–
Proposed FTM	Real-Life Trial Dataset	<u>88.00</u>	0.96

Using face templates as cryptographic keys

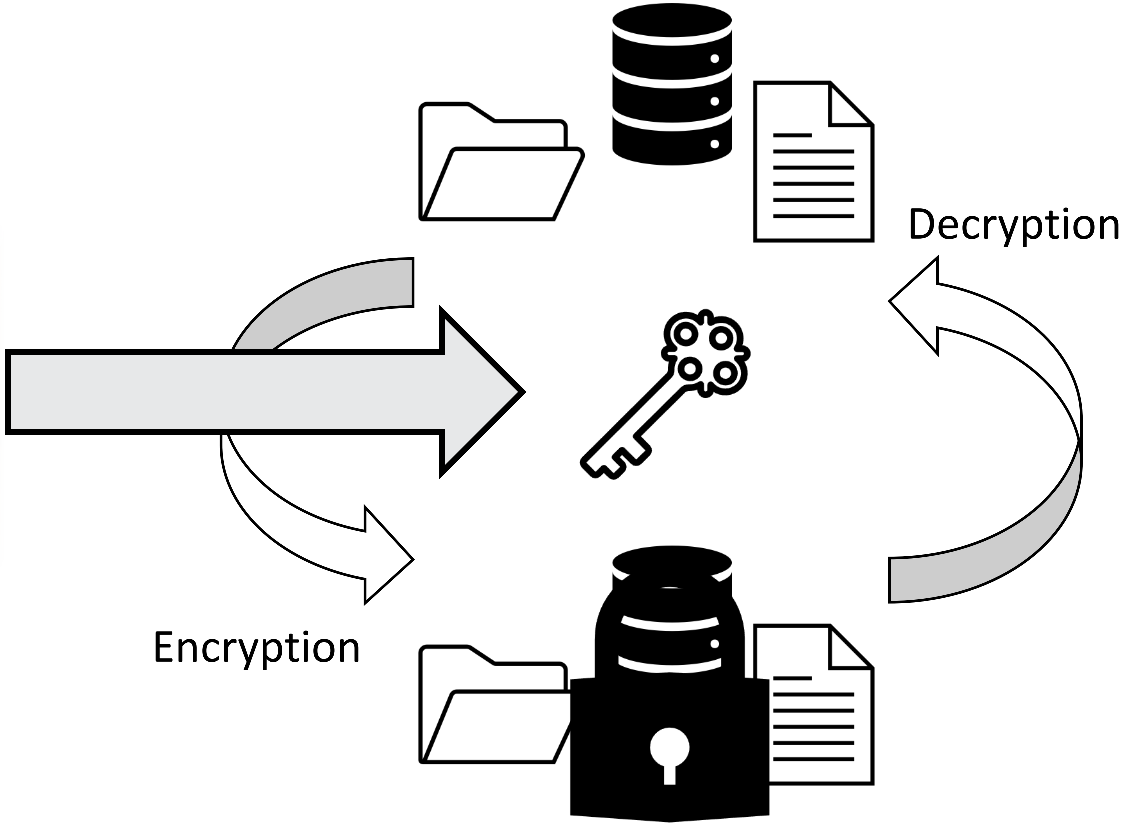
- **Graduation work by Aldo Cascone**

  - The work focused on analyzing the process of features extraction from face templates, exploring techniques and models in the literature to reduce *intra-class variability* and improve the *reproducibility* of extracted data.



