PARTIAL FACE DETECTION FOR CONTINUOUS AUTHENTICATION

Upal Mahbub[⋆]   *Vishal M. Patel*[†]   *Deepak Chandra*[§]   *Brandon Barbello*[§]   *Rama Chellappa*[⋆]

[⋆]Department of Electrical and Computer Engineering and the Center for Automation Research,
UMIACS, University of Maryland, College Park, MD 20742
[†]Rutgers, The State University of New Jersey, 723 CoRE, 94 Brett Rd, Piscataway, NJ 08854
[§]Google Inc., 1600 Amphitheatre Parkway, Mountain View, CA 94043

umahbub@umiacs.umd.edu, vishal.m.patel@rutgers.edu,
dchandra@google.com, bbarbello@google.com, rama@umiacs.umd.edu

ABSTRACT

In this paper, a part-based technique for real time detection of users' faces on mobile devices is proposed. This method is specifically designed for detecting partially cropped and occluded faces captured using a smartphone's front-facing camera for continuous authentication. The key idea is to detect facial segments in the frame and cluster the results to obtain the region which is most likely to contain a face. Extensive experimentation on a mobile dataset of 50 users shows that our method performs better than many state-of-the-art face detection methods in terms of accuracy and processing speed.

Index Terms— Active authentication, smartphone front-camera images processing, facial segments, partial face detection

1. INTRODUCTION

Face detection, being one of the most fundamental problems in computer vision, plays an important role in the performance of various face-based applications ranging from face identification and verification to clustering, tagging and retrieval [20]. There has been substantial progress in the development of efficient face detection techniques and many such techniques are now available in commercial products such as digital cameras, smartphones and social networking websites [8]. Both in academia and industry, the current trend of developing new face detectors is primarily centered around detecting faces in unconstrained environments with huge variations of pose and illumination. This has led to the development of a number of recent face detectors that try to outperform each other on the Labeled Faces in the Wild (LFW) [7], Annotated Facial Landmarks in the Wild (AFLW) [9] and other such datasets. Another focus of current face detection research is in developing fast face detectors for real time implementation on devices such as smartphones, tablets etc.

In today's world, mobile devices are being used not only for verbal communication but also for accessing bank accounts and performing transactions, managing user profiles, accessing e-mail accounts, etc. With increasing usage, there is a growing concern about ensuring the security of users' personal information on these devices. Going beyond passwords and fingerprints, concepts have emerged for actively verifying users by analyzing faces from the front-facing camera, the swipe and keystroke patterns from the touchscreen and

Fig. 1: Sample frames from the AA-01-FD mobile face dataset where one can clearly see the presence of partial faces.

motions patterns from inertial sensors, among others [14], [21], [2], [13]. Face-based authentication systems rely on accurate detection of faces in their first step and successful verification in the next. Several research works have recently been published on face-based continuous user authentication techniques for smartphones, using representation-based or attribute-based approaches [4], [13], [16]. However, in all of these methods the faces and landmarks are assumed to be available by performing face detection within milliseconds.

