Recognition of complex gestures for real-time emoji assignment

Rosa Zuccarà, Alessandro Ortis[0000−0003−3461−4679], and Sebastiano Battiato[0000−0001−6127−2470]

Department of Mathematics and Computer Science
University of Catania, Italy
rosazuccara@outlook.com,{ortis,battiato}@dmi.unict.it

Abstract. Gesture recognition allows humans to interface and interact naturally with the machine. This paper presents analytical and algebraic methods to recognize specific combinations of facial expressions and hand gestures, including interactions between hands and face. The methodologies for extracting the features for both faces and hands were implemented starting from landmarks identified in real-time by the MediaPipe framework. To benchmark our approach, we selected a large set of emoji and designed a system capable of associating chosen emoji to facial expressions and/or hand gestures recognized. Complex poses and gestures combinations have been selected and assigned to specific emoji to be recognized by the system. Furthermore, the Web Application we created demonstrates that our system is able to quickly recognize facial expressions and complex poses from a video sequence from standard camera. The experimental results show that our proposed methods are generalizable, robust and achieve on average 99,25% of recognition accuracy.

Keywords: Facial expressions · Hand gestures · Real time recognition · Emoji

1 Introduction and Motivations

Human communication is often a complex combination of facial expressions, hand gestures and speech, all of which contribute significantly to a spoken message [1]. Facial expression is one of the main ways by which human beings communicate their intentions and emotions. For this reason, Facial Expression Recognition (FER) is very important and can be used in many applications such as driver safety, healthcare, video conferencing, virtual reality, and Human-Machine Interface (HMI) [2,3,4]. The state of the art of the FER shows that most of the designed systems perform three phases: image pre-processing, feature extraction, expression classification. The calculation of the feature vector describing a facial expression is often based on the landmarks estimated by face detection algorithms. Classification is usually done by training a learning model. The method proposed in [5] uses an estimator of 68 facial landmarks and calculates a feature vector of distance by a selected set of landmarks around mouth area and eyes since muscles in those areas change with facial expressions. The difference between emotional and neutral feature vectors is considered as final features to identify emotions using Random Forest Classifier. In the work in [6] feature vector is constructed using all 68 estimated landmarks and considering for each: coordinates, the distance from a fixed point and the direction of the direct line towards the fixed point. Facial expression recognition was achieved using Support Vector Classifier (SVC). The paper [7] presents the FER method based on Convolution Neural Network (CNN). Considering the complexity of the system, for CNN it is necessary to collect a huge amount of data, so that the trained network has better

generalization performance and can reduces overfitting. Hand detection plays an important role in hand gesture recognition for applications such as sign language recognition, Human-Computer Interaction (HCI), driver hand behaviour monitoring and virtual/augmented reality interaction [8,9]. Gupta et al. [10] proposed a method for the "static hand gesture" recognition based on 15 local Gabor filters, to reduce the complexity, followed to a combination of Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) for make system invariant to scale and rotation problem. Classification of gestures is done with the help of a one-against-one multiclass Support Vector Machine (SVM). Chen et al. [11] introduced a "hand dynamic gesture" recognition system from 2D video. First is applied a real-time hand tracking algorithm, then is calculated the vector of spatial and temporal features that is as the input to the Hidden Markov Model (HMM) based recognition system. Koh [12] designs a two-stage hand gesture recognition system (trajectory-based, shape-based) for distinguishes 15 user-defined hand gestures that are highly representative to Visual Communication Markers (VCMs) such as emoji. After different steps of image pre-processing and feature extraction, the classification is performed by Machine Learning (ML) algorithms. Some research efforts focus on integrating more aspects of communication, such as facial expression, hand gestures or speech recognition activities [13]. The researchers dealt with both the extraction phase of the characteristics and the phase of recognition of the expression or hand gestures using ML algorithms. The drawbacks of the solutions based on Deep Learning (DL) models are often related to high computational requirements, or specific hardware needs to perform the inferences. Indeed, research solutions are often time and computational expensive. This represents the main gap between research state of the art and its applications for the deployment of services and feasible solutions. Therefore, in our research on the recognition of complex combinations of facial expressions and gestures, the recognition phase was carried out using the characteristics calculated by analytical and algebraic methods. The development of an efficient recognition system must overcome challenges in the face or hand detection phase such as: the segmentation, in the presence of complex backgrounds in which there are many objects in an image, the representation of the local form of the hand [14], in the representation of the global configuration of the body and the modelling of the gestural sequence [15]. Well-performing face and hand detection models that also guarantee to be performed quickly in real-time and on mobile devices are respectively: MediaPipe Face Mesh [16] and MediaPipe Hands [17], both available in MediaPipe, an open-source framework for building multimodal and cross-platform applied ML pipelines. The ML pipeline of the two solutions: "MediaPipe Face Mesh" and "MediaPipe Hands" consist of two real-time deep neural network models that work together: a detector that operates on the full image and computes face or hand locations and a 3D face landmark model or 3D hand landmark model that operates on those locations and predicts the approximate surface geometry via regression. MediaPipe Hands infers 21 3D landmarks of a hand, while MediaPipe Face Mesh estimates 468 3D face landmarks. With the aim to recognize complex gestures, we defined a pool of proper analytical geometry methods on the landmarks estimated by the MediaPipe models avoiding to lean on further DL models for classification. This allowed the implementation of a real-time complex gesture recognition system, which can be executed without high computational or specific hardware requirements.

