Person Identification by Face Recognition on Portable Device for Teaching-aid System: Preliminary Report

Albadr Lutan Nasution
albadr@aisl.cs.tut.ac.jp

Bima Sena Bayu D.
bima@aisl.cs.tut.ac.jp

Jun Miura
jun.miura@tut.jp

Department of Computer Science and Engineering
Toyohashi University of Technology
Toyohashi, Japan

***Abstract*—We propose a face recognition system to identify a person and obtain his/her information, especially for teaching-aid contexts. This system is based on the communication between a portable device and a server. We evaluate face detection-recognition methods provided by OpenCV that will be used in the system. We also combine these methods with our illumination normalization and prove it can improve the detection and the recognition rate. With haar-based face detection and the illumination normalization, detection rate is stable at 95% in simple and severe illumination situations. Using Fisherface method with normalization, three training images per person are enough to achieve on average 96.4% recognition rate on Yale B Extended Database. Online prototype has been built and achieves up to 10 fps in performance.**

Keywords—face recognition, face detection, illumination normalization, fisherface, eigenface, lbph, opencv

I. Introduction

Nowadays many people have to keep track on every person they met with, as personal data become more important. If provided with database or internet, we can search people's information by their names or other clues. However, sometimes we may hardly remember anything about him/her especially in a real time encounter. Automatic person identification has existed ranging from a mass-related system, such as visa check counter and surveillance camera, into simple device unlock. Extending this for aiding social interaction is equally helpful.

The most important clue for person identification is *face*. If information is related to a face, we can identify the person and search for details using face query. This is very prospective as cameras now exist on every portable device. Not to mention wearable device era initiated by Google Glass is scheduled to begin this year. System to utilize face recognition using portable device and server communication are promising.

Simple instance is a system to help us remember the person in front of us. This system can be connected to social network database to cover our friends or all registered persons in the network. Virtually, everyone can be recognized. Although it is controversial, many are now trying to research and realize it.

Another instance will be to use this kind of system in a specific context. On a conference, a meeting, or a party, organizers can provide face recognition application that index all speakers information using their face. Participants can then use the application to find a speaker or an important person in crowd and see their public profile using a portable device.

Technology may improve many sectors of human lives but education – especially from educators' or lecturers' perspective – remains the same. Very few lecturers can remember the faces of their students moreover their performance. Consider a class setting, usually all students will be treated equally with the same non-interactive teaching. Using technology, we can help lecturer know which student needs more attention in class. This way, we can improve class interactivity, lecturer's productivity, and education as a whole.

One of our main aims is to build a system for helping an educator teach in class. In order to realize such system, an evaluation of face detection and face recognition methods, especially an already established and publicly available, is the basic part. This paper explains the basic idea of the teaching-aid system and preliminary investigations in regard to the methods that will be used. We investigated existing OpenCV library and incorporate our previous work in illumination normalization [1]. Evaluation of this combination and building the system prototype will be the main focus of this paper.

II. Related Work

There are some recent interesting works regarding system that utilized face recognition system for person identification, using a mobile phone or a glass-type wearable device.

Dantone *et al.* [2] proposed a complete face recognition system that used Facebook as image pool. In this way, training uses the manually tagged photos by Facebook users. They also proposed the use of social context e.g. user friends when recognizing faces. They argue that this work is complete in the sense that they have implemented all aspects in the system i.e. database crawling, face tracking, face retrieval, and information augmentation on portable devices.

Iwamura *et al.* [3] with *Haven't we met before?* proposed a system to remember person we met using a head-mounted display. This system automatically record video when we meet another person. On next encounter, this system provides the video of prior meeting and the event's time and location.

Yus *et al.* [4] with FaceBlock demonstrated a proof of concept that addresses privacy concern of face-related recognition system. One can announce face-signature to be captured by nearby Google Glass. Using this signature, Glass can recognize their face on camera. Glass will blur this face after recognizing it. Wang *et al.* [5] with InSight use similar approach of announcing signature to be used by others for recognizing us. However, they use non-face signatures e.g. shirt color.

Wang *et al.* [6] proposed a system to help prosopagnosics, patient with inability to recognize face. This system provides identity and relationship of the person in front of the patient.

Regarding methods provided by OpenCV i.e. Eigenface, Fisherface, and LBPH, many works [7] had been comparing them. This paper confirms their report and presents additional comparison of these methods in combination with illumination normalization method [1].

