A Comparison of Face Verification with Facial Landmarks and Deep Features

Giuseppe Amato*, Fabrizio Falchi*, Claudio Gennaro* and Claudio Vairo*
*Institute of Information Science and Technologies of the National Research Council of Italy (ISTI-CNR)
via G. Moruzzi 1, 56124 Pisa, Italy
Email: {giuseppe.amato, fabrizio.falchi, claudio.gennaro, claudio.vairo}@isti.cnr.it

***Abstract*—Face verification is a key task in many application fields, such as security and surveillance. Several approaches and methodologies are currently used to try to determine if two faces belong to the same person. Among these, facial landmarks are very important in forensics, since the distance between some characteristic points of a face can be used as an objective measure in court during trials. However, the accuracy of the approaches based on facial landmarks in verifying whether a face belongs to a given person or not is often not quite good. Recently, deep learning approaches have been proposed to address the face verification problem, with very good results. In this paper, we compare the accuracy of facial landmarks and deep learning approaches in performing the face verification task. Our experiments, conducted on a real case scenario, show that the deep learning approach greatly outperforms in accuracy the facial landmarks approach.**

Keywords–Face Verification; Facial Landmarks; Deep Learning; Surveillance; Security.

I. Introduction

Face verification is getting higher importance recently. Face verification consists in determining if two faces in two different images belong to the same person or not. Face recognition, on the other hand, aims at assigning an identity to the person the faces belong to. In this paper, we are interested in the face verification problem.

To address the face verification problem, several approaches and techniques have been proposed. Some approaches are based on local features of the images, such as Local Binary Pattern (LBP) [1]. Some other approaches are based on detecting the facial landmarks from the detected face and on measuring the distance between some of these landmarks. Recently, Deep Learning approach and Convolutional Neural Networks (CNNs) have been proposed to address the face verification problem, such as [2]. Facial landmarks are particularly useful when forensics cases have to be discussed in court since they provide objective measures that can be presented to discuss face verification. However, as we will show in the paper, face verification with distances of automatically extracted facial landmarks, is outperformed by methods based on Deep Learning. Facial landmarks should be used after verification is executed using Deep Learning approaches, to provide objective motivation to the decision.

In this paper, we compare the results of performing the face verification with facial landmarks and a Deep Learning based approach. We validated our comparison by analyzing some videos taken in a real-scenario by surveillance cameras placed in the Instytut Ekspertyz Sdowych in Krakow [3]. To this purpose, we used the Labeled Faces in the Wild (LFW) dataset [4] as confusion dataset. In particular, we used the faces detected in these videos as queries to perform a Nearest Neighbor (NN) search with a joined dataset comprising both LFW and the test set videos, in order to classify the persons according to their face similarity.

The rest of the paper is organized as follows: Section II gives a brief overview of the current approaches to the face verification problem. In Section II-A, we describe the features obtained from the facial landmarks that we analyze and compare in this work. In Section II-B, we present the deep feature that we compare to the facial landmarks features. Section III presents an analysis on some of the facial landmarks features and the experiments on the accuracy of all the features considered. Finally, Section IV concludes this work.

II. Face Verification

The use of face information to verify the identity of a person is a research area experiencing rapid development, thanks to recent advances in deep learning. This approach falls under the umbrella of the more general identity verification problem [5]. Among the various types of facial information that can be used a fairly obvious one is that coming from the facial landmarks [6]–[9]. Deep Features learned from convolutional networks have shown impressive performance in classification and recognition problems. For instance, 99.77% accuracy of LFW under 6.000 pair evaluation protocol has been achieved by Liu at al. [10] and 99.33% by Schroff et al. of Google [11]. As in our proposed approach, approximate nearest neighbor search methods can be used to improve scalability and works very well as a lazy learning method [12], [13] and also a full-text search engine [14].