2 Proposed method

This paper aims to recognize complex combinations of facial expressions and hand gestures, including interactions that could occur between the face and hands. To this end, we defined a set of complex gestures associated to emoji and implemented a gesture-to-emoji recognition system

based on analytical and algebraic methods able to process a video flow in real time. Emojis are pictograms or ideograms used to express emotions through facial images or to describe concepts through images of objects, places, activities, foods, plants, animals. They are usually introduced to emphasize the message in digital conversations and social media post sharing. They are used in text communications to emulate visual cues such as facial expressions, poses and gestures [18]. To recognize hand gestures, we have taken into consideration: if it is the right or left hand, the orientation of the hand (vertical or horizontal), the region (palm or back), the position of the hand in the plane, the bending of the hand, the poses of each finger (closing, folding) and finally the reciprocal position between the fingers (dilation, proximity (touch) or crossing). Whereas features extracted from eyebrows, eyes and mouth have been considered for the facial expression.

2.1 Methods for recognizing facial expressions

To estimate the state of the eye opening (closed, open or wide open) or of the mouth (closed, ajar, open, or wide open) the most significant landmarks are selected and the corresponding 2D coordinate values are used to compute the following AR_1 (i.e., Aspect Ratio) metric:

$$AR_1 = \frac{\sum_{i=1}^{3} d_i}{2d_{C_{12}}} = \frac{\sum_{i=1}^{3} \sqrt{(P_i^u.x - P_i^l.x)^2 + (P_i^u.y - P_i^l.y)^2}}{2\sqrt{(C_1.x - C_2.x)^2 + (C_1.y - C_2.y)^2}} \tag{1}$$

In particular, we extracted the same number of landmarks around the mouth and each eye and applied the same equation for the estimation of the closing degree. Figure 1 shows the landmarks of eyelids, whereas Figure 2 shows the landmarks extracted around the mouth. The metric (Eq. 1)

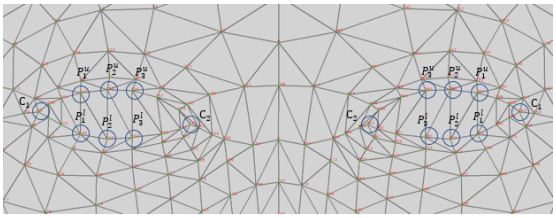


Fig. 1. *Landmarks of eyelids (P_i^k) and of the left and right eye corners (C_j).*

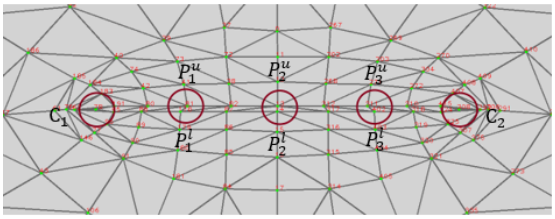


Fig. 2. *Landmarks of the upper and lower inner border (P_i^k) and the right and left angles of the mouth vermilion (C_j).*

is partially insensitive to the small variations in the proportions of the eyes or mouth that occur between different individuals, it is invariant with respect to uniform resizing of the image and rotation of the face in the plane. For the eyes this invariance is obtained by normalizing the sum of the distances between the contours of the eyelids with respect to the distance between the corners of the eye (see Figure 1). Whereas for the mouth, the landmarks taken into consideration are those relating to the inner edge and the corners of the vermilion (see Figure 2). A similar feature was suggested in [19] to measure the eye blink and for correct recognition a classifier that takes a larger temporal window of a frame into account is trained. Instead, in our approach we have empirically

established the threshold values used to determine the appropriate state of opening the mouth or eyes, collecting the values of the "Aspect Ratio" extracted from several acquisition sessions with different participants. It has been observed that the value of the metric increases with increasing eye or mouth opening. The separation values between the states are determined as the average between two averages of the adjacent states. Intervals of values that define the eye-opening states are determined as follows: less than 0.5 for closed eyes, between 0.5 and 0.7 for open eyes and greater than 0.7 for wide eyes. Intervals of values that define the mouth-opening states are determined as follows: less than 0.22 for closed mouth, between 0.22 and 0.55 for ajar mouth, between 0.55 and 0.9 for open mouth and greater than 0.9 for wide open mouth. Oral commissures are the points

