CHAPTER 10

Face Recognition

Andrey Gubenko and Abhishek Verma

CONTENTS

10.1 PUBLIC DATASETS

Since the face recognition problem has been here for a while, there are several public datasets available. In most cases, there is no need to adjust the general purposes datasets for mobile biometrics. Of course, some observation could be made regarding features more important for biometrics, such as environment, noise, different illumination, and rotation. Below we describe a few datasets.

10.1.1 PubFig: Public Figures Face Database

The PubFig dataset consists of 58,797 real-world images of 200 people collected from the Internet. Most of these images were taken in uncontrolled situations which makes a huge difference from other popular datasets.

10.1.2 Labeled Faces in the Wild

Labeled faces in the wild is a dataset of face images developed to test an algorithm in solving the unconstrained face recognition problem. The dataset consists of more than 13,000 images collected from the Internet. Each image was classified according to the name of person. A total of 1680 of these people have two or more distinct pictures in this dataset. Faces in these dataset were detected using the Viola–Jones face detector.

10.1.3 Columbia Gaze Dataset

Columbia gaze dataset contains 5880 images of 56 people taken with varying gaze directions and head poses. For each person, there are 5 head poses and 21 gaze directions per head pose. The dataset's subjects are ethnically diverse and 21 of the people wore glasses. This dataset was created to train a detector to sense eye contact, but it can be used for any other gaze estimation or tracking problems.

10.1.4 3D Mask Attack Database

The 3D mask attack database (3DMAD) is a biometric (face) spoofing database. It currently comprises 76,500 frames covering 17 people. These were recorded using Kinect for both real access and spoofing attacks. Each frame consists of a depth image, the corresponding RGB image and manually annotated eye positions (with respect to the RGB image). The data were collected in three different sessions from the subjects. Five videos of 300 frames were captured for each session. The recordings were done under controlled conditions and depict frontal view and neutral expression.

10.1.5 10k US Adult Faces Database

US adult faces dataset contains 10,168 natural face photographs and several measures for 2222 of the faces, including memorability scores, computer vision and psychology attributes, and landmark point annotations. The face pictures are stored in as a JPEGs format 72 pixels/in resolution and 256-pixel height.

10.2 RELEVANT WORK ON MOBILE FACE RECOGNITION

One of the most important requirements for mobile biometrics in comparison with classical approaches is that they have to work in real time. Due to the use cases, images should be captured and processed immediately when authentication is required.

CASE STUDY 1

Choi et al. [1] introduce the extraction method of local features using a random basis alongside with sequential neural network. An algorithm uses these extracted features to train the neural network incrementally, adjusting weights on each step. This paper also introduces a real-time training process which is essential for mobile biometrics. Secondary, the introduced model solves the problem of large variation pictures of the same faces, taken with different illumination and with occlusions.

There are two main distinctive features of this algorithm:

1. *Local random basis.* The algorithm generates a nonorthogonal local random basis which is robust to local deformation. Compared with a common basis which extracts orthogonal and global features, a nonorthogonal basis is more efficient in terms of computational requirements.
2. *Real-time incremental training.* The model is trained incrementally using a nontraining-based feature extractor with a sequential classifier. This method avoids retraining which significantly decreases computational time.

While most of the well-known training-based features extraction methods require a lot of training samples and hyper parameters tuning, random projection (RP) [2] was created as a dimensional reduction technique which can operate without a lot of training samples. Because there is no need in training samples, the basis generation approach doesn't require a lot of computational power which makes it perfect to use for mobile platforms. Besides this, applying of the local random basis allows the convolution of high-dimensional features to low-dimensional features and we decrease amount of memory required.

RP projects multidimensional data to low-dimensional subspaces using a set of randomly generated basis, in such a way that the projection matrix consists of entries with zero mean and a constant variance. This approach is based on the sparse RP method proposed by Achlioptas [3], when the matrix is projected onto low-dimensional spaces without normalization.