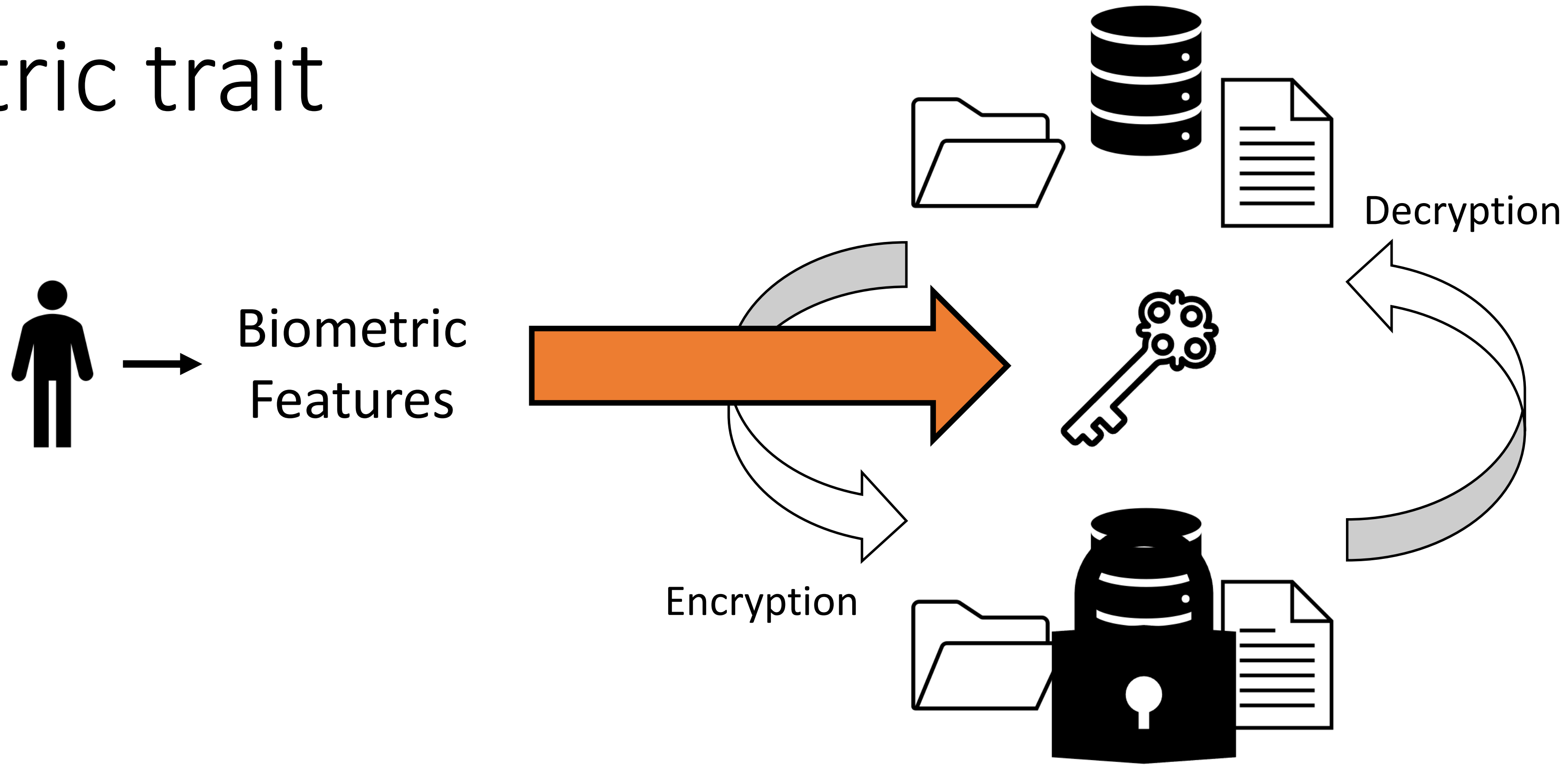
Traditional

Passwords
(or passphrases)