III. System Flow and Functionality

A. *Description of Teaching-aid System*

This section will describe the flow and the functionality of the proposed system.

Our system aims to help a lecturer know their students better, especially when teaching. Considering researches mentioned in Section II, the most relevant to this research are Dantone *et al.* [2] and Iwamura *et al.* [3]. However, the first dealt with social network-related database and the second with growing-as-used database. This approach seems unfit for a course's student database. In the context of teaching-aid, we think that the student will be the same most of the time. Every student also must register to the class. In this registration part, we can ask for additional information to be used in the system.

At least the following functionality must be present in the system. A lecturer will use a portable device and takes a face image and sends it to the server. The simplest approach will be that the device sends only one image. After that, the server does the heavy processing e.g., face detection and recognition for locating and identifying face. After information is retrieved from database, the server sends it to the device to be displayed. This processing flow is presented in Fig. 1.

This simple approach is minimum functionality in the sense that the device is only for taking images and presenting data (Fig. 1, green approach). However, for a good user experience, it is better to augment information on the target face image in real time when capturing scene. This approach may involve face detection in the portable device itself (Fig. 1, blue approach). We have implemented both approaches and present the comparison in Section IV of this paper.

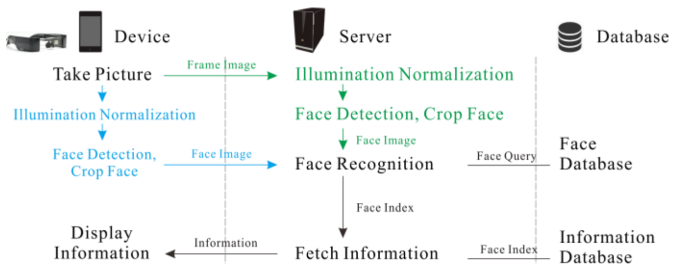


Figure 1. Processing Flow of System

While teaching, lecturer will use portable device preferably glass-type wearable device. Every time students' face is captured on camera, device will provide lecturer with information. Lecturer can then change their teaching, contents, or focus on specific student accordingly.

The information presented in this context is student-related information. The basic is student's identity to help lecturer remember their student. Another will be performance of the students. Server can record student activity in class. Quiz result, attendance, or homework submission will be useful for the lecturer to know which students need more attention in class. Additionally, server can record information that students set themselves e.g., their favorite topics in the lecture.

Every student in the class will have to provide some images of their face. Ideally, they provide multiple face images but if possible not too many. An evaluation is done to determine this number. The system will save the faces as index on database. All information will be based on this index.

B. *OpenCV Face Detection and Recognition Method*

In this preliminary research, we use an existing OpenCV library for face detection and recognition. Face detection uses haar-cascade method with frontal face models. For face recognition, there are three methods available: Eigenface, Fisherface, and LBPH that can be explained briefly as follows.

Eigenface reduces image dimensionality using PCA (Principal Component Analysis) to find the smallest meaningful components. It produces eigenfaces that represent the principal faces with some weight values. An image query is projected to these eigenfaces and its weight values are extracted. Face recognition is done by comparing these values to weight values of principal faces e.g. by Euclidean distance.

Fisherface uses LDA (Linear Discriminant Analysis) which reduce dimensionality within class specific. It maximizes the separation between classes and minimizes the variance in a class. Discriminative information between classes can be lost in Eigenface using dimensionality reduction of PCA but preserved in Fisherface using LDA.

LBPH (Local Binary Pattern Histogram) uses local feature i.e. pixel compared to its eight neighbors. This feature forms a binary pattern to replace each pixel. After that, image is divided by some regions and for each region histogram is calculated. Histogram of image query will be compared to these histograms on face recognition.

C. Illumination Normalization

Considering a classroom setting, usually the light is uniformly spread. However, face images can look differently on different light condition, for example with the color of light, illumination, and even angle. In addition, artificial light is usually put in many places. There will be a situation in which lighting is on the background. Images from camera will be darkened in this backlight situation. Thus, we think it is better to use illumination normalization within the system.