Another part of the proposed method is the multilayer feedforward neural network (MLP). This machine learning technique is widely used for solving different classification problems. If we think about MLP as a single hidden layer an output layer will be formed by neurons with linear activation functions. Such a model is called a single layer feedforward network (SLFN) [4]. SLFN can be used as a universal approximator with adjustable weights. Normally, a hidden neuron parameter can be randomly generated according to any continuous sampling distribution. In this case, hidden node parameters can be independent of the training data and by new incoming data the learning machine can be retained only by using additional samples via a recursive least-squares formulation. In one of the previous works, the authors propose to call that online sequential extreme learning machine (OS-ELM) [5]. OS-ELM provides great accuracy along with robustness, but is highly computational intensive for high-dimensional face images.

The authors introduce an architecture which uses both techniques together. First, local implementation of RP applies to image, which performs projection from high- to low-dimensional space. Local RP is more robust to local deformations and illumination conditions. The effect is achieved by using local techniques such as independent component analysis [6], local features analysis [7], and local nonnegative matrix factorization [8]. After this the online sequential ELM is applied to the projection. Because we have already decreased the number of dimensions, OS-ELM does not require a lot computational resources which makes it applicable for mobile platforms.

Experiments reveal that such an approach works fast on most of the datasets with acceptable good accuracy. Although experiments showed different training time on different datasets, it showed low constant time in most cases.

CASE STUDY 2

There is much implementation of face detection using dedicated hardware and coprocessors built in digital cameras. However, most of these approaches cannot be applied to mobile biometrics due to high computational or memory costs.

As it was mentioned before, it is critical for mobile biometrics to perform image recognition in real time. Observing existing face recognition algorithms shows that the key algorithm for face detection is the algorithm introduced by Viola and Jones [9]. This algorithm uses a cascaded Adaboost classifier, trained on a number of sub images. One of the requirements of the Adaboost is that the classifier has to be previously trained offline, due to the amount of time and computational resources. While algorithms based on the Viola approach give good accuracy, it has a limitation when there is rotation or occlusion. This can be solved by using different assembling models with pretraining. However, pretraining requires additional computational and storage resources and it is not possible to do it in real time.

Viola and Jones [10] present a hybrid solution for robust face detection which does not require any pretraining and image preprocessing. The authors introduce a hybrid solution which consists of the Viola algorithm alongside a color-based approach.

In the color-based part of the algorithm, skin color in Cr–Cb color space is represented by the Gaussian mixture model (GMM). Experimentally it was shown [11] that the optimal number of Gaussians is two, which gives the best accuracy–speed ratio.

The procedure for face detecting consists of two stages. In the first stage of the algorithm, the images convert into a binary image by applying skin color extraction using the GMM with 1's representing pixels of the image corresponding to skin color pixels and 0's representing nonskin color pixels. Then, by applying a subblock shape processing scheme to the binary image algorithm detects the face based on face size, aspect ratio, and probability.

Such a skin color detection approach increases robustness to the rotational variance as well as decreases computational requirements. The favorable difference of this algorithm to other similar algorithms is that it still works well when the surroundings have the same color as the skin.

Adding the Viola and Jones algorithm to this hybrid solution helps to avoid the limitations of illumination conditions and computation requirements which are incredibly important for real-time mobile face detection.

Since the Viola and Jones algorithm has already shown its robustness to lighting and brightness variations, the authors use it to detect frontal view faces. Then, the detected area is used to calibrate a previously introduced color-based model for illumination conditions. After calibration, the color-based algorithm can detect faces regardless of any brightness issues. Also, this hybrid approach does not require any additional data collection for classifier training.

The real-time implementation of the algorithm was tested using Texas Instruments OMAP3430 mobile platform. Platform consists of ARM Cortex-A8 processor, GPU, and C6400 DSP. For testing, the authors used combination of tests with different lighting such as florescent (average accuracy 94.4%), incandescent (91.8%), mixed (93%), and sunlight (92.3%). Experiments were performed under different image conditions, for example, rotation, covered eyes, and frontal faces. Also, experiments showed that average processing time was around 800 ms of the OMAP3430 mobile device.