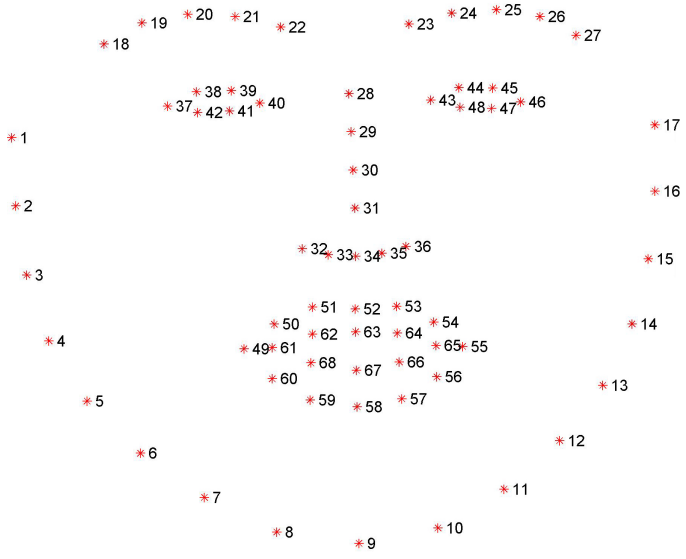
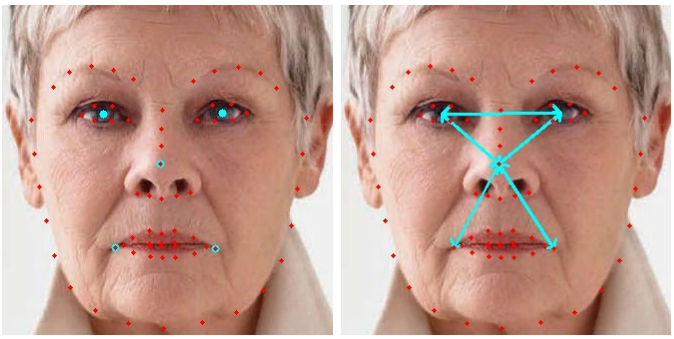


Figure 1. 68 facial landmarks.



(a) Selected 5 nodal points. (b) 5 nodal points distances.

Figure 2. Nodal points and distances used to build the 5-points features.

A. Facial Landmarks Features

Facial landmarks are key points along the shape of the detected face and they can be used as face features to perform several tasks like improve face recognition, align facial images, distinguish males and females, estimate the head pose, and so on.

Key points from landmarks are rarely used as a representation of face verification tasks, typically *facial nodal points* are used instead. As nodal points, we can either use directly some of the facial landmarks or we can compute some new points starting from the facial landmarks. For example, the eyes, the nose, and the mouth are very representative parts of a person's face, so points relative to these parts of the face can be relevant to represent that face. In particular, for example, for the eyes, we can use the centroid of the eye instead of using the facial landmarks that constitute the contour of the eye.

In order to perform the face detection and to extract the facial landmarks from an image, we used the dlib library [15]. In particular, the face detector is made using the Histogram of Oriented Gradients (HOG) feature combined with a linear classifier, an image pyramid, and sliding window detection scheme. The facial landmark detector is an implementation of the approach presented by Kazemi et al. in [16]. It returns an array of 68 points in form of (x,y) coordinates that map to facial structures of the face, as shown in Figure 1. The computational time for extracting the facial landmarks from the image reported in Figure 2 on a MacBook Pro 2013 with an i7 Intel Core 2.5 GHz is about 70 ms.

The distances between nodal points and facial landmarks can be used to build a feature of the face that can be compared with other faces features. In particular, we computed three features based on the distances between nodal points and facial landmarks: the *5-points* feature, the *68-points* feature and the *Pairs* feature. All the distances used to compute these features are normalized to the size of the bounding box of the face. In particular, each distance is divided by the diagonal of the bounding box.

1) 5-points feature: In order to build the 5-points feature, we used five specific nodal points: the centroids of the two eyes, the center of the nose, and the sides of the mouth. The centroids of the two eyes are computed from the six facial landmarks for each eye returned by the dlib library. For the nodal points of the nose and of the mouth, instead, we used directly some of the facial landmarks, respectively the landmark #31 for the nose and the landmarks #49 and #55 for the sides of the mouth (see Figure 2(a)). We used these nodal points to compute the following 5 distances (see Figure 2(b)):

- left eye centroid - right eye centroid
- left eye centroid - nose
- right eye centroid - nose
- nose - left mouth
- nose - right mouth

This produces a 5-dimensional float vector that we used as 5-point feature of the face.