We integrate our previous work on illumination normalization [1] for face detection and recognition. In brief, this method transform illuminated image into appearance and shadow model. Appearance model is extracted from block-overlapped histogram equalization. Shadow model is pixel intensity normalized. Values from both models are used as inputs into a fuzzy inference system. This inference system has rule to convert both values into normalized image value. Genetic algorithm was also proposed to optimize this rule.

IV. Preliminary Experiment

This section describes our experiment on the preliminary stage of research. First, we evaluate haar-based face detection. Second, we compare performance of Eigenface, Fisherface, and LBPH face recognition methods. Third, we describe the prototype system that has been built and present processing time of each component in the system.

A. Evaluation of face detection and illumination normalization

We did evaluation of face detection on Yale A [8] and Yale B Extended [9] database. Yale A database represents simple light variation. Yale B Extended database represents severe light variation but we used only the first 10 persons. The results are presented in Table I.

We used frontal face haar-cascade default model provided by OpenCV. We also build our own frontal face model using haar-cascade training using normalized face images with 2400 positive images and 4500 negative images. We did evaluation on original image using the OpenCV model and on normalized image using our model.

On Yale A database, the default model performs well. Only one face image is undetected. Using our trained model, we performed slightly worse with eight faces undetected.

On Yale B Extended database, the default model has 20% faces undetected. Using our trained xml model, undetected faces are only around 5%, approximately the same as Yale A database evaluation, showing the stability of this combination.

TABLE I. Face Detection Result

Database	Total Image Used	Number of Undetected Image	
		Original Image	*Normalized Image*
Yale A	165	1	8
Yale B Extended	5850	1256	305

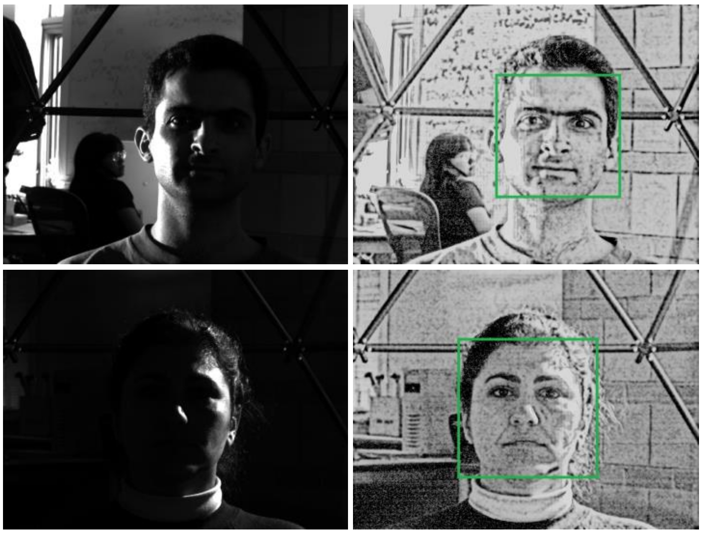


Figure 2. Face detection is failed on original image (left) but successful on normalized image (right)

Fig. 2 show examples result from Yale B Extended database. On such dark scenes, we can perform face detection well. This result showed that face detection in a severe light condition can be improved using illumination normalization, in this case using method described in [1].

B. Performance comparison of Eigenface, Fisherface, and LBPH

We did a performance comparison between three face recognition methods provided by OpenCV. We also compare the performance using original image and normalized image.

First, we did a cross validation on Yale A database [8]. This database has 15 persons with 11 observations each. We manually crop the face and resize it into 128x128 pixels. This database represents condition with simple light variation.

We did 11-fold cross validation i.e. one image is treated as evaluation set and the rest are as training set. The result is presented in Table II. We can see that the illumination normalization method in [1] improve the recognition rate but not much. This shows that the illumination normalization method is not quite effective on simple light variation.

Second, we did cross validation in the cropped images set of Yale B Extended database [9, 10]. This set has 38 persons with 64 observations per person with various illumination conditions. This database represents severe light variations.

In this evaluation, we divide the database into four sets randomly i.e. one set has 16 images per person. We treated one set as a training set and three set as evaluation set. The result of this evaluation is shown in Table III.

We can see in Table III that Fisherface method is robust under severe light variation even with original image. We can also see that illumination normalization is effective on severe light variation. Fisherface achieves recognition rate 99.8% in this evaluation.