Viola and Jones's [19] ground-breaking method of face detection popularized Adaboost cascade structures and simple feature extraction techniques for realtime face detection. Hadid *et al.* [5] evaluated the feasibility of face and eye detection on cell phones using Adaboost cascade classifiers with Haar-like and local binary pattern (LBP) features, as well as a skin color based face detector. On a Nokia N90 mobile phone that has an ARM9 220 MHz processor and a built-in memory of 31 MB, their work reported that the Haar+Adaboost method can detect faces in 0.5 seconds from 320×240 pixel images. This approach, however, is not effective when wide variations in pose and illumination are present. Some variations of this method are available in the literature such as [6], [10] but most of them are computationally very expensive. Zhu *et al.* [22] proposed a method that uses a deformable part model where a face is defined as a collection of parts at each facial landmark, and the parts are then trained side-by-side with the face using a spring-like constraint. The method uses a mixture of tree-structured models resilient to viewpoint changes. Marín-Jiménez *et al.* [12] also followed a similar approach for 'Looking At Each Other' head detection. Whereas, Shen *et al.* [18] proposed an exemplar-based face detector that exploits advanced image retrieval techniques to avoid an expensive sliding window search. Apart from academic research, many face detectors such as Google Picasa, PittPatt and Face++ have been developed commercially as face detection has gotten more and
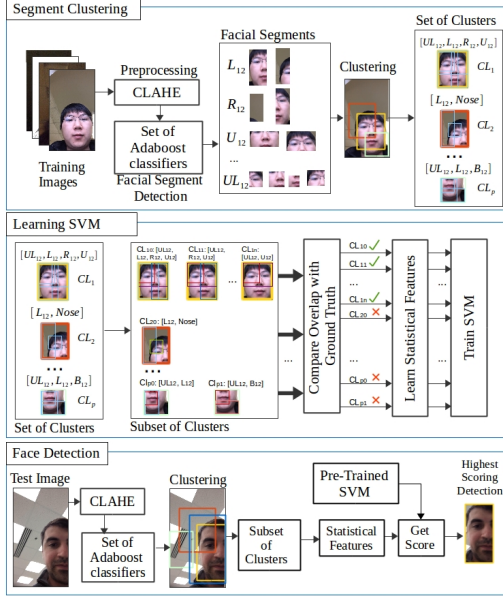
Fig. 2: The block diagram of the overall system.

Fig. 3: All of the 14 facial segments.

more attention over time. However, most of the state-of-the-art techniques are not designed to detect partially visible or cropped faces (see Figure 1). Hence, their recall rate is usually very low when operating at high precision. Furthermore, many of these methods require long computation times because of algorithmic complexity and are thus not suitable for real time implementation.

In this paper, we introduce a face detection scheme for detecting faces from images captured by the front-facing cameras of mobile devices used for continuous authentication. This paper makes the following contributions: (1) A method for realtime face detection on a smartphone is proposed based on facial segments that can detect cropped and partial faces of users. (2) A simple yet effective technique of clustering the facial segments is proposed. (3) Through extensive experimentation, it is shown that the proposed method can achieve superior performance over some of the state-of-the-art methods in terms of accuracy and processing time.

In the next section the proposed method is described in detail followed by a brief summary of the dataset. Then, the experimental setup, and results are explained. Finally, a summary of the work with future directions is provided in the section on conclusion.

2. FACIAL SEGMENT BASED FACE DETECTOR (FSFD)

The system block diagram for the proposed face detector is shown in Fig. 2. The process is divided into three phases. In the segment clustering phase, facial segments are logically clustered to estimate face regions from the training images. In the Support Vector Machine (SVM) learning phase, a linear SVM is trained with statistical features obtained from face proposals that are derived from the estimated faces. Finally, in the face detection phase, statistical features are obtained in a similar manner for each face proposal from test images and confidence scores are obtained for each proposal using the SVM classifier. The proposal with the highest confidence score higher than a certain threshold θ is considered to be a face. Because of its simple architecture, the method can be implemented in real-time. In addition, it is fairly accurate and provides a good recall rate since it can efficiently detect partially visible faces. Thus, the FSFD method is suitable for real time face verification tasks on mobile de-
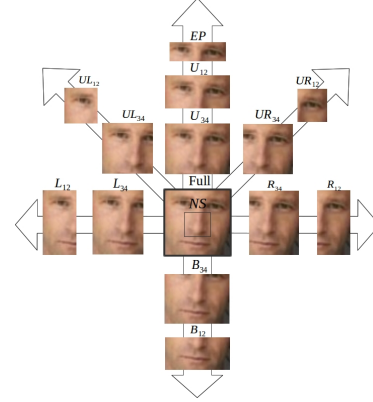
vices. The three main steps of the detector are discussed in detail in the following subsections.