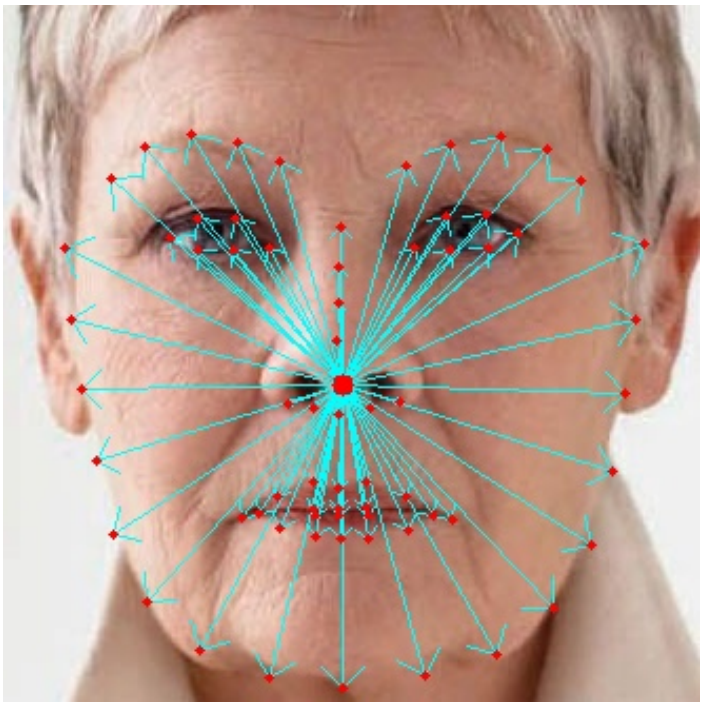


Figure 3. Distances from the centroid of the face to all 68 facial landmarks, used to build the 68-points features.

2) 68-points feature: For the 68-points feature, we computed the centroid of all the 68 facial landmarks returned by the dlib library and we computed the distance between this point and all the 68 facial landmarks (see Figure 3). This produces a 68-dimensional float vector that we used as 68-feature of the face.

3) Pairs feature: The pairs feature is obtained by computing the distance of all unique pairs of points taken from the 68 facial landmarks computed on the input face, as suggested in [9]. This produces a vector of 2.278 float distances that we used as Pairs feature of the face.

B. Deep Features

Deep Learning [17] is a branch of machine learning that uses lots of labeled data to teach computers how to perform perceptive tasks like vision or hearing, with a near-human level of accuracy. In particular, in computer vision tasks, CNNs are exploited to learn features from labeled data. A CNN learns a hierarchy of features, starting from low level (pixels), to high level (classes). The learned feature is thus optimized for the task and there is no need to handcraft it. Deep

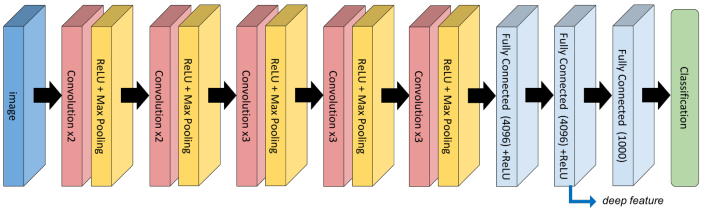


Figure 4. Structure of the VGG-Face CNN used to extract the deep features.

(a) Sample from P1-video2.

(b) Sample from P1-video3.

Figure 5. Samples of videos for Person1.

(a) Sample from P2-video1.

(b) Sample from P2-video2.

(c) Sample from P2-video3.

Figure 6. Samples of videos for Person2.

Learning approaches give very good results in executing tasks like image classification, object detection and recognition, scene understanding, natural language processing , traffic sign recognition, cancer cell detection and so on [18]–[21].