TABLE II. CROSS VALIDATION ON YALE A DATABASE

Method	Recognition Rate	
	Original Image	*Normalized Image*
Eigenface	0.788	0.878
Fisherface	0.758	0.860
LBPH	0.903	0.915

TABLE III. CROSS VALIDATION ON YALE B EXTENDED DATABASE

Method	Recognition Rate	
	Original Image	*Normalized Image*
Eigenface	0.536	0.974
Fisherface	0.854	0.998
LBPH	0.663	0.825

TABLE IV. RECOGNITION USING ONE NORMALIZED IMAGE ON YALE B AND YALE B EXTENDED DATABASE

Method	Recognition Rate	
	Yale B	*Yale B Extended*
Eigenface	0.956	0.920
Fisherface	0.937	0.890
LBPH	0.670	0.497

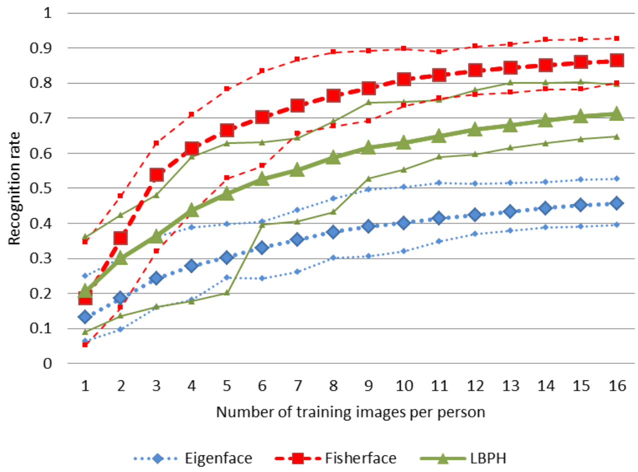
Table IV shows the recognition rate using only one training image per person on Yale B database and Yale B Extended database. This image is the first image in the database. It has the best illumination condition. We performed illumination normalization before training each method using this collection of images. We omit recognition rate using original images because it is less than 50% on every method except LBPH on Yale B which achieve 63%.

On [1], we train the illumination normalization method with Yale B database. Table IV shows that the recognition rate did not fall very much when using Yale B Extended. Using only one normalized images per person, we can achieve good recognition rate. Note that on these two databases (cropped set), pose and expression of the face do not change much.

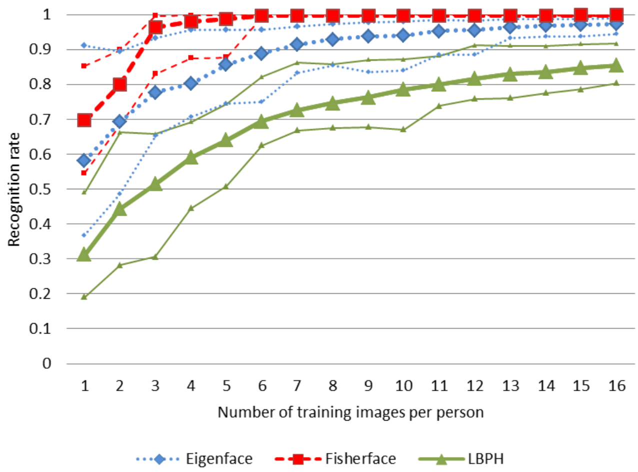
Lastly, we perform evaluation to understand how many training images per person are needed to have acceptable recognition rate. Every algorithm is evaluated with various numbers of training images from 1 to 16 with Yale B Extended database. In each number of training images, we chose randomly this number of images from the 64 available images. This contrasts with the previous evaluation that we intentionally choose the best illuminated image as training set. In real implementation, having good illuminated image is not guaranteed. Thus we do random choosing for this evaluation.

We did 20 times evaluation for each method. The result is shown in Fig. 2. Bold line represents the average recognition rate. Two thin lines on top and bottom of average line represent maximum and minimum recognition rate.

Fig. 3 shows results of this validation. Just as expected, Fisherface is the first as it deals with class separation, at least better than Eigenface. On severe light variation, Fisherface is superior though it is comparable with Eigenface on simple light variation as shown on Table II and III.



(a) Original image



(b) Normalized image

Figure 3. Recognition rate versus number of training image per person on Yale B Extended database

Figure 4. Some Yale B Extended Database cropped image used in training

Recognition rate of Eigenface and Fisherface improve greatly using the normalized images. In principle, both reduce dimensionality of image. Using normalized normalization, there is virtually no light variation present in input image. In other words, images has most meaningful component. This helps the computation for Eigenface and Fisherface.