2.1. Segment Clustering

As shown in Fig. 2, in the segment clustering phase, contrast limited adaptive histogram equalization (CLAHE) is performed first on each of the training images to reduce the impact of illumination on facial segment detectors. Each training image is then passed through a set of facial segment detectors. A total of 14 facial segment detectors were trained using facials segments obtained from the cropped and aligned Labeled Faces in the Wild (LFW) dataset [7], [17] and negative examples were produced from the Pascal VOC dataset [3]. These detectors are basically adaboost cascade classifiers trained using a local binary pattern (LBP) representation of the images for better feature representation and faster training [11], [1]. The facial segments for which the classifiers were trained are Eye-pair (EP), Upper-left-half of face (UL_{12}), Upper-half of face (U_{12}), Upper-right-half of face (UR_{12}), Upper-left-three-fourth of face (UL_{34}), Upper-three-fourth of face (U_{34}), Upper-right-three-fourth of face (UR_{34}), Left-half of face (L_{12}), Left-three-fourth of face (L_{34}), Nose (NS), Right-three-fourth of face (R_{34}), Right-half of face (R_{12}), Bottom-three-fourth of face (B_{34}) and Bottom-half of face (B_{12}). An example of all the 14 facial segments of a full face from the LFW dataset is shown in Fig. 3. Each detector may return one or more facial segments in a frame. For example, the segment clustering phase of Fig. 2, L_{12} returned two detection results while UL_{12} returned four detection results. For each facial segment, the bounding box of the full face is estimated according to the ideal position of that segment relative to the whole face. For example, if the top left and bottom right corners of the bounding box obtained for segment L_{12} are (x_1^{L12}, y_1^{L12}) and (x_2^{L12}, y_2^{L12}), respectively, then those for the estimated full face are (x_1^{L12}, y_1^{L12}) and $(min(w_{img}, x_2^{L12} + (x_2^{L12} - x_1^{L12})), y_2^{L12})$, where w_{img} is the width of the image. The estimated face center from this segment is $(x_2^{L12}, y_1^{L12} + (y_2^{L12} - y_1^{L12})/2)$. For each estimated face center p, a cluster of segments CL_p that depicts the full face is formed where (a) the number of segments in that cluster is above a threshold c, and (b) the other segments of that cluster have estimated face centers within a certain radius r pixels from the center.

2.2. Learning SVM

In the learning phase, the first ζ subsets of the total set of facial segments from each cluster are regarded as proposed faces. Assuming that m facial segments satisfy the condition for clustering around

the k-th segment, there are $m + 1$ segments in that cluster where $m + 1 \geq c$. For example, if the 4 segments U_{12}, B_{12}, L_{34} and UR_{12} form a cluster around NS and $c = 2$, then the viable subsets are $[[NS, U_{12}], [NS, B_{12}], \ldots, , [NS, U_{12}, B_{12}, L_{34}, UR_{12}]]$. The total number of subset here is $\sum_{j=1}^{4} \binom{4}{j} = 15$ including the complete set. Keeping the k-th segment fixed, ζ random subsets are considered for face proposals. In this example, ζ varies from 1 to 14. For $m+1$ segments, the number of subsets is in the order of 2^{m+1}, which introduces huge computation and memory requirements. Hence, the number of subset is limited to $\zeta << 2^{m+1}$.

The bounding box of the face proposal is the smallest bounding box that encapsulates all the estimated faces from all the facial segments in that proposal. Intuitively, the greater the number of facial segments with better detection accuracy in a proposal, the higher the probability of that proposal being a face. Further, experimentally, it is found that some particular sets of facial segments are more likely to return a face than others, and some sets of segments provide more accurate bounding boxes with greater consistency than the others. The amount of overlap is expressed as the ratio of intersection areas to joined area $I(d_i, a_j)$, between the detected region d_i and annotated regions a_j, and is expressed as

$$I(d_i, a_j) = \frac{area(d_i) \cap area(a_j)}{area(d_i) \cup area(a_j)}. \tag{1}$$

If this ratio is above a certain threshold δ, then the detection result is considered to be correct.