Risks:
- brute force attack
- theft of the combination

Biometric trait



Pros →

- Biometric data cannot be lost, loaned, stolen, or forgotten
- Uniqueness

Cons →

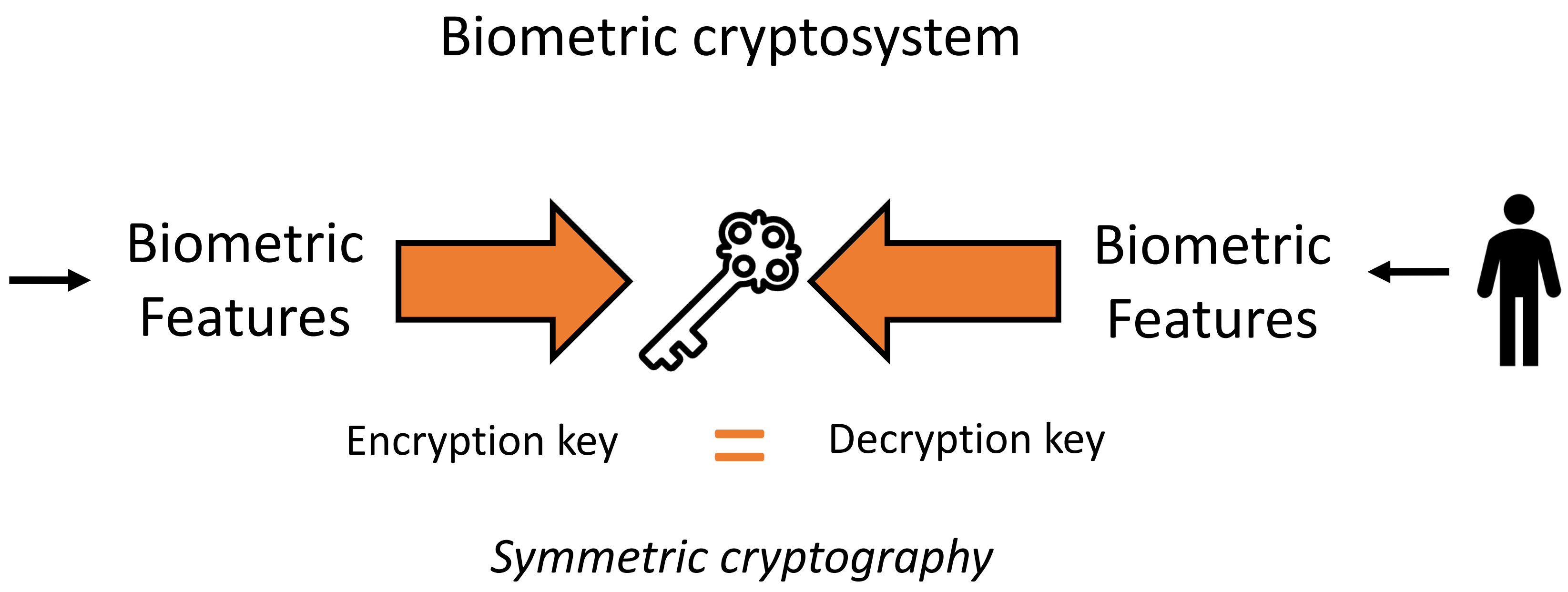
- Intra-class variations

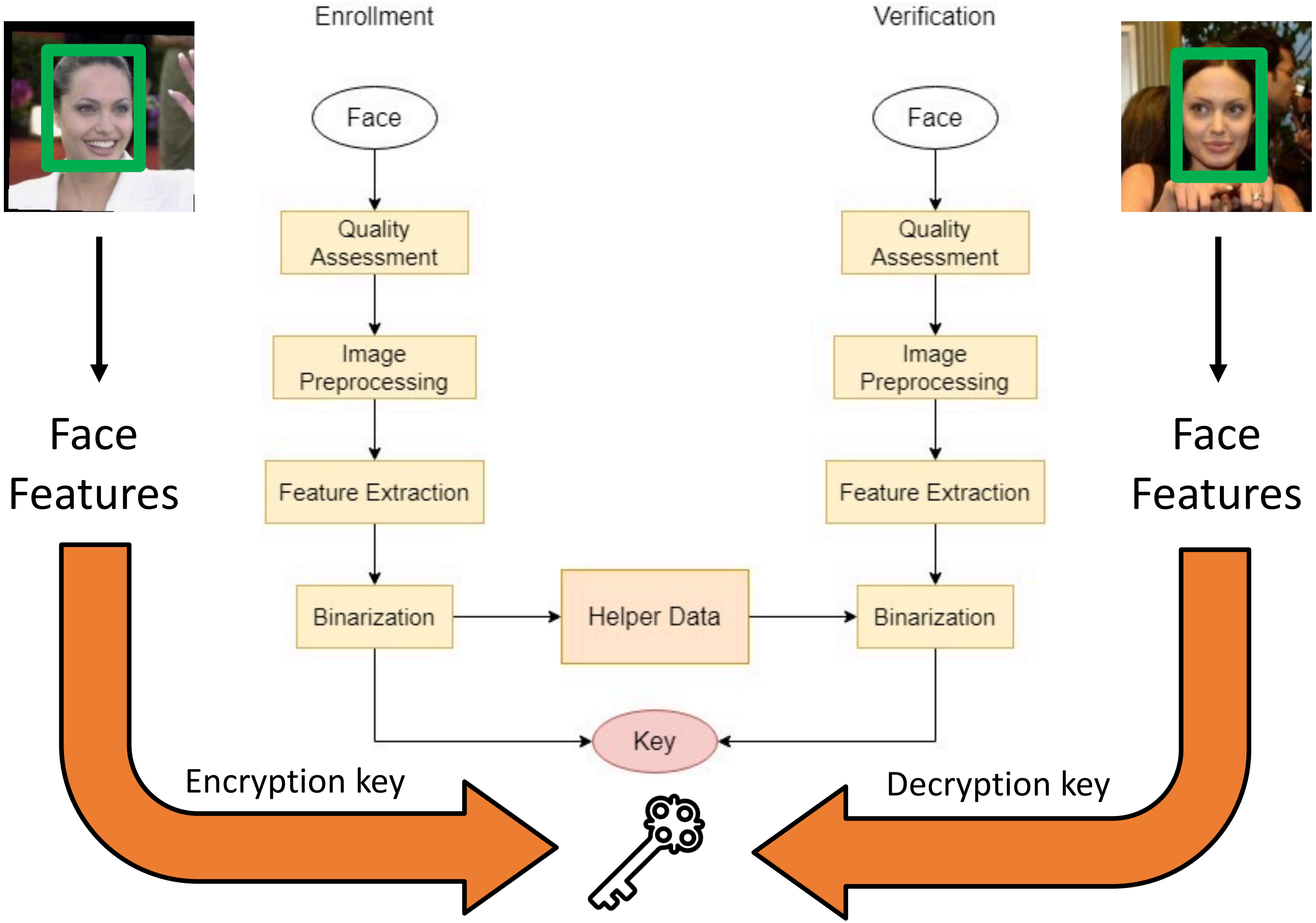
Problem

Intra-class variations

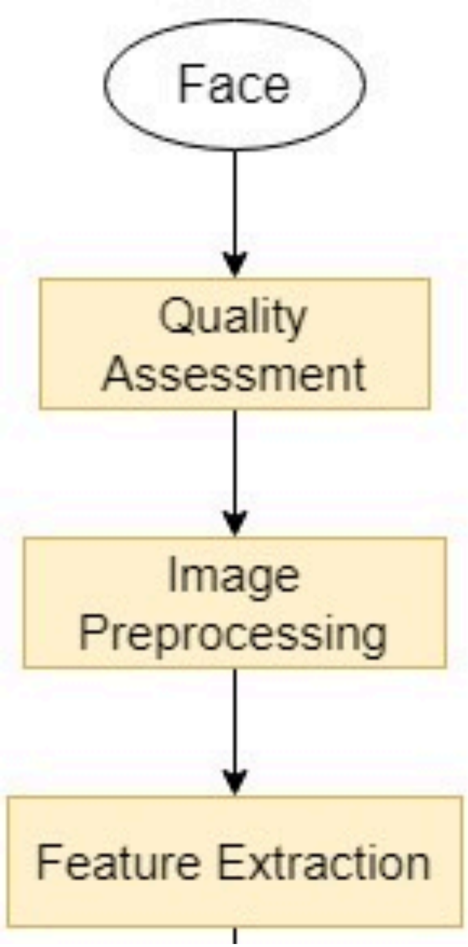
- Environmental conditions
- Capturing sensors
- User's cooperation in positioning themselves correctly.

Approach


Biometric cryptosystem
Biometric Features
Biometric Features
Encryption key
Decryption key
Symmetric cryptography

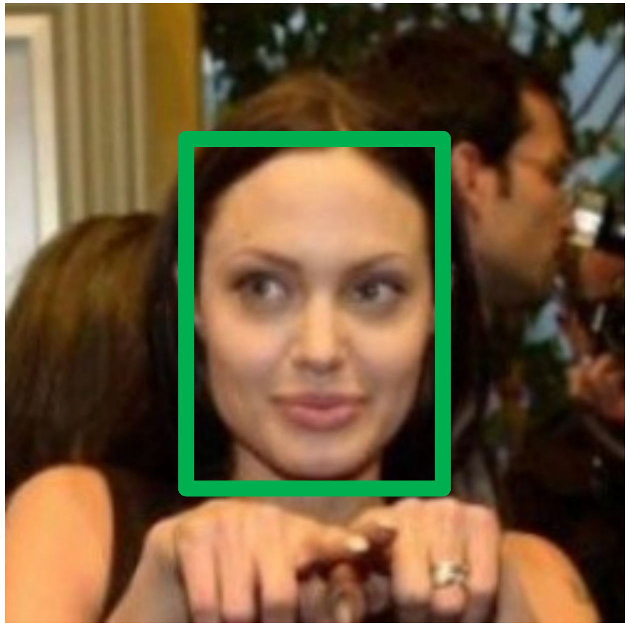

Enrollment
Verification
Face
Quality Assessment
Image Preprocessing
Feature Extraction
Binarization
Helper Data
Key
Face
Quality Assessment
Image Preprocessing
Feature Extraction
Binarization
Face Features
Face Features
Encryption key
Decryption key