The recognition rate of LBPH using normalized image did improve but not much, only about 10%, and fell into third place in this evaluation. This shows that LBPH and illumination normalization does not get along very well.

Fig. 3(a) shows that we need at least 10 to 12 images per person to achieve a good recognition rate for Fisherface. This number is bigger for Eigenface and LBPH. Fig. 3(b) shows that using illumination normalization [1] can reduce number of training image. On six images, maximum and minimum recognition rate of Fisherface converge. This method achieve 96.4% average recognition rate on three images per person.

Considering these results, we can say that among three methods provided by OpenCV, Fisherface is the best. Using illumination normalization, we need only 3 images per person to achieve good recognition rate. This is ideal for the system i.e., each student can provide multiple but not too many images if possible. Three is a reasonable number. Considering Table IV also, the system can also work with good result if provided with only one good illumination image.

C. *Processing time and online system*

The evaluation is done in a notebook PC (Intel Core™i5 1.70 GHz, 4GB RAM, Windows 8 x86-64). The processing time of each component is presented in Table V.

The processing time of face detection and illumination normalization is the average of the 5850 experiments done for the face detection evaluation on Yale B Extended database. We use image size 640x480 pixels. The processing time of training and recognition component is the average time of the number of training image evaluation described before i.e., average of training time with from 1 to 16 images per person using Yale B Extended cropped images (64x64 pixels). Recognition time using Eigenface and Fisherface is fast and quite stable with number of training images. Recognition time of LBPH is slow but varied with number of training images. However, training time of LBPH is faster than the other.

TABLE V. PROCESSING TIME OF SYSTEM COMPONENTS

Component	**Processing time (ms)**	
	Original Image	***Normalized Image***
Illumination Normalization	57	-
Face detection	137	280
Training – Eigenface	3620	3533
Training – Fisherface	4947	4864
Training – LBPH	244	251
Face recognition – Eigenface	0.26	0.26
Face recognition – Fisherface	0.25	0.25
Face recognition – LBPH	17.78	18.76
Illumination Normalization in Android	400	-

TABLE VI. EVALUATION WITH AISL FACE DATABASE

Method	**Recognition Rate**			
training images per person	*3*	*4*	*5*	*6*
Eigenface	0.521	0.592	0.619	0.638
Eigenface with normalization	0.612	0.689	0.720	0.743
Fisherface	0.558	0.652	0.687	0.734
Fisherface with normalization	0.581	0.682	0.728	0.761
LBPH	0.659	0.722	0.756	0.775
LBPH with normalization	0.663	0.736	0.780	0.820

TABLE VII. PERFORMANCE OF SYSTEM PROTOTYPE

Location of Face Detection Implementation					
Server		**Device**		**Device with Normalization**	
Time (ms)	*FPS*	*Time (ms)*	*FPS*	*Time (ms)*	*FPS*
603	1-2	100	9-10	330	3-4

We have built the system for both server and device with scheme presented in Fig. 1. Screenshot of the application in portable device is presented in Fig. 5. We use Nexus 7 with Android 4.4.3 for portable device. The performance results are summarized in Table VII.

We begin with scheme that put all heavy process on the server. Average latency from sending frame from device to server until receiving result is 603 milliseconds. This includes time for sending image and processing time in server. We use university internal internet network for this measurement. As presented in Table V, half of this latency is processing time.

With this scheme, the performance of stream approach i.e., send frame image from the camera and overlay information on the device in real time, was at most 2 fps. Because it is too slow, one image approach is better for the implementation of this scheme. User takes an image when he/she want to identify a person and server only processes this one image.

However, one image approach does not have a good user experience. It is better if the system only need minimum user input for displaying information. Thus, we also implement stream approach with face detection in portable device using *OpenCV for Android*. Device only send cropped face image after face is detected. Because the cropped face image has small size, the time for sending this image is lower. The processing time for face detection with OpenCV for Android is approximately the same as on the server. With this approach, we got better performance around 10 fps.