A linear SVM classifier is trained on the proposed faces using the following statistical features.

1. Probability of a proposal S_{set} being a face $Pr^T(S_{set})$ and not being a face $Pr^F(S_{set})$ are defined as

$$Pr^T(S_{set}) = \frac{\sum |S_{set} \in P_F^T|}{|P_F^T|} \tag{2}$$

$$Pr^F(S_{set}) = \frac{\sum |S_{set} \in P_F^F|}{|P_F^F|} \tag{3}$$

where, $P_F^T = \{S | I(S, S_{GT}) \geq \delta; S \in P_{Tr}^F\}$, and $P_F^F = \{S | I(S, S_{GT}) < \delta; S \in P_{Tr}^F\}$. Here, P_{Tr}^F denotes the set of clusters of facial segments that the detector proposed as faces.

2. For each facial segment in a set P_{Set}^i where $i \in S_{set}$, the probabilities of P_{Set}^i being in a cluster depicting the true face $Pr^T(P_{set}^i)$ or a non-face $Pr^F(P_{set}^i)$ are calculated as

$$Pr^T(P_{set}^i) = \frac{\sum |P_{set}^i \in S_{set}; S_{set} \in P_F^T|}{|P_F^T|} \tag{4}$$

$$Pr^F(P_{set}^i) = \frac{\sum |P_{set}^i \in S_{set}; S_{set} \in P_F^F|}{|P_F^F|}. \tag{5}$$

Experimentally, it is found that the nose detector is the most accurate of all, while B_{12} is the least accurate.

If n segments are considered then the feature size is $2n + 2$ for each proposal. There are $2n$ values corresponding to the face and non-face probabilities of each of the n segment and the rest 2 values are the probabilities of the cluster being and not-being a face. Among the $2n$ values, only those corresponding to the segments present in the proposal are non-zero.

2.3. Face Detection

For each pre-processed test image, the proposed faces are obtained in a similar manner as the training faces. Thus, there are ζ face proposals from each face and feature vectors of size $2n + 2$ for each proposal. The SVM classifier returns a confidence score for each proposal. The proposal that has the highest confidence score above a threshold is chosen as the detected face.

3. DATASET

The performance of the proposed face detector is evaluated on the Active Authentication Dataset (AA-01) [21], [16]. The AA-01 Dataset contains the front-facing camera face video for 50 iphone users (43 male, 7 female) with three different ambient lighting conditions: well-lit, dimly-lit, and natural daylight. In each session, the users performed 5 different tasks: enrollment, scroll test, picture count, document reading and picture drag. In order to evaluate the face detector, face bounding boxes were annotated in a total of 8036 frames of the 50 users. This dataset, denoted as AA-01-FD, contains 1607 frames without faces and 6429 frames with faces. For training, 20% of these frames are randomly picked and the rest are used as test data. Some sample face images from the AA-01-FD Dataset are shown in Fig. 1.

4. EXPERIMENTAL RESULTS

The experimental results are presented in this section, demonstrating the effectiveness of the FSFD method over other state-of-the-art face detectors. In particular, experimental results on the AA-01-FD dataset are compared with a. the Viola-Jones face detector (VJ) [19], b. the deformable part-based model face detector (DPM) [22], and c. the Looking At Each Other (LAEO) head detector [12].