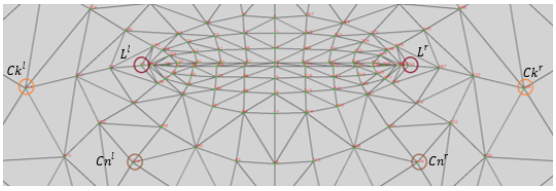


Fig. 3. *Landmarks of the oral commissures (L^s), of the cheek (Ck^s) and of the chin (Cn^s) in the left and right side of the face.*

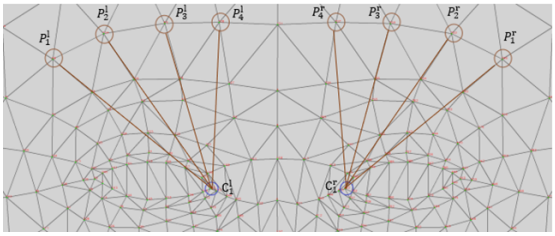


Fig. 4. *Landmarks (C_1^s) of the inner corner of the right and left eye and landmarks of the lower part of the forehead (P_i^s).*

where the upper and lower lips meet, usually known as the corners of the mouth. Happiness and sadness are detected comparing the positions of the landmarks of the oral commissures with the are compared with those of the chin and cheek, as detailed in the following Definition 1 and Definition 2 (see Figure 3), respectively.

Definition 1 (Smiling lips). $L^l.y < Ck^l.y$ *and* $L^l.x < Cn^l.x$, $L^r.y < Ck^r.y$ *and* $L^r.x > Cn^r.x$

Definition 2 (Sad lips). $L^l.y > Ck^l.y$ *and* $L^l.x > Cn^l.x$, $L^r.y > Ck^r.y$ *and* $L^r.x < Cn^r.x$

To determine whether a subject has raised or lowered eyebrows, landmarks of the inner corners of the eyes were selected because these points are stable in the two eyebrow poses. In addition, landmarks of the lower forehead were selected because their position varies with the occurrence of the muscular movements of the eyebrows (see Figure 4). We have built a metric AR_2 robust enough to be invariant under different factors such as face size or the distance between the face and the camera. Indeed, as shown in Equation 2, the sum of the distances between inner corner eye and points of the forehead is normalized by the distance between the landmarks C_1^l and P_4^l (left side) or between C_1^r and P_4^r (right side).

$$AR_2^s = \frac{\sum_{i=1}^{4} d_i^s}{2d_4^s} = \frac{\sum_{i=1}^{4} \sqrt{(P_i^s.x - C_1^s.x)^2 + (P_i^s.y - C_1^s.y)^2}}{2\sqrt{(P_4^s.x - C_1^s.x)^2 + (P_4^s.y - C_1^s.y)^2}} \qquad s = l \text{ (left side) or } s = r \text{ (right side)} \tag{2}$$

A similar metric is proposed by the authors in [5] who have developed a method to compute, normalize and extract a feature vector which represent facial emotions that are classified using

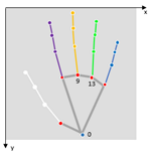
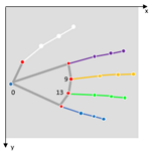
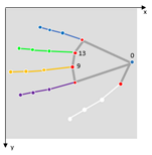
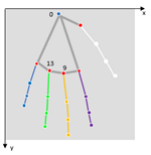
the random forest-based classification technique. While in our approach the threshold values are established to determine the states that indicate the positions of the eyebrows ("normal", "raised", "lowered") and these are set using the same procedure seen for the other components of the face, described above.

2.2 Methods for recognizing hand gestures

For each hand detected by the MediaPipe Hand detector, the characteristics that describe its configuration are extracted. Then, approaches are proposed to determine the orientation, the fold, the position in the plane of the hand and to identify the palm or the back. Furthermore, for each finger, the closure, the folding, possible reciprocal position between the other fingers and alignment with respect to the palm is established.