Workflow

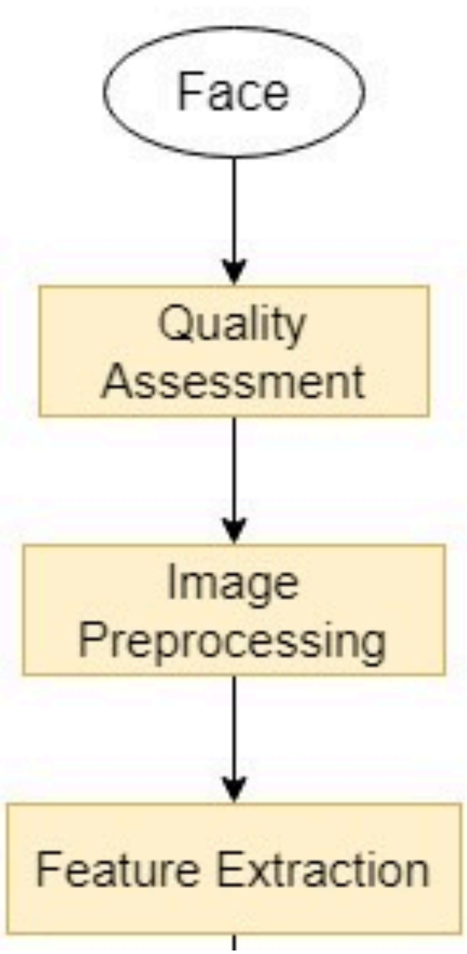


Face Detection →

Given an input image, the process aims to identify all faces within the image.

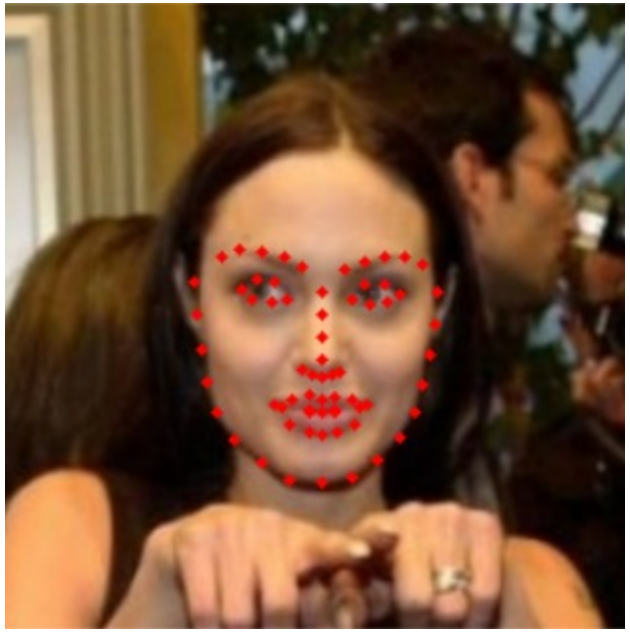


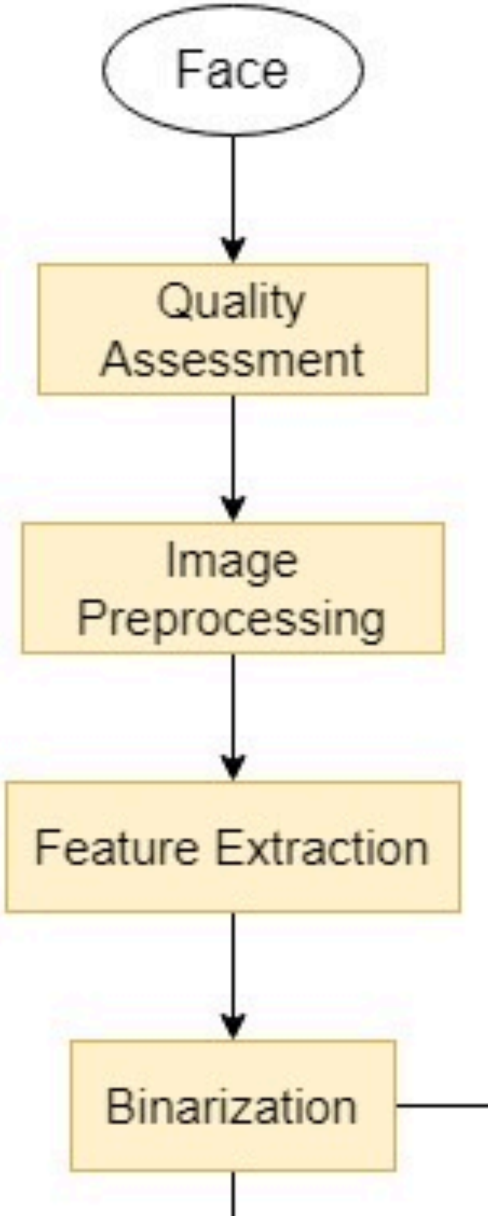
Workflow



Facial Landmark Detection →

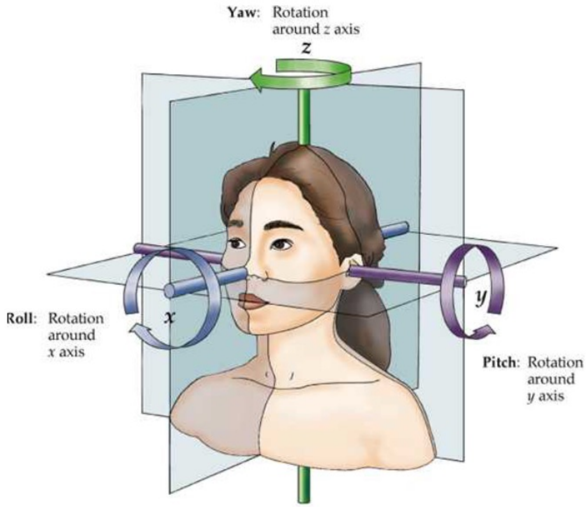
The process locates the facial landmarks, salient regions of the face, including eyes, eyebrows, nose, mouth, and jawline.