4.1. Evaulation Metrics and Experimental Setup

The results are evaluated in terms of F1-Score, true positive rate (TPR) at 1% false positive rate (FPR), recall at 99% precision, and processing time. The F1-Score, which is the harmonic mean of precision and recall, is defined as

$$F1 = \frac{2TP}{2TP + FP + FN} \tag{6}$$

where, TP, FP and FN are the numbers of true positive, false positive and false negative detections, respectively. Higher F1-Score employs better overall precision and recall. For the active authentication problem, the goal is to achieve the best recall at a very high precision. Hence, the value of recall achieved by a detector at 99% precision is considered another evaluation metric. Finally, for detectors that return a confidence score rather than a binary face/non-face decision, the TPR at 1% FPR is used as a metric for evaluation. Here, also, higher values of TPR represent a better detector.

Based on the relative importance of each segment, several combinations of segments are considered to determine the optimum number of segments to produce the best result. The basic combination of all 14 segments is labeled as $C0$ and the best performing combination is labeled $Cbest$. $Cbest$ consists of 9 segments - NS, EP, UL_{34}, UR_{34}, U_{12}, L_{34}, UL_{12}, R_{12}, L_{12}.

All the experiments in this paper were performed on an Intel Xeon(R) CPU E5506 operating at 2.13GHz with 5.8GB memory and a 64 bit UBUNTU 14.04 LTS operating system. No multi-threading was done to parallelize the operations of different facial segment detectors, hence, speed up is achievable [15]. The images, initially 1280×720 pixels, are downsampled by 4. It is experimentally verified that for all the methods this downsampling ensures best performance in quickest time. The minimum face size is set to be 64×64 pixels. For clustering facial segments, the value of c is
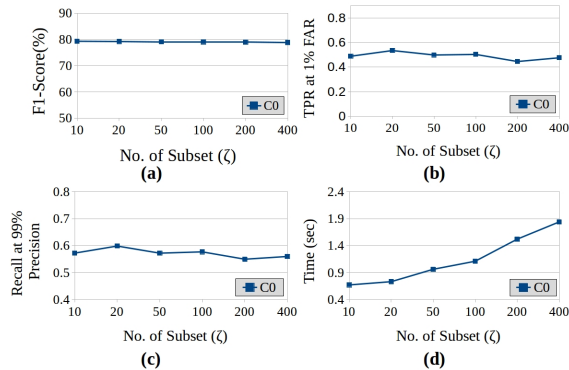
Fig. 4: Performance of $C0$ for different values of ζ - (a) F1-Score vs. ζ, (b) TPR at 1% FAR vs. ζ, (c) Recall at 99% Precision vs. ζ, and (d) Time vs. ζ.
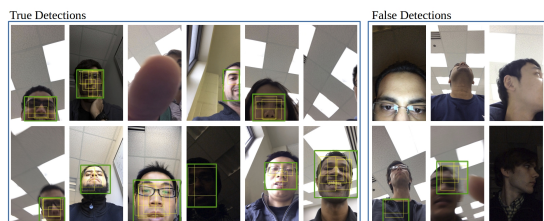
Fig. 5: Detection results for $C0$. Left - true positive and true negative detection results. Yellow boxes for facial segments, green boxes for final estimated face. Right - false positive and false negative results.

set to two and r is considered to be one-sixth of the half-diagonal of the estimated face of the center segment. A lower value of r may improve the precision while possibly decreasing the recall.

In order to determine the optimum value of ζ, the evaluation metrics and average processing time per image are plotted in Fig. 4 with increasing ζ for combination $C0$ on AA-01-FD. It can be inferred that the performance of the system does not change significantly, while the time consumption per image rapidly increases with ζ. Hence, a value of $\zeta = 20$ is chosen to ensure the best performance at the smallest time.

4.2. Results and Comparison

Some sample face detection results obtained by the proposed method for combination $C0(\zeta = 20, c = 2)$ are shown in Fig. 5. In this figure, the correct and the false detection results are shown. The yellow bounding boxes denote different facial segments and the green bounding box encapsulating all the yellow boxes is the final estimated face. It is apparent from this images that FSFD is robust enough to detect partial faces, such as the upper-half or left-half of a face, as well as full faces. Also, it can be seen that the operation of the proposed detector is mostly unaffected by illumination and pose variation. The images where the detector fails are mostly extreme situations where only a fragment of the face is visible, the face is critically posed, or the amount of motion blur is too much.