Orientation: vertical or horizontal: to determine the hand orientation, three fixed landmarks (marked as 0, 9, 13) of the rigid part of the hand are considered and the coordinate values of these points are compared. Table 1 shows the various conditions for establishing the orientation of the hand.

Table 1. Conditions for establishing the orientation of the hand. The notation "0.y" stands for: coordinate y of the point marked as 0.

Vertical orientation with fingers pointing up	Horizontal orientation with fingers pointing to the right	Horizontal orientation with fingers pointing to the left	Vertical orientation with fingers pointing down
Orientation: 1	Orientation: 2 Slice: top	Orientation: 2 Slice: down	Orientation: 3
$0.y \geq 9.y$ and $0.y > 13.y$	$9.y \leq 0.y$ and $0.y \leq 13.y$	$13.y \leq 0.y$ and $0.y < 9.y$	$0.y < 9.y$ and $0.y < 13.y$
			

Hand region: palm or back: the MediaPipe hand detector can distinguish the right hand from the left one. This information, estimated by the detector, together with the orientation, assigned by our algorithm, is exploited to determine whether the palm or the back of the hand is shown. In particular, we observed that in each orientation of the left or right hand, the point to be examined corresponds to the metacarpal of the thumb (i.e., landmark 1). We thus found relationships between the coordinates of this point and the coordinates of the landmarks of the rigid part of the hand, marked as 0, 5, 9, 13, and 17. We have noticed that these relationships vary within certain regions of the plane, thus allowing it to be divided into "slices". In this way, when the hand is positioned in the XY plane of the "world" with a certain inclination, the algorithm assigns to it the appropriate "slice" in which it lies. In particular, for "Orientation 1" and "Orientation 3", the x (or y) coordinate of landmark 1 is compared with the maximum or minimum computed from the set of x (or y) coordinate values of the metacarpal landmarks, marked as 1, 5, 9, 13, 17. Furthermore, a similar

comparison is performed for the landmark 0, corresponding to the wrist of the hand[1]. Figure 5 shows an example of the conditions for the palm of the left hand positioned in "Orientation 1". For the hand positioned in "Orientation 2", to establish the region (palm or back) the coordinates of the landmarks 0 and 1 are compared, while to determine thumb-up or thumb-down the comparison is performed between the landmarks 0, 9, and 13. Figure 6 shows an example of the conditions for the back of the right hand positioned horizontally with the thumb up or down.

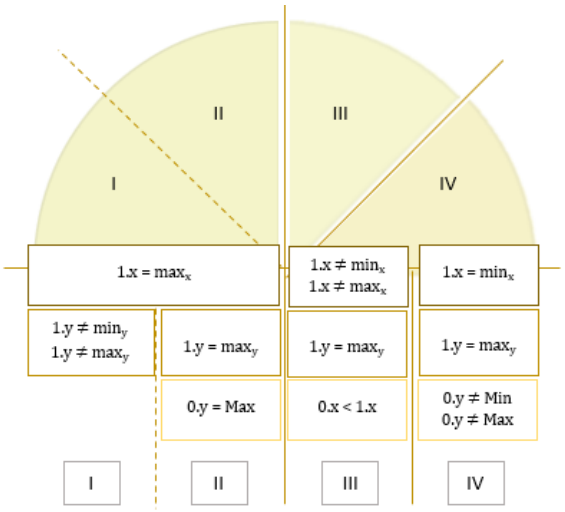


Fig. 5. Conditions for identifying the palm of the left hand positioned in "Orientation 1".

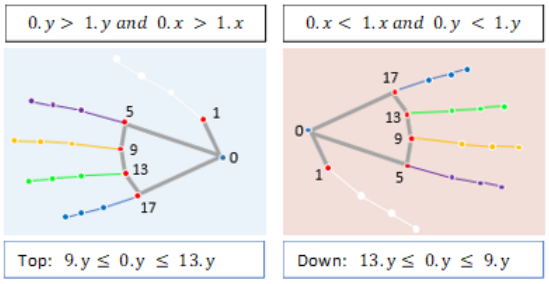


Fig. 6. *Conditions that determine the back of the right hand positioned in "Orientation 2".*

Alignment of the finger with respect to the metacarpus: the direction of the fingers with respect to the metacarpus was analysed to establish the alignment of the fingers to the palm. Then we calculate the normal to the plane of the palm formed by two vectors having as extremes, respectively the landmarks 0 and 5 and the landmarks 0 and 17. For each of the four fingers, the unit direction vectors of the straight lines passing through the reference points corresponding to MCP and tip were determined. The alignment was examined by calculating the angle of inclination of the unit vector of the normal with respect to the unit vector of direction of the finger. We define the hand in the "straight state" when the four fingers have an inclination of no less than 70°.