CASE STUDY 3

As we know, most face recognition techniques rely on sophisticated machine learning algorithms which can be computationally intensive. Due to the limited resources of mobile devices, client–server architecture becomes more and more popular, because all data processing can be completed on the more powerful server side.

Kremic et al. [12] proposed a new authentication model, which reduces risks with cell phone authentication. At the very beginning of wireless technology, most mobile authentication systems were an adaptation of the classic client–server model. However, due to the specifics of mobile biometrics this was not perfect (Figure 10.1).

System proposed by authors aimed to decrease all potential vulnerabilities by having multiple authentication frameworks.

A system's architecture consists of server and client side. When the client side is a personal cell phone, the server side is more sophisticated. The key part of the server part is a machine learning MATLAB algorithm, which operates with the client through the Tomcat server.

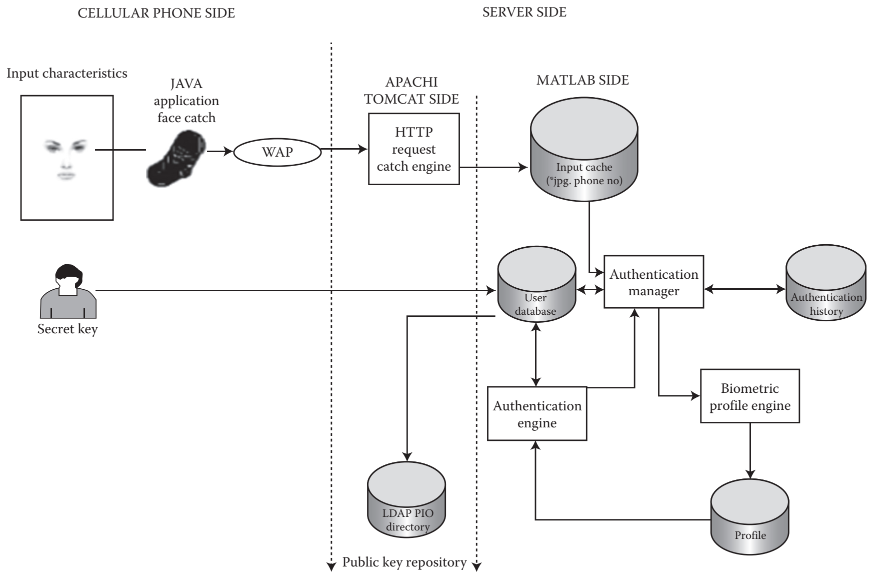

FIGURE 10.1 Client–server based mobile authentication system.

Machine learning algorithm in MATLAB works with the user database, authentication engine, biometric profile, and authentication manager. The authentication manager is the key of the systems as it redirects queries depending on input.

The database stores information about users and mobile devices, which are configured to work with this system, as well as supported biometric systems. The proposed system has great flexibility, any biometrics system can be used. Also in the presented system public key repository using public key infrastructure (PKI) was implemented, which aims to guarantee image authenticity after the image was sent to the server over the network.

CASE STUDY 4

Park and Yoo [13] propose a mobile face recognition algorithm which operates at low computational and memory cost. The algorithm is based on extracting features using Gabor– linear binary pattern (LBP) histogram alongside the scale invariant feature transform (SIFT)-based local feature descriptor.

Gabor–LBP is well known for detection texture features of objects, that is why this extractor is responsible for representing the local texture and shape of the face.

On the other hand, the SIFT-based part works better on the detection of local feature points in regions of interest (regions most likely to contain distinct information for each face). Such regions are eyes, mouth, nose, ears, etc.