In Fig. 6 the F1 score of $C0$ and $Cbest$ are compared with DPM, LAEO and VJ face detection methods while varying the threshold η. It can be observed that the FSFD method outperforms all of the other methods at all values of η. $Cbest$ especially provides superior performance at a high percentage of overlap with the ground truth.

In Table 1, FSFD is compared with DPM in terms of TPR at 1% FPR and Recall at 99% Precision. For both metrics, $Cbest$ provides
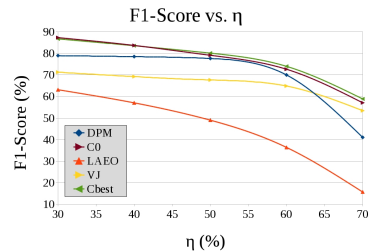
Fig. 6: Comparison of F1-Score for varying overalpping threshold η for $Cbest$ ($\zeta = 20, c = 2$) with state-of-the-art methods.

Table 1: Comparison between methods at 50% overlap

Method	TPR at 1% FPR	Recall at 99% Precision
DPM	0.4155	0.5957
LAEO	0.1612	-
FSFD $C0$ (ζ =20, c=2)	0.4940	0.5577
FSFD $Cbest$ (ζ =20, c=2)	**0.5635**	**0.6372**
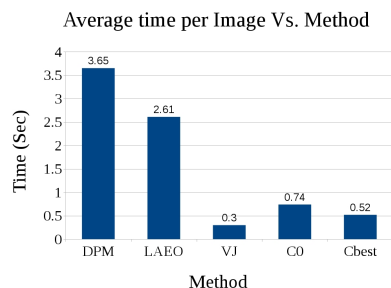
Fig. 7: Time per image for different methods.

significantly better performance than DPM as can be seen from the table. Note that LAEO cannot even reach 99% precision with any value of recall on the dataset. In Fig. 7 the average detection times per image for the four methods are compared. It can be seen that, though $Cbest$ utilizes 9 cascade classifiers, the time consumption is not 9-fold that of VJ which uses only one such classifier. This is because the scaling factor of VJ was set to a small value to ensure finer search in order to obtain the best results. Conversely, the scale factor of each cascade classifier in $Cbest$ was relatively big. Hence each classifier is individually weak but can produce much better outcomes when combined together. The LAEO and DPM detectors require 5 to 7 times more time than $Cbest$ and are not suitable for real time implementation for continuous authentication.

5. CONCLUSION AND FUTURE DIRECTIONS

In this paper, a novel facial segment-based face detection technique is proposed that is suitable for face-based continuous authentication on mobile devices due to its high recall at excellent precision. A total of 14 facial segment detectors have been trained and, an algorithm is introduced for clustering these segments to estimate a full or partially visible face. Through extensive experimentation, it is shown that the proposed method can achieve superior performance over state-of-the-art methods using fewer facial segment cascade classifiers, even as there still remains a lot of provision for speeding up the process. The future direction of this experiment is toward accurate landmark detection from overlapping facial segments and toward face authentication by fusing segment-wise verification scores.

6. REFERENCES

[1] Cascade classifier trining. *OpenCV 2.4.11.0 documentation - OpenCV User Guide*, 2000.

[2] D. Crouse, H. Han, D. Chandra, B. Barbello, and A. K. Jain. Continuous authentication of mobile user: Fusion of face image and inertial measurement unit data. In *International Conference on Biometrics (ICB)*, May 2015.

[3] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, June 2010.

[4] M. E. Fathy, V. M. Patel, and R. Chellappa. Face-based active authentication on mobile devices. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015.