Recognizing the fingers in the "closed" state: the thumb is defined to be in the "closed" state when at least one of its parts (i.e., TIP, PIP, MCP) reaches the top of the palm region. Whereas the remaining four fingers are defined in the "closed" state when their tip reaches the palm of the hand. To simulate this mechanism, the following two Regions of Interest (RoI) have been defined:

[1] max_x=max{1.x,5.x,9.x,13.x,17.x}, min_x=min{1.x,5.x,9.x,13.x,17.x}, max_y=max{1.y,5.y,9.y,13.y,17.y}, min_y=min{1.y,5.y,9.y,13.y,17.y}, Max=max{0.y,1.y,5.y,9.y,13.y,17.y}, Min=min{0.y,1.y,5.y,9.y,13.y,17.y}.

Definition 3 (Thumb Closing RoI). *Thumb Closing RoI is the region defined around the landmarks 5, 9, 13, and 17 and having the vertices in the points: A=(min_x, min_y), B=(max_x, min_y), C=(min_x, max_y), D=(max_x, max_y). Where:*
min_x=min{5.x,9.x,13.x,17.x}-offset, min_y=min{5.y,9.y,13.y,17.y}-offset,
max_x=max{5.x,9.x,13.x,17.x}+offset, max_y=max{5.y,9.y,13.y,7.y}+offset.

Definition 4 (Finger Closing RoI). *Finger Closing RoI is the region defined around the landmarks 1, 5, 9, 13, and 17 and having the vertices in the points: A=(min_x, min_y), B=(max_x, min_y), C=(min_x, max_y), D=(max_x, max_y). Where:*
min_x=min{1.x,5.x,9.x,13.x,17.x}-offset, min_y=min{1.y,5.y,9.y,13.y,17.y},
max_x=max{1.x,5.x,9.x,13.x,17.x}+offset, max_y=max{1.y,5.y,9.y,13.y,17.y}.

Therefore, the thumb is in the "closed" state when at least one of its landmarks (i.e., 2, 3, 4) is included in the Thumb Closing RoI. Each of the four fingers (thumb excluded) is defined as being in the "closed" state when the landmark corresponding to the "tip" is included in the Finger Closing RoI. The proposed approach is invariant to the size, different positions and inclinations of the hand, and to the distance between the hand and the camera.

Recognizing fingers in the "bent" state: another movement considered for the fingers is the bending of the three phalanges. To describe this behaviour, a similar approach to the one described above has been implemented. In fact, for each finger three regions have been determined: one which includes the landmarks corresponding to three interphalangeal joints (i.e., MCP, PIP, DIP), the second innermost which includes two of the joints (i.e., PIP, MCP) and the third which presents the MCP joint. To define the finger in the "bent" state, at least one of the three conditions must occur:

- Condition 1[2]: $min_x \leq tip.x \leq max_x$ and $min_y \leq tip.y \leq max_y$
- Condition 2[3]: $min_x \leq dip.x \leq max_x$ and $min_y \leq dip.y \leq max_y$
- Condition 3[4]: $min_x \leq pip.x \leq max_x$ and $min_y \leq pip.y \leq max_y$

Reciprocal position between the fingers: dilation, proximity and crossing are the reciprocal position between the fingers. To recognize when the fingers are two by two dilated, the distance of the reference point MCP_i from the line passing through the points MCP_{i+1} and TPC_{i+1} of the adjacent finger was compared with the distance of the reference point TCP_i from the same line. If the distance calculated from the TCP_i landmark is greater than that calculated by the MCP_i landmark, then the pair of fingers is defined in the "dilated" state. To describe the junction of two fingertips, a region (RoI) has been defined around one of the two fingertips and then a check is executed to see if the tip of the other finger belongs to that region. To represent the two crossed fingers, a region (RoI) around the three interphalangeal joints (TIP, DIP, PIP) was defined on one of the two fingers, and then it was checked whether the landmark DIP of the other finger belongs to that region.

[2] max_x=max{mcp.x,pip.x,dip.x}+offset, min_x=min{mcp.x,pip.x,dip.x}-offset,
max_y=max{mcp.y,pip.y,dip.y}+offest, min_y=min{mcp.y,pip.y,dip.y}-offset.

[3] max_x=max{mcp.x,pip.x}+offest, min_x=min{mcp.x,pip.x}-offset,
max_y=max{mcp.y,pip.y}+offset, min_y=min{mcp.y,pip.y}-offest.