Quality Assessment →

Quality Assessment in terms of pose and illumination distortions.



Pose distortions

Illumination distortions

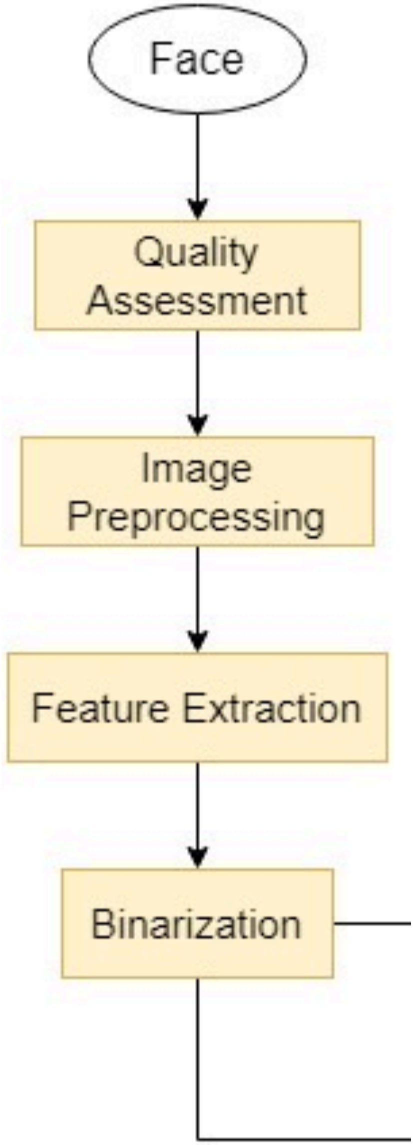
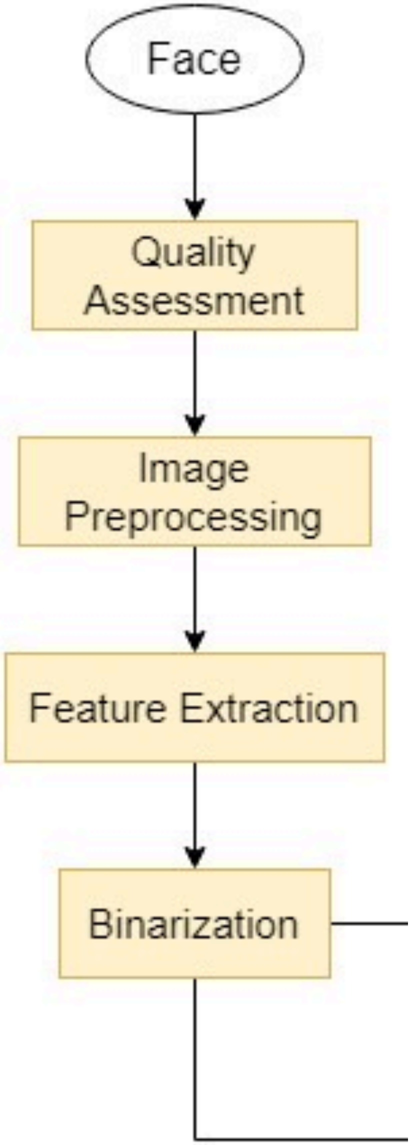


Image Preprocessing →

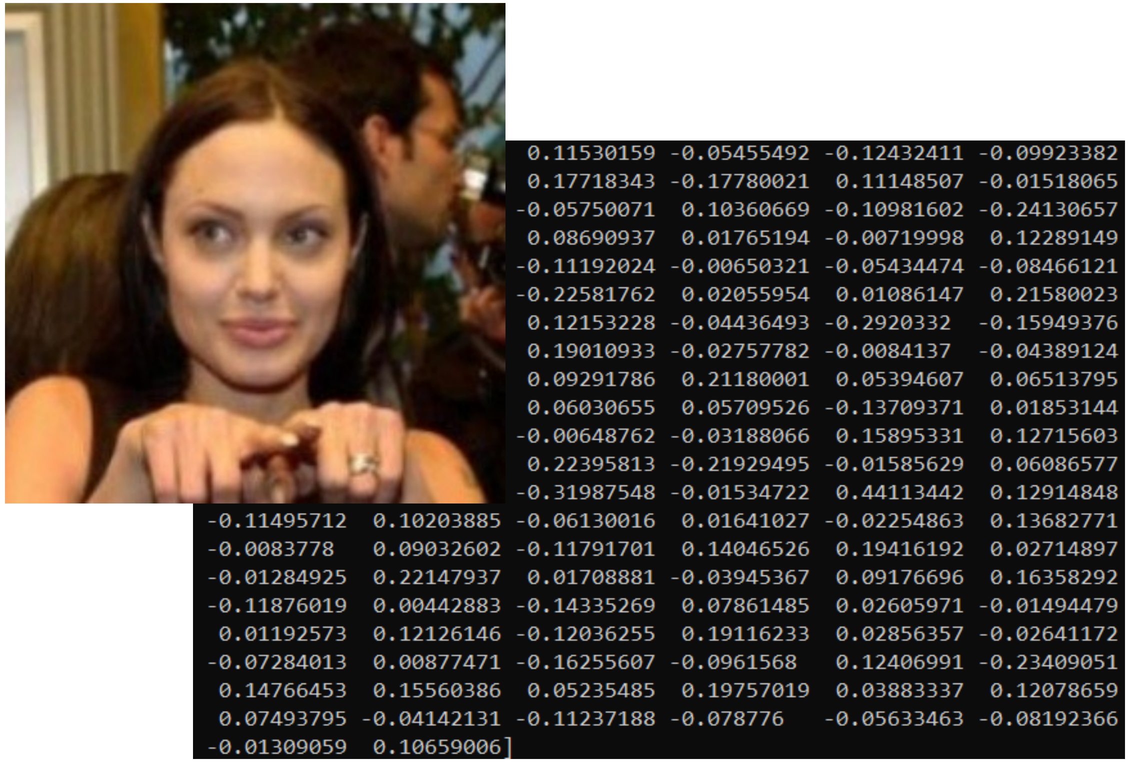
Objective: obtain images with more similar starting characteristics.