However, CNNs are good not only for classification purposes. In fact, as said before, each convolutional layer of a CNN learns a feature of the input image. In particular, the output of one of the bottom layers before the output of the network, is, in fact, a high-level representation of the input image, that can be used as a feature for that image. We call *deep feature* this representation of the image. This feature can be compared to other deep features computed on other faces, and close deep features vectors mean that the input faces are semantically similar. Therefore, if their distance is below a given threshold, we can conclude that the two faces belong to the same person.

For this work, we used the VGG-Face network [2] that is a CNN composed of 16 layers, 13 of which are convolutional. We took the output of the fully connected layer 7 (FC7) as deep feature, that is a vector of 4.096 floats (see Figure 4). The computational time for extracting the deep feature from the image reported in Figure 2 on a MacBook Pro 2013 with an i7 Intel Core 2.5 GHz is about 300 ms, that is four times the time needed to extract the facial landmarks from the same image.

III. Experimental Evaluation

In this section, we describe the experiments performed to compare the accuracy of the different features described in Sections II-A and II-B in performing the face verification task. We first describe the test set used in our experiments, that is constituted by six videos acquired by surveillance cameras deployed in some of the corridors of the Instytut Ekspertyz Sdowych in Krakow and by the famous face dataset LFW, that we used as confusion set. We then present an analysis of the distances computed over the facial landmarks and, finally, we report some accuracy results obtained by our experiments on the considered features.

A. Test set

We used six videos as test set, provided by the EU Framework Programme Horizon 2020 COST Association COST Action CA16101 [22]. These videos are taken from three different surveillance cameras deployed in the Instytut Ekspertyz Sdowych in Krakow and they capture two different persons (we call them ”Person1” and ”Person2”). Each of them is recorded in all the environments where the cameras are installed. So, we have three videos for Person1 and three videos for Person2. For each video, we analyzed each frame independently. In particular, for each frame, we executed the face detection phase, and for the frames where a face has been detected, we executed the facial landmarks detection algorithm. We then computed the 5-points, 68-points and Pairs features, by exploiting the 68 detected landmarks.

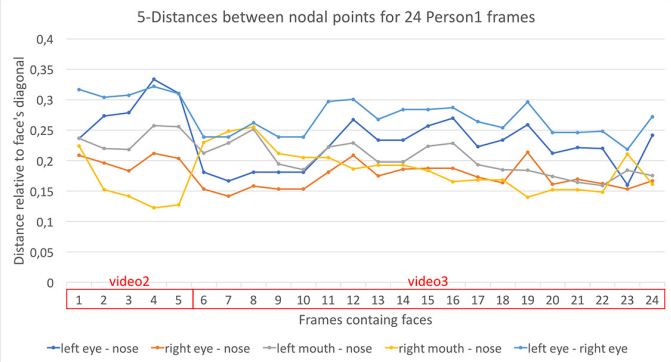
The videos used in our experiments are very challenging because the resolution is low (768x576), and the person is in the foreground of the scene. We have obtained 59 total frames containing faces in all the six videos, that are composed as follows:

- Person1 (P1):
  - video1: 0 faces detected (the face was never recorded clearly in the video);
  - video2: 5 faces detected;
  - video3: 19 faces detected;
- Person2 (P2):
  - video1: 5 faces detected;
  - video2: 16 faces detected;
  - video3: 14 faces detected;

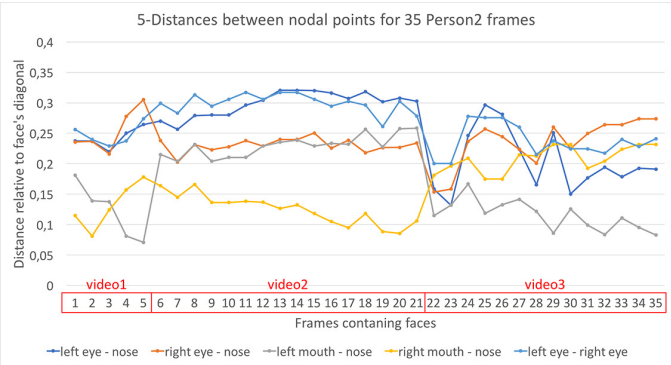
Figures 5 and 6 show some samples of the Person1 videos and Person2 videos, respectively.