[4] max_x={mcp.x}+offset2, min$_x$={mcp.x}-offset2, max_y={mcp.y}+offest2, min_y={mcp.y}-offset2.

Interaction between face and hands: the method proposed to analyse the interaction between the face and the hands is to trace regions (RoI-s) around specific landmarks of the face, and then map the coordinates of the vertices of the RoI-s on the image plane. So, when significant landmarks of the hand, also mapped on the image plane, satisfy the conditions of belonging to these regions, then a "hand-face" interaction is considered to have taken place. This method has the characteristic of being invariant to the scale and inclinations of poses.

Head tilt: the inclination of the head was determined by calculating the angle (α) between the x-axis of the image plane and the straight line passing through the two landmarks corresponding to the vertex of the head and the vertex of the chin. Condition to define the head in the "inclined" state is $|\alpha| \leq \frac{\pi}{2.5}$.

3 Real-time Emoji Assignment

The above detailed methods have been implemented in a real-time video analysis process. At each frame, hands and face landmarks are extracted and related features are stored, considering the structures detailed in Table 2 and Table 3 for face (eyebrows, eyes, mouth) and hands respectively.

Table 2. Dictionary for facial components.

	Keys	Values
eyes_dict	Left	close, open, wide open
	Right	close, open, wide open
	State	closeLeft, closeRight, closeTwo, openTwo, wideLeft, wideRight, wideTwo
mouth_dict	State	close, ajar, open, wide
	State_Lips	smile, sad
eyebrows_dict	Left	normal, raised, lowered
	Right	normal, raised, lowered
	State	normalTwo, upperLeft, upperRight, upperTwo, upLow, lowerLeft, lowerRight, upperTwo, lowUp

Table 3. Dictionary for hand gestures.

Objects	Keys	Subkeys	Values
Left Right	Orientation		1, 2, 3
	Slice		1,2,3, top, down
	Part		palm, back
	Finger_Closed	thumb, first, second, third, fourth	True, False
	Finger_Bent	first, second, third, fourth	True, False
	Dilatate		True, False
	Folded		true, false
	Tag_gesture		All_Closed, 4_No_Closed, All_Bent, All_Stretched, All_Dilatate, All_Together, Hand_Straight

Were selected a large set of emojis and defined the task of assigning the correct emoji to recognized move. In particular, thirteen facial expression emojis were chosen, thirteen types of hand gestures, collecting 208 hand emojis to represent the right or left hand, palm or back, and the possible positions. Furthermore, emojis can be displayed according to the inclination of the hand [5]. To show that the system is able to recognize the possible interactions between hands and face, 17 moves have also been chosen to be represented with emoji (see Table 4).

[5]For further details refer to the complete set of encoded emojis and related gestures reported in the supplementary material available at the following link.

Table 4. Complex emoji and detailed gesture description.

				
Hand: left or right; back **Fingers:** dilated **Tips of:** index finger and middle finger touch chin **Mouth:** closed **Lips:** smile **Eyes:** closed	**Hand:** left or right; palm **Index gesture:** touches nose **Eyes:** open	**Hand:** left or right; back. **Fingers:** dilated **Tips of:** index finger, middle finger touch chin **Mouth:** wide open **Eyes:** closed	**Head:** tilted left or right **Hand:** right or left straight **Hand:** touches sphenoid and cheek **Mouth:** not closed **Eyes:** closed	**Eyes:** wide open **Hands:** left and right **Tips of:** index finger, middle finger, and ring finger touch cheeks
				
Hands: left, right; back **Hand orientation:** vertical upwards **Index finger, middle finger, ring finger, little finger:** not closed **Middle-Index, Middle-Ring:** not dilated **Thumb:** closed **Tips of:** index finger, middle finger touch mouth **Eyes:** open	**Hands:** left, right; back **Hand orientation:** vertical with fingers up **Left hand:** inclined to the right **Right hand:** inclined to the left **Left hand:** touches left eye **Right hand:** touches right eye **Mouth:** closed **Lips:** smile	**Hands:** left and right **Hand orientation:** vertical upwards **Fingers:** not bent **Tips of:** little finger, ring finger, and middle finger touch sphenoid **Eyes:** open **Mouth:** closed **Lips:** smile	**Hand:** left or right; back. **Thumb-Index:** dilated **Middle finger, ring finger, little finger:** closed **Thumb:** touches cheek **Index finger:** touches chin **Eyes:** open **Mouth:** closed or ajar **Lips:** normal or sad	**Hands:** left, right; back **Index gesture:** touch cheeks **Mouth:** closed or **Lips:** sad **Eyes:** closed
				