Face alignment



Feature Extraction →

The preprocessed image is mapped into the feature space, where the features of the same identity are close, and those of different identities are far apart.



Binarization →

The real-valued feature vector is converted into a binary vector

```
[-0.21192636  0.01090579  0.11530159 -0.05455492 -0.12432411 -0.09923382
 -0.11360563 -0.17770834  0.17718343 -0.17780021  0.11148507 -0.01518065
 -0.26995713  0.03514797 -0.05750071  0.10360669 -0.10981602 -0.24130657
 -0.03885939 -0.09809047  0.08690937  0.01765194 -0.00719998  0.12289149
 -0.2159812  -0.29686308 -0.11192024 -0.00650321 -0.05434474 -0.08466121
  0.03594787  0.05778139 -0.22581762  0.02055954  0.01086147  0.21580023
  0.02652583 -0.11954784  0.12153228 -0.04436493 -0.2920332  -0.15949376
  0.09857918  0.23736741  0.19010933 -0.02757782 -0.0084137  -0.04389124
  0.09397443 -0.29531577  0.09291786  0.21180001  0.05394607  0.06513795
  0.08195408 -0.24346486  0.06030655  0.05709526 -0.13709371  0.01853144
  0.11546178 -0.10312104 -0.00648762 -0.03188066  0.15895331  0.12715603
 -0.14583316 -0.14995976  0.22395813 -0.21929495 -0.01585629  0.06086577
 -0.16153075 -0.22271793 -0.31987548 -0.01534722  0.44113442  0.12914848
 -0.11495712  0.10203885 -0.06130016  0.01641027 -0.02254863  0.13682771
 -0.0083778   0.09032602 -0.11791701  0.14046526  0.19416192  0.02714897
 -0.01284925  0.22147937  0.01708881 -0.03945367  0.09176696  0.16358292
 -0.11876019  0.00442883 -0.14335269  0.07861485  0.02605971 -0.01494479
  0.01192573  0.12126146 -0.12036255  0.19116233  0.02856357 -0.02641172
 -0.07284013  0.00877471 -0.16255607 -0.0961568   0.12406991 -0.23409051
  0.14766453  0.15560386  0.05235485  0.19757019  0.03883337  0.12078659
  0.07493795 -0.04142131 -0.11237188 -0.078776   -0.05633463 -0.08192366
 -0.01309059  0.10659006]
```

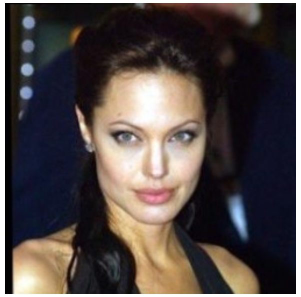
0 1 1 0 0 0 0 0 1 0 1 0 0 1 0 1 0 0 0 0 1 1 0 1 0
0 0 0 0 0 1 1 0 1 1 1 1 0 1 0 0 0 1 1 1 0 0 0 1 0
1 1 1 1 1 0 1 1 0 1 1 0 0 0 1 1 0 0 1 0 0 1 0 0 0
0 1 1 0 1 0 1 0 1 0 1 0 1 1 1 0 1 1 0 1 1 0 1 0 1
1 0 1 1 0 1 1 0 0 1 0 0 1 0 1 1 1 1 1 1 1 0 0 0 0
0 0 1

Binarization →

The real-valued feature vector is converted into a binary vector

0 1 1 0 0 0 0 0 1 0 1 0 0 1 0 1 0 0 0 0 1 1 0 1 0 0 0 0 0 0 1 1 0 1 1 1
1 0 1 0 0 0 1 1 1 0 0 0 1 0 1 1 1 1 1 0 1 1 0 1 1 0 0 0 1 1 0 0 1 0 0 1
0 0 0 0 1 1 0 1 0 1 0 1 0 1 0 1 1 1 0 1 1 0 1 1 0 1 0 1 1 0 1 1 0 1 1 0
0 1 0 0 1 0 1 1 1 1 1 1 1 0 0 0 0 0 0 1

0 1 1 0 0 0 0 0 1 0 1 0 0 1 0 1 0 0 0 0 1 1 1 1 0 0 0 1 0 0 1 1 0 0 0 1
0 0 1 1 0 0 1 1 1 1 1 0 1 0 1 1 1 1 1 0 1 1 0 1 1 0 1 0 1 1 0 0 1 0 0 1
0 0 0 0 1 1 0 1 0 1 1 1 1 0 1 1 0 0 1 1 0 1 1 0 1 0 1 0 0 0 1 0 1 1 1 1
0 0 0 0 1 0 1 1 1 1 1 1 1 0 0 0 0 0 1 1



Small intra-user variation

0 1 1 0 0 0 0 0 1 0 1 0 0 1 0 1 0 0 0 0 1 1 0 1 0 0 0 0 0 0 1 1 0 1 1 1
1 0 1 0 0 0 1 1 1 0 0 0 1 0 1 1 1 1 1 0 1 1 0 1 1 0 0 0 1 1 0 0 1 0 0 1
0 0 0 0 1 1 0 1 0 1 0 1 0 1 0 1 1 1 0 1 1 0 1 1 0 1 0 1 1 0 1 1 0 1 1 0
0 1 0 0 1 0 1 1 1 1 1 1 1 0 0 0 0 0 0 1

0 1 1 1 0 1 0 0 1 0 1 0 0 1 0 1 0 0 0 0 0 1 1 0 0 0 0 0 1 0 0 1 0 1 1 1
0 0 1 1 0 1 1 1 1 0 1 0 1 0 1 1 1 1 1 0 1 1 0 1 1 1 0 0 1 1 0 0 1 0 0 1
0 0 0 1 1 1 0 0 0 1 1 0 0 0 0 1 1 0 1 1 0 0 1 0 0 0 0 0 0 1 1 0 1 0 0 0
1 0 0 1 1 0 1 1 1 1 1 1 1 1 0 0 0 0 0 0