B. Facial landmarks distances measurements

In order to understand if there is a way to better exploit the distance between facial landmark points, we have performed an analysis and computed some measurements on the distances between 5 nodal points and on the distances between the 68 facial landmarks and the centroid, in different frames collected by the sample videos that we used as test set.

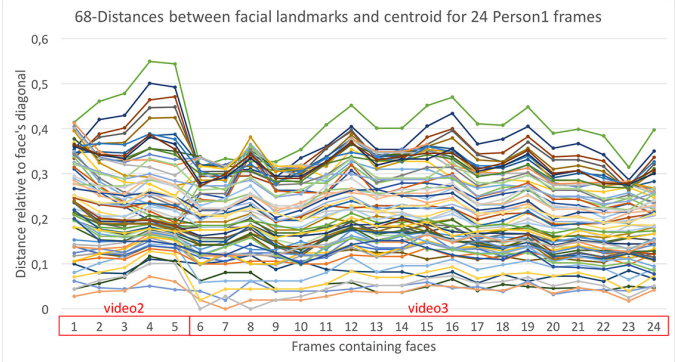


(a) 5-points features for Person1 videos.

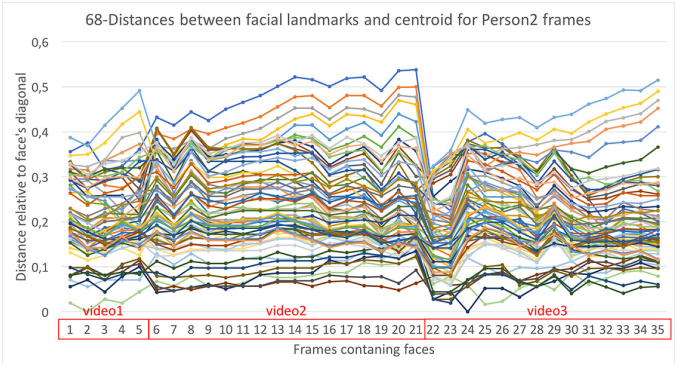


(b) 5-points features for Person2 videos.

Figure 7. Distances between the 5 nodal points in different frames of Person1 (a) and Person2 (b) videos.



(a) 68-point features for Person1 videos.



(b) 68-points features for Person2 videos.

Figure 8. Distances between the 68 facial landmarks and the face centroid in different frames of Person1 (a) and Person2 (b) videos.

Figures 7 and 8 show, respectively, the trend of the components of the 5-points and 68-points features in different frames of the videos, for both persons. Please, recall that Person1 face has been detected in just two videos, while Person2 face has been detected in all three videos. It is possible to notice that, for frames of the same video, the lines of the distances are quite regular, while they have a great difference when moving to another video. This shows that, while a person is seen by the same camera, with the same angle of view, it is possible to use the distance of facial landmarks to recognize a person by its face with good accuracy.

We also computed the average and the variance of the distances between nodal points and facial landmarks reported in Figure 9. In particular, Figure 9(a) reports the average and the variance of the distances between the 5 nodal points and Figure 9(b) reports the average and variance of the distances between the centroid of the face and the 68 facial landmarks. In both cases, the average and the variance are computed on the distance of the same pair of points in all the different frames of Person1 and Person2 videos. The figure shows that the variance is very small in almost every pair of points, and also that the average value of the two persons is quite different in four of five pairs of the nodal points (Figure 9(a)) and in lots of 68 facial landmarks (Figure 9(b)). This means that, by analyzing consecutive frames of a video, when this is feasible, it is possible to increase the possibility to recognize a certain person by using the distance of the facial landmarks.