Eyes: open or closed **Hands:** right, left; palm **Heart:** touch of the fingertips of two hands **Mouth:** closed **Lips:** smile				

4 Experimental Results

Through our Web Application we tested our solution on real-time videos acquired by a common webcam, involving 15 people varying age, gender, with different facial features or proportions of the hands. During the testing each user is asked to perform multiple sequences of complex gestures selected at random and corresponding to five Categories of Emoji: a single facial expression (Cat.1), a single hand gesture (Cat.2), two hands (Cat.3), a face accompanied by at least one hand (Cat.4) and finally an emoji representing the interaction between face and hands (Cat.5). The time available to carry out each pose is 120 seconds. If the required pose is correctly translated to corresponding emoji within this time the matching has success, otherwise it is considered as failed [6]. For each Category of Emoji, the time needed by the participants to achieve success were collected and the statistics are shown in Table 5. The recorded times and accuracy reflect the level of complexity of the move to be represented and the skill of the subject in simulating emoji. The user-independent recognition time is very low. Figure 7 shows for each Category of Emoji six examples of test involving different subjects, with different background, lighting conditions, camera distance and orientation. It is possible to observe how the system is able to recognize complex gestures even with a very large range of variabilities in real-time. Experiments shown that our methods for the recognition of expressions and gestures are generalizable, robust, and achieve on average 99,25% of accuracy.

[6] The video related to a complete test is available on the supplementary material.

Table 5. Time performance and accuracy of the recognition system for Categories of Emoji.

	Min (sec.)	Max (sec.)	Mean (sec.)	Accuracy (%)
Cat.1	0,010969	0,87204	1,295784	100
Cat.2	0,020056	8,657423	0,49301	100
Cat.3	0,015962	46,93312	2,268474	98,76
Cat.4	0,03921	48,07147	3,415971	97,53
Cat.5	0,019948	21,2524	1,014331	100
Avg.				99,25

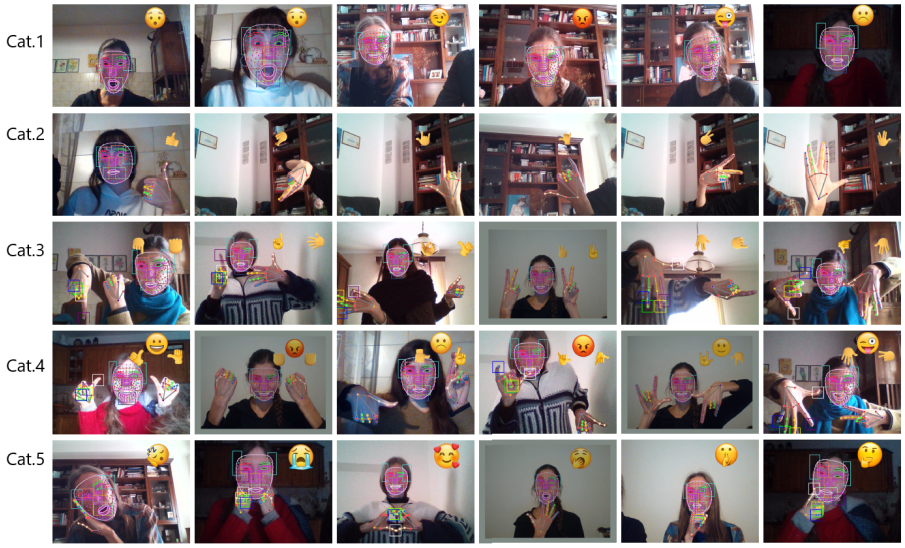


Fig. 7. Examples of complex gesture recognition in real-time for Categories of Emoji.

5 Conclusion

The proposed recognition method is based on the detection models offered by MediaPipe and consists in recognizing complex facial expressions, hand gestures, and interactions between face and hands in real-time. With the aim of recognizing complex gestures, we analysed the components of the face and hands in the various configurations assumed in the different moves. We paid attention to some peculiarities of the hand, being able to distinguish the palm from the back, the orientation and the direction of the hand. We established relation between the landmarks estimated by the detectors and defined a pool of analytical and algebraic methods. This allowed the implementation of a system for recognizing complex gestures in real time, avoiding computational or time expensive approaches. To show the validity of our approach, we have selected a large set of emojis and defined the task of assigning the correct emoji to the recognized gestures. Test results obtained from the real-time video analysis process show that complex gesture recognition system is fast, generalizable to various people and robust to a large set of variabilities. This work represents a first step toward a generalizable real-time complex gesture recognition. In the future we could define a more rigorous evaluation protocol and carry out large-scale experiments.