[5] A. Hadid, J. Heikkila, O. Silven, and M. Pietikainen. Face and eye detection for person authentication in mobile phones. In *Distributed Smart Cameras, 2007. ICDSC '07. First ACM/IEEE International Conference on*, pages 101–108, Sept 2007.

[6] C. Huang, H. Ai, Y. Li, and S. Lao. High-performance rotation invariant multiview face detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(4):671–686, April 2007.

[7] 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, October 2007.

[8] V. Jain and E. Learned-Miller. Fddb: A benchmark for face detection in unconstrained settings. Technical Report UM-CS-2010-009, University of Massachusetts, Amherst, 2010.

[9] M. Kostinger, P. Wohlhart, P. Roth, and H. Bischof. Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization. In *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, pages 2144–2151, Nov 2011.

[10] S. Li, L. Zhu, Z. Zhang, A. Blake, H. Zhang, and H. Shum. Statistical learning of multi-view face detection. In A. Heyden, G. Sparr, M. Nielsen, and P. Johansen, editors, *Computer Vision ECCV 2002*, volume 2353 of *Lecture Notes in Computer Science*, pages 67–81. Springer Berlin Heidelberg, 2002.

[11] S. Liao, X. Zhu, Z. Lei, L. Zhang, and S. Li. Learning multiscale block local binary patterns for face recognition. In S.-W. Lee and S. Li, editors, *Advances in Biometrics*, volume 4642 of *Lecture Notes in Computer Science*, pages 828–837. Springer Berlin Heidelberg, 2007.

[12] M. Marin-Jimenez, A. Zisserman, M. Eichner, and V. Ferrari. Detecting people looking at each other in videos. *International Journal of Computer Vision*, 106(3):282–296, 2014.

[13] C. McCool, S. Marcel, A. Hadid, M. Pietikainen, P. Matejka, J. Cernocky, N. Poh, J. Kittler, A. Larcher, C. Levy, D. Matrouf, J.-F. Bonastre, P. Tresadern, and T. Cootes. Bi-modal person recognition on a mobile phone: using mobile phone data. In *IEEE ICME Workshop on Hot Topics in Mobile Multimedia*, July 2012.

[14] V. M. Patel, R. Chellappa, D. Chandra, and B. Barbello. Continuous user authentication on mobile devices. *IEEE Signal Processing Magazine*, 2016.

[15] K. Pulli, A. Baksheev, K. Kornyakov, and V. Eruhimov. Real-time computer vision with opencv. *Commun. ACM*, 55(6):61–69, June 2012.

[16] P. Samangouei, V. M. Patel, and R. Chellappa. Attribute-based continuous user authentication on mobile devices. In *International Conference on Biometrics Theory, Applications and Systems (BTAS), Arlington, VA*, Sept 2015.

[17] C. Sanderson and B. Lovell. Multi-region probabilistic histograms for robust and scalable identity inference. In M. Tistarelli and M. Nixon, editors, *Advances in Biometrics*, volume 5558 of *Lecture Notes in Computer Science*, pages 199–208. Springer Berlin Heidelberg, 2009.

[18] X. Shen, Z. Lin, J. Brandt, and Y. Wu. Detecting and aligning faces by image retrieval. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 3460–3467, June 2013.

[19] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–511–I–518 vol.1, 2001.

[20] B. Yang, J. Yan, Z. Lei, and S. Z. Li. Fine-grained evaluation on face detection in the wild. In *Automatic Face and Gesture Recognition (FG), 11th IEEE International Conference on*. IEEE, 2015.

[21] H. Zhang, V. Patel, M. Fathy, and R. Chellappa. Touch gesture-based active user authentication using dictionaries. In *Applications of Computer Vision (WACV), 2015 IEEE Winter Conference on*, pages 207–214, Jan 2015.

[22] X. Zhu and D. Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, CVPR '12, pages 2879–2886, Washington, DC, USA, 2012. IEEE Computer Society.