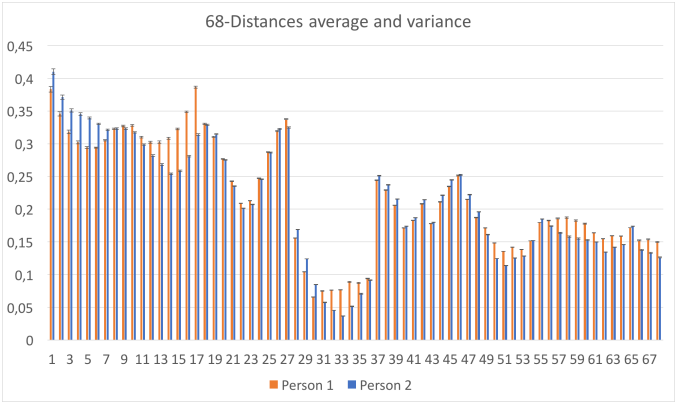
C. *Classification Accuracy*

We performed some experiments to compare the accuracy in performing the face verification task by using the four different features described above. To this purpose, the faces extracted from the videos were merged with LFW, that has been used as distractor.

LFW is a very famous face dataset, which contains around 13 thousand faces and 5.750 different identities. All images in LFW are 250x250 pixels and the face is aligned to be in the center of the image. However, there is a lot of background in the images, sometimes capturing also other people faces. This could lead to multiple face detection. Therefore, we cropped each image in the LFW dataset to the size of 150x150 pixels, by keeping the same center, in order to cut the background and avoid multiple face detection. In this case also, we performed the face detection and we computed the facial landmark points by using the dlib library (Figure 10 shows some examples of LFW faces with facial landmarks highlighted). We merged the LFW dataset with the 59 faces that we detected in the test videos and we created a unified dataset. We then extracted the four different features (5-points, 68-points, Pairs and deep features), from all the faces in the new dataset.

We used each of the faces detected in the test set videos as a query for a NN search in the unified dataset. We used the Euclidean distance as dissimilarity measure between features and sorted the entire dataset according to this distance with the given query, from the nearest to the farthest. We discarded the first result of each query since it is the query itself.

(a) 5- points feature average and variance for Person1 videos.



(b) 68-points feature average and variance for Person2 videos.

Figure 9. Average and variance of the 5 and 68 distances for Person1 (a) and Person2 (b).

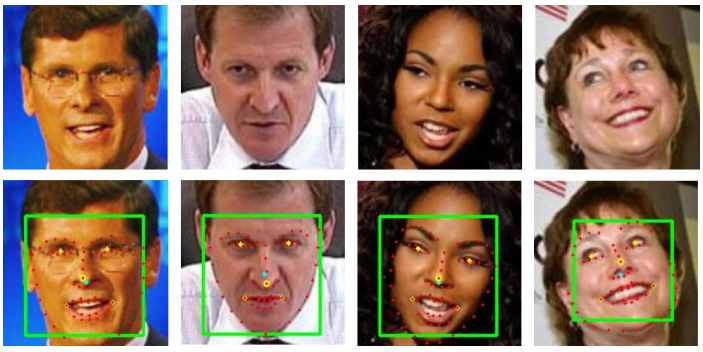


Figure 10. Some examples from LFW dataset and the corresponding detected faces with facial landmarks.

Figure 11 reports some query examples with the Top5 results, for all the features analyzed. For each feature, we report the best and the worst result, in which the biggest number of, respectively, correct and wrong matches in the first five results is obtained. The best result of 5-points feature only got three correct matches in the Top5 results, while all the other features got all correct matches in the Top5 results. The worst result is the same for all the facial landmarks features, that is no correct match in the Top5 results. On the other hand, the deep feature worst result only has one wrong match, that is ranked in the last of the Top5 results.

The different size of the faces detected in the videos is due to the different size of the bounding box of the face computed by the face detector library. This is caused by the different position of the person in the scene with respect to the camera; a bigger face means that the person is closer to the camera.

TABLE I. MEAN AVERAGE PRECISION COMPUTED FOR ALL THE FOUR DIFFERENT FEATURES.

Feature	*mAP*
5-points feature	0.03
68-points feature	0.06
Pairs feature	0.07
Deep feature	0.81

We compared all the four different features by computing the mean Average Precision (mAP) on the results of the queries, so we measured how well the results are ordered according to the query. In particular, for each query, we sum the number of correct results, weighted by their position in the result set, and we divide this value by all the correct elements in the dataset. We then average the precision of all queries, thus obtaining the mean Average Precision for each feature.