Small inter-user similarity

Helper Data →

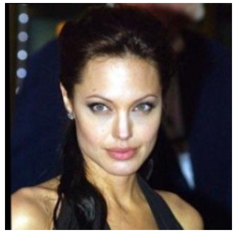
They handle the intra-class variations

Bit sequence generated during enrollment (original key)

0 1 1 0 0 0 0 0 1 0 1 0 0 1 0 1 0 0 0 0 1 1 0 1 0 0 0 0 0 0 1 1 0 1 1 1 1 0 1 0 0 0 1 1 1 0 0 0 1 0 1 1 1 1 1
0 1 1 0 1 1 0 0 0 1 1 0 0 1 0 0 1 0 0 0 0 1 1 0 1 0 1 0 1 0 1 0 1 1 1 0 1 1 0 1 1 0 1 0 1 1 0 1 1 0 1 1 0 0 1
0 0 1 0 1 1 1 1 1 1 1 0 0 0 0 0 0 1

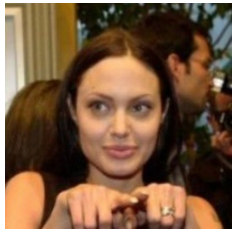
Bit sequence generated during verification phase

0 1 1 0 0 0 0 0 1 0 1 0 0 1 0 1 0 0 0 0 1 1 1 1 0 0 0 1 0 0 1 1 0 0 0 1 0 0 1 1 0 0 1 1 1 1 1 0 1 0 1 1 1 1 1
0 1 1 0 1 1 0 1 0 1 1 0 0 1 0 0 1 0 0 0 0 1 1 0 1 0 1 1 1 1 1 0 1 1 0 0 1 1 0 1 1 0 1 0 1 0 0 0 1 0 1 1 1 0 0
0 0 1 0 1 1 1 1 1 1 1 0 0 0 0 0 1 1



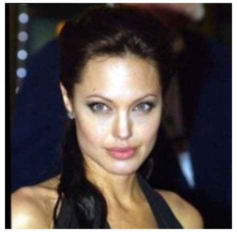
Bit sequence generated during enrollment (original key)

0 1 1 0 0 0 0 0 1 0 1 0 0 1 0 1 0 0 0 0 1 1 0 1 0 0 0 0 0 0 1 1 0 1 1 1 1 0 1 0 0 0 1 1 1 0 0 0 1 0 1 1 1 1 1
0 1 1 0 1 1 0 0 0 1 1 0 0 1 0 0 1 0 0 0 0 1 1 0 1 0 1 0 1 0 1 0 1 1 1 0 1 1 0 1 1 0 1 0 1 1 0 1 1 0 1 1 0 0 1
0 0 1 0 1 1 1 1 1 1 1 0 0 0 0 0 0 0 1



Bit sequence generated during verification phase

0 1 1 0 0 0 0 0 1 0 1 0 0 1 0 1 0 0 0 0 1 1 1 1 0 0 0 1 0 0 1 1 0 0 0 1 0 0 1 1 0 0 1 1 1 1 1 0 1 0 1 1 1 1 1
0 1 1 0 1 1 0 1 0 1 1 0 0 1 0 0 1 0 0 0 0 1 1 0 1 0 1 1 1 1 1 0 1 1 0 0 1 1 0 1 1 0 1 0 1 0 0 0 1 0 1 1 1 0 0
0 0 1 0 1 1 1 1 1 1 1 0 0 0 0 0 0 1 1



+ Error Correction Codes (ECCs)

Only if the number of different bits is less than or equal to the chosen threshold T!

=

Original key

0 1 1 0 0 0 0 0 1 0 1 0 0 1 0 1 0 0 0 0 1 1 0 1 0 0 0 0 0 0 1 1 0 1 1 1 1 0 1 0 0 0 1 1 1 0 0 0 1 0 1 1 1 1 1
0 1 1 0 1 1 0 0 0 1 1 0 0 1 0 0 1 0 0 0 0 1 1 0 1 0 1 0 1 0 1 0 1 1 1 0 1 1 0 1 1 0 1 0 1 1 0 1 1 0 1 1 0 0 1
0 0 1 0 1 1 1 1 1 1 1 0 0 0 0 0 0 0 1

Key Generation Process
Evaluating Information Extraction Repeatability

Objectives:

- Understand the influence of the implemented quality filter on the repeatability of the extracted information;

- Understand which feature extraction mode among those analyzed can be used effectively to generate a cryptographic key.

  - *Dlib* (length of the extracted feature vector: 128);
  - *FaceNet* (length of the extracted feature vector: 128);
  - *Dlib+FaceNet*, a simple concatenation of the previous models (length of the extracted feature vector: 256);
  - *FaceNet512* (length of the extracted feature vector: 512).

Labeled Faces in the Wild (LFW)

Variations in pose, illumination, expression, background, race, ethnicity, age, gender, clothing, hairstyles, camera quality, color saturation, focus, and other parameters.

AT&T

Face images have minimal variations (scenario of a user assuming the correct pose for using the Biometric Cryptosystem).

- *AT&T*
  Number of images: 400
  Number of subjects: 40
  Number of genuine claims: 360
  Number of impostor claims: 15600

Evaluation of the effect of different quality

- LFW-0
  Filter parameters: `SP_min=0, yaw_distortion_max=1, SI_min=0`
  [*Basic Filter*]
  Number of images: 8604
  Number of subjects: 1591
  (Note: This is the baseline case without restrictions on pose and illumination quality; the database is narrowed down only with subjects with at least two face images that have passed the detection stage, as explained earlier.)
- LFW-1
  Filter parameters: `SP_min=0.8, yaw_distortion_max=0.2, SI_min=0.5`
  Number of images: 3809
  Number of subjects: 792
- LFW-2
  Filter parameters: `SP_min=0.9, yaw_distortion_max=0.15, SI_min=0.5`
  Number of images: 778
  Number of subjects: 204
- LFW-3
  Filter parameters: `SP_min=0.93, yaw_distortion_max=0.1, SI_min=0.6`
  Number of images: 126
  Number of subjects: 47

- *Dlib* (length of the extracted feature vector: 128);
- *FaceNet* (length of the extracted feature vector: 128);
- *Dlib+FaceNet*, a simple concatenation of the previous models (length of the extracted feature vector: 256);
- *FaceNet512* (length of the extracted feature vector: 512).