References

1. Clough, S., & Duff, M. C. (2020). The role of gesture in communication and cognition: Implications for understanding and treating neurogenic communication disorders. Frontiers in Human Neuroscience.
2. Battiato, S., Conoci, S., Leotta, R., Ortis, A., Rundo, F., & Trenta, F. (2020, November). Benchmarking of computer vision algorithms for driver monitoring on automotive-grade devices. In 2020 AEIT International Conference of Electrical and Electronic Technologies for Automotive (AEIT AUTOMOTIVE) (pp. 1-6). IEEE.
3. Altameem, T., & Altameem, A. (2020). Facial expression recognition using human machine interaction and multi-modal visualization analysis for healthcare applications. Image and Vision Computing, 103, 104044.
4. Sourav Dey, Arcoprova Laha, Apurba Paul , Shrijoyee Roy, Suman Paul, 2021, Facial Expression Recognition in Video Call, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) NCETER – 2021 (Volume 09 – Issue 11)
5. Munasinghe, M. I. N. P. (2018, June). Facial expression recognition using facial landmarks and random forest classifier. In 2018 IEEE/ACIS 17th International Conference on Computer and Information Science (ICIS) (pp. 423-427). IEEE.
6. Rohith Raj, S., Pratiba, D., & Ramakanth Kumar, P. (2020). Facial Expression Recognition using Facial Landmarks: A novel approach.
7. Wang, M., Tan, P., Zhang, X., Kang, Y., Jin, C., & Cao, J. (2020, August). Facial expression recognition based on CNN. In Journal of Physics: Conference Series (Vol. 1601, No. 5, p. 052027). IOP Publishing.
8. Matos, A., Filipe, V., & Couto, P. (2016, December). Human-computer interaction based on facial expression recognition: A case study in degenerative neuromuscular disease. In Proceedings of the 7th International Conference on Software Development and Technologies for Enhancing Accessibility and Fighting Info-exclusion (pp. 8-12)
9. Borghi, G., Frigieri, E., Vezzani, R., & Cucchiara, R. (2018, May). Hands on the wheel: a dataset for driver hand detection and tracking. In 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018) (pp. 564-570). IEEE.
10. Gupta, S., Jaafar, J., & Ahmad, W. F. W. (2012). Static hand gesture recognition using local gabor filter. Procedia Engineering, 41, 827-832.
11. Chen, F. S., Fu, C. M., & Huang, C. L. (2003). Hand gesture recognition using a real-time tracking method and hidden Markov models. Image and vision computing, 21(8), 745-758.
12. Koh, J.I. (2020). Developing a Hand Gesture Recognition System for Mapping Symbolic Hand Gestures to Analogous Emoji in Computer-Mediated Communications. Proceedings of the 25th International Conference on Intelligent User Interfaces Companion.
13. Song, N., Yang, H., & Wu, P. (2018, May). A gesture-to-emotional speech conversion by combining gesture recognition and facial expression recognition. In 2018 First Asian Conference on Affective Computing and Intelligent Interaction (ACII Asia) (pp. 1-6). IEEE
14. Liu, N., & Lovell, B. C. (2005, December). Hand gesture extraction by active shape models. In Digital Image Computing: Techniques and Applications (DICTA'05) (pp. 10-10). IEEE.
15. Elmezain, M., Al-Hamadi, A., Pathan, S. S., & Michaelis, B. (2009, September). Spatio-temporal feature extraction-based hand gesture recognition for isolated american sign language and arabic numbers. In 2009 Proceedings of 6th International Symposium on Image and Signal Processing and Analysis (pp. 254-259). IEEE.
16. Kartynnik, Y., Ablavatski, A., Grishchenko, I., & Grundmann, M. (2019). Real-time facial surface geometry from monocular video on mobile GPUs. arXiv preprint arXiv:1907.06724.
17. Zhang, F., Bazarevsky, V., Vakunov, A., Tkachenka, A., Sung, G., Chang, C. L., & Grundmann, M. (2020). Mediapipe hands: On-device real-time hand tracking. arXiv preprint arXiv:2006.10214.
18. Ortis, A., Farinella, G. M., & Battiato, S. (2020). Survey on visual sentiment analysis. IET Image Processing, 14(8), 1440-1456.
19. Soukupova, T., & Cech, J. (2016, February). Eye blink detection using facial landmarks. In 21st computer vision winter workshop, Rimske Toplice, Slovenia.