The results are reported in Table I. They show that the 68-points feature is two times better than the 5-points feature, and the Pairs feature slightly improves the 68-points feature result. However, the deep feature is more than one order of magnitude better than all the features based on the facial landmarks.

TABLE II. TOP1 AND TOP5 ACCURACY COMPUTED FOR ALL THE FOUR DIFFERENT FEATURES.

Feature	*Top1*	*Top5*
5-points feature	24%	47%
68-points feature	51%	76%
Pairs feature	64%	78%
Deep feature	97%	98%

We also computed the Top1 and Top5 accuracy for all the features considered. The Top1 accuracy counts the percentage of queries in which the first person of the result set is the same person of the corresponding query. The Top5 accuracy considers the first five persons of the result set to check if the correct one is present. Table II shows that 5-points feature works very bad in this scenario with small and low-resolution faces with a Top1 accuracy of only 24% and a Top5 accuracy of 47%. The 68-points feature and the Pairs features, improve the Top1 accuracy of more than twice with respect to the 5-points feature, and up to 78% in case of the Top5 accuracy. Also in this case, however, the deep feature works much better obtaining a 97% Top1 accuracy and a 98% Top5 accuracy.

The facial landmarks have indeed the property of being an accepted proof in trials, and they can be used to classify people in some conditions and with a certain accuracy; they are also faster to be computed with respect to the deep features. However, the deep feature shows much better performance, especially in challenging scenarios with low-resolution faces.

IV. CONCLUSION

In this paper, we presented a comparison between facial landmarks and deep learning approaches in performing the

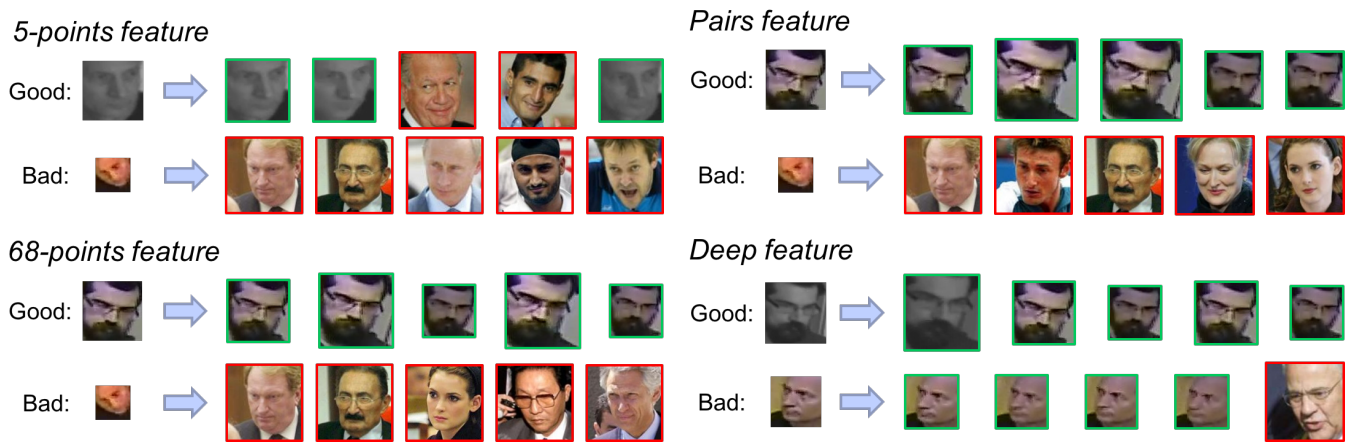


Figure 11. Query examples for the all the kinds of features, with Top5 results. For each feature, the best and the worst results are reported.

face verification task. Facial landmarks are very important in forensics because they can be used as objective proof in trials. We performed our experiments on videos taken in a real scenario and by exploiting the widely used face dataset LFW. Results show that the accuracy of the deep features in verifying whether a face belongs to a given person is much greater than the one of facial landmarks based approach. On the other hand, the deep learning results cannot be used as proof in court. We think, however, that deep features approach should help the forensics process along with facial landmarks. In particular, the latter should be used after the face verification has been executed with deep features, in order to provide an objective measure for the decision.