With dlib+FaceNet usinf LFW

- LFW-2
  Number of images: 778
  Number of subjects: 204
  Number of genuine claims: 573
  Number of impostor claims: 157731

threshold	FAR (%)	FA	FRR (%)	FR	GAR (%)	GA
<13	0.0	0	100.0	573	0.0	0
13	0.0	0	99.825	572	0.175	1
14	0.0	0	99.825	572	0.175	1
15	0.0	0	99.476	570	0.524	3
16	0.0	0	99.302	569	0.698	4
17	0.0	0	99.127	568	0.873	5
18	0.0	0	98.778	566	1.222	7
19	0.0	0	98.08	562	1.92	11
20	0.0	0	97.906	561	2.094	12
21	0.0	0	97.557	559	2.443	14
22	0.0	0	97.208	557	2.792	16
23	0.0	0	96.51	553	3.4[illegible]	[illegible]
24	0.0	0	95.637	548	4.[illegible]	[illegible]
25	0.0	0	94.59	542	5.[illegible]	[illegible]
26	0.0	0	93.717	537	6.283	36
27	0.0	0	92.67	531	7.33	42
28	0.0	0	90.75	520	9.25	53
29	0.0	0	89.878	515	10.122	58
30	0.0	0	88.831	509	11.169	64
31	0.0	0	86.387	495	13.613	78
32	0.0	0	84.293	483	15.707	90
33	0.0	0	82.024	470	17.976	103
34	0.0	0	79.232	454	20.768	119
35	0.0	0	76.614	439	23.386	134
36	0.0	0	72.949	418	27.051	155
37	0.0	0	69.284	397	30.716	176
38	0.0	0	64.921	372	**35.079**	201
39	0.002	3	62.653	359	37.347	214
40	0.003	4	[illegible]	[illegible]	[illegible]	[illegible]

38	0.0	0	64.921	372	**35.079**	201

- LFW-3
  Number of images: 126
  Number of subjects: 47
  Number of genuine claims: 79
  Number of impostor claims: 5796

threshold	FAR (%)	FA	FRR (%)	FR	GAR (%)	GA
<21	0.0	0	100.0	79	0.0	0
21	0.0	0	97.468	77	2.532	2
22	0.0	0	96.203	76	3.797	3
23	0.0	0	93.671	74	6.329	5
24	0.0	0	92.405	73	7.595	6
25	0.0	0	91.139	72	8.861	7
26	0.0	0	91.139	72	8.861	7
27	0.0	0	91.139	72	8.861	7
28	0.0	0	88.608	70	11.392	9
29	0.0	0	87.342	69	12.658	10
30	0.0	0	86.076	68	13.924	11

38	0.0	0	49.367	39	**50.633**	40

threshold	FAR (%)	FA	FRR (%)	FR	GAR (%)	GA
33	0.0	0	77.215	61	22.785	[illegible]
34	0.0	0	70.886	56	29.114	23
35	0.0	0	67.089	53	32.911	26
36	0.0	0	60.759	48	39.241	31
37	0.0	0	53.165	42	46.835	37
38	0.0	0	49.367	39	**50.633**	40
39	0.035	2	48.101	38	51.899	41
40	0.035	2	46.835	37	53.165	42

… but in controlled conditions

- *AT&T*
  Number of images: 400
  Number of subjects: 40
  Number of genuine claims: 360
  Number of impostor claims: 15600

threshold	FAR (%)	FA	FRR (%)	FR	GAR (%)	GA
<8	0.0	0	100.0	360	0.0	0
8	0.0	0	99.722	359	0.278	1
9	0.0	0	98.889	356	1.111	4
10	0.0	0	98.611	355	1.389	5
11	0.0	0	97.5	351	2.5	9
12	0.0	0	97.5	351	2.5	9
13	0.0	0	95.833	345	4.167	15
14	0.0	0	94.722	341	5.278	19
15	0.0	0	92.778	334	7.222	26
16	0.0	0	90.556	326	9.444	34
17	0.0	0	89.167	321	10.833	39
18	0.0	0	86.111	310	13.889	50
19	0.0	0	83.333	300	16.667	60
20	0.0	0	79.722	287	20.278	73
21	0.0	0	76.111	274	23.889	86
22	0.0	0	71.389	257	28.611	103
23	0.0	0	68.056	245	31.944	115
24	0.0	0	64.722	233	35.278	127
25	0.0	0	63.056	227	36.944	133
26	0.0	0	59.444	214	40.556	146
27	0.0	0	57.222	206	42.778	154
28	0.0	0	54.444	196	45.556	164
29	0.0	0	50.833	183	49.167	177
30	0.0	0	48.333	174	51.667	186
31	0.0	0	45.833	165	54.167	195
32	0.0	0	41.944	151	58.056	209
33	0.0	0	38.889	140	61.111	220
34	0.0	0	35.278	127	64.722	233
45	0.0	0	11.944	43	**88.056**	317
37	0.0	0	26.667	96	73.333	264
38	0.0	0	25.278	91	74.722	269
39	0.0	0	23.611	85	76.389	275
40	0.0	0	19.444	70	80.556	290
41	0.0	0	16.944	61	83.056	299
42	0.0	0	15.0	54	85.0	306
43	0.0	0	14.167	51	85.833	309
44	0.0	0	12.778	46	87.222	314
45	0.0	0	11.944	43	**88.056**	317
46	0.006	1	10.833	39	89.167	321
47	0.006	1	9.444	34	90.556	326