The algorithm divided into two stages: train and test. In the training stage 40 Gabor filters are applied to one image which produces 40 different Gabor images. We need many different filters to decompose the images. Next, these images are transformed into Gabor–LBP images by applying the LBP operator. For each image, we extract the LBP histogram on the regular regions. During the test stage, the algorithm extracts the LBP histogram using the same technique. Besides, local feature points are extracted using the SIFT descriptor. SIFT-based features points are used in order to select local feature blocks. At the last stage, both LBP histograms are compared between training local features and test local features.

The proposed algorithm showed great accuracy performance: 96.1% for Electronics and Telecommunications Research Institute (ETRI) dataset and 98.4% for XM2VTS dataset with a small processing time which is critical for mobile devices.

10.3 CONCLUSION

Face recognition algorithms have been known for a while. New challenges in mobile biometrics created a new set of problems to be solved by researchers. Most of these solutions aim to decrease the differences between face recognition approaches on desktop computers and mobile devices. Observing the latest research, there are a few different ways to do this.

Using distributed architecture, with the client on a mobile device and a sophisticated face classification algorithm on servers is a straightforward naive solution. Such an approach allows the adoption of classic face classification algorithms and uses them for mobile biometrics. The downside of this method is that it requires a connection between the mobile device and server, which makes the method unreliable and vulnerable to hackers' attacks.

Another way of implementing mobile biometrics is to change the algorithms in the way they will be using the advantages of mobile platforms, such as graphics processing units (GPUs). Since GPUs were designed for matrix multiplication which makes them perfect fit for computer vision.

Due to the limited amount of memory resources and the necessity of getting results without long pretraining, online machine learning is another helpful approach. Instead of using computationally heavy offline training, in this case the prediction model which is stored on the mobile device itself gets updated with every new input.

Besides the techniques presented, there are several different assemblies of these methods, which are used depending on current tasks and resources.

REFERENCES

1. K Choi, KA Toh, H Byun. Realtime training on mobile devices for face recognition applications. *Pattern Recognition*, 44(2), 386–400, 2011.
2. N Goel, G Bebis, A Nefian. Face recognition experiments with random projection. In *Defense and Security*, Orlando, Florida, 2005.
3. D Achlioptas. Database-friendly random projections. *Proceedings of the Twentieth ACM*, Santa Barbara, California, 2001. pp. 274–281.
4. KA Toh. Deterministic neural classification. *Neural Computation*, 20(6), 1565–1595, 2008, MIT Press.
5. NY Liang, GB Huang, P Saratchandran. A fast and accurate online sequential learning algorithm for feedforward networks. *Neural Networks*, 17, 1411–1423, 2006.
6. MS Bartlett, JR Movellan, TJ Sejnowski. Face recognition by independent component analysis. *IEEE Transactions on Neural Networks*, 13(6), 1450–1464, 2002.
7. PS Penev, JJ Atick. Local feature analysis: A general statistical theory for object representation. *Network: Computation in Neural Systems*, 7(3), 477–500, 1996.
8. DD Lee, HS Seung. Algorithms for non-negative matrix factorization. In *Advances in Neural Information Processing*, Vancouver, BC, Canada, 2001.
9. M Rahman, N Kehtarnavaz. A hybrid face detection approach for real-time depolyment on mobile devices. In *Image Processing (ICIP)*, Cairo, Egypt, 2009.
10. P Viola, M Jones. Rapid object detection using a boosted cascade of simple features. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Kauai, Hawaii. Vol. 1, 511–518, December 2001.
11. T Caetanoa, S Olabarriagab, D Baronea. Do mixture models in chromaticity space improve skin detection? *Pattern Recognition*, 36(12), 3019–3021, 2003.
12. E Kremic, A Subasi, K Hajdarevic. Face recognition implementation for client server mobile application using PCA. In *Technology Interfaces (ITI)*, Dubrovnik, Croatia, 2012.
13. S Park, JH Yoo. Real-time face recognition with SIFT-based local feature points for mobile devices. In *Modelling and Simulation (AIMS)*, Kota Kinabalu, Malaysia, 2013.