ACKNOWLEDGMENTS

This work has been partly funded by the "Renewed Energy" project of the DIITET Department of CNR and by the EU Framework Programme Horizon 2020 COST Association COST Action CA16101. Special thanks to Prof. Dariusz Zuba and the Instytut Ekspertyz Sdowych in Krakow for the videos used as test set.

REFERENCES

[1] T. Ahonen, A. Hadid, and M. Pietikainen, "Face description with local binary patterns: Application to face recognition," IEEE transactions on pattern analysis and machine intelligence, vol. 28, no. 12, 2006, pp. 2037–2041.

[2] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in British Machine Vision Conference, 2015.

[3] "Instytut ekspertyz sdowych - krakow," http://ies.krakow.pl/, accessed: 2018-04-13.

[4] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," Technical Report 07-49, University of Massachusetts, Amherst, Tech. Rep., 2007.

[5] P. Verlinde, G. Chollet, and M. Acheroy, "Multi-modal identity verification using expert fusion," Information Fusion, vol. 1, no. 1, 2000, pp. 17–33.

[6] Y. Sun, Y. Chen, X. Wang, and X. Tang, "Deep learning face representation by joint identification-verification," in Advances in neural information processing systems, 2014, pp. 1988–1996.

[7] C. Sanderson, M. T. Harandi, Y. Wong, and B. C. Lovell, "Combined learning of salient local descriptors and distance metrics for image set face verification," in Advanced Video and Signal-Based Surveillance (AVSS), 2012 IEEE Ninth International Conference on. IEEE, 2012, pp. 294–299.

[8] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar, "Attribute and simile classifiers for face verification," in Computer Vision, 2009 IEEE 12th International Conference on. IEEE, 2009, pp. 365–372.

[9] A. G. Rassadin, A. S. Gruzdev, and A. V. Savchenko, "Group-level emotion recognition using transfer learning from face identification," arXiv preprint arXiv:1709.01688, 2017.

[10] J. Liu, Y. Deng, T. Bai, Z. Wei, and C. Huang, "Targeting ultimate accuracy: Face recognition via deep embedding," arXiv preprint arXiv:1506.07310, 2015.

[11] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 815–823.

[12] J. Park, K. Lee, and K. Kang, "Arrhythmia detection from heartbeat using k-nearest neighbor classifier," in Bioinformatics and Biomedicine (BIBM), 2013 IEEE International Conference on. IEEE, 2013, pp. 15–22.

[13] D. Wang, C. Otto, and A. K. Jain, "Face search at scale," IEEE transactions on pattern analysis and machine intelligence, vol. 39, no. 6, 2017, pp. 1122–1136.

[14] G. Amato, F. Carrara, F. Falchi, and C. Gennaro, "Efficient indexing of regional maximum activations of convolutions using full-text search engines," in Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval. ACM, 2017, pp. 420–423.

[15] "Dlib library," http://dlib.net/, accessed: 2018-04-13.

[16] V. Kazemi and J. Sullivan, "One millisecond face alignment with an ensemble of regression trees," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 1867–1874.

[17] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, 5 2015, pp. 436–444.

[18] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in Advances in neural information processing systems, 2012, pp. 1097–1105.

[19] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Region-based convolutional networks for accurate object detection and segmentation," IEEE transactions on pattern analysis and machine intelligence, vol. 38, no. 1, 2016, pp. 142–158.

[20] G. Amato, F. Falchi, and L. Vadicamo, "Visual recognition of ancient inscriptions using convolutional neural network and fisher vector," Journal on Computing and Cultural Heritage (JOCCH), vol. 9, no. 4, 2016, p. 21.

[21] G. Amato, F. Carrara, F. Falchi, C. Gennaro, C. Meghini, and C. Vairo, "Deep learning for decentralized parking lot occupancy detection," Expert Systems with Applications, vol. 72, 2017, pp. 327–334.

[22] "Eu framework programme horizon 2020 cost action ca16101," http://www.cost.eu/COST_Actions/ca/CA16101, accessed: 2018-04-13.