We also use our illumination normalization algorithm on the Android device. However, this process is very slow in the portable device. For one frame with size 640x480 pixels, it took 400 milliseconds. For this processing, we resize the image into 480x360 pixels and it took 230 milliseconds or around 4 fps in online system. We conclude that illumination normalization have to be implemented efficiently so that it can have a better performance on portable device. Considering this result, probably it is better not to use the illumination normalization on the device or at least not on every frame.

In the screen shoot that presented in Fig. 5, we provide image with bright lighting in the background. We can see that the system can detect the face and fetch information of the detected face in backlight situation.

We also did an online evaluation with our own database. We collected some frontal face images from our lab members. We called it Aisl Face Database. This database has 20 persons all male with 8 images per person. The size of each face image is 128x128 pixels, cropped with face detection algorithm from original picture. We did not control anything about these images, thus they have variation in background, illumination,

Figure 5. Image result of face detection and recognition which received from server is displayed on portable device

Figure 6. Some Images on Aisl Face Database

emotion, glasses, etc. We consider this database just like real photos that will be uploaded by students for their profile picture. Examples of this database are presented in Fig. 6.

We evaluated recognition rate of the face recognition methods with and without illumination. We did the evaluation with 3-6 training images per person. Images are chosen randomly from 8 available images. We did the evaluation 50 times. The average recognition rate is presented in Table VI.

The result of this online evaluation is quite different with using Yale B Extended Database. Eigenface and Fisherface has comparable recognition rate. Fisherface is still superior to Eigenface just like prior evaluation but in this evaluation it did not improve much with illumination normalization.

LBPH also has the same result as prior evaluation as it did not improve with illumination normalization. However, it has the highest recognition rate in this case. We suspect that the background variations in this database made Eigenface and Fisherface did not perform as well as in the Yale B Extended.

Using Aisl face database in the online system, we can identify our lab members. However, in case of online system the face is not always frontally captured. In those cases, false recognition is prevalent. On that note, we need to expand the database and the research to account for most face poses.

V. CONCLUSION

We present the functionality of face recognition system for teaching-aid using a portable device. Toward the goal of building this system, we did some evaluations of face detection and recognition that would become part of the system. We evaluated methods that are provided by OpenCV.

Evaluation was also done in combination with illumination normalization. We showed that illumination normalization [1] can improve detection rate in severe illumination conditions. Face detection with illumination normalization is stable at 95% detection rate on simple and severe illumination conditions. We also showed that integrating illumination normalization can give high improvement in recognition rate.

Among face recognition methods provided by OpenCV, Fisherface achieves the highest recognition rate. Fisherface yields 99.8% accuracy on cross validation using normalized images of Yale B Extended database. Investigating numbers of training images, we also showed that Fisherface give the best result with the smallest number of training images. It achieved 96.4% average recognition rate using only three normalized images per person in training.

We also did an evaluation with our database that represents real profile picture that will be used by students. Using this database, we built an online prototype of the face recognition system. Using the approach with face detection done on the server, the performance is less than 2 fps. Doing face detection in device and sending only face image improved the performance to 10 fps. However, the implementation of the illumination normalization algorithm was not efficient enough to be used on the portable device as it reduced the performance to 4 fps.

On the context of teaching-aid system, we have to explore what kind information is really needed by the lecturer on class, how this information is presented, and whether it is really useful for the lecturer. We plan to take a survey and an online experiment to have a better understanding about this topic.

REFERENCES

[1] B. S. Bayu D. and J. Miura, “Fuzzy-based illumination normalization for face recognition,” in Proc. IEEE ARSO, 2013, pp.131–136.

[2] M. Dantone, L. Bossard, T. Quack, and L. J. Van Gool, “Augmented faces,” in Proc. IEEE ICCV, 2011, pp.24–31.

[3] M. Iwamura, K. Kunze, Y. Kato, Y. Utsumi, and K. Kise, "Haven't we met before?: a realistic memory assistance system to remind you of the person in front of you.," in Proc. 5th AH’2014, ACM, 2014.

[4] R. Yus, P. Pappachan, P. Kumar, E. Mena, A. Joshi and T. Finin, “Demo: FaceBlock: Privacy-Aware Pictures for Google Glass”, in Proc. MobiSys’14, ACM, New York, USA, 2014.

[5] H. Wang, X. Bao, R. Choudhury, and S. Nelakuditi, “InSight: recognizing humans without face recognition,” in Proc. HotMobile’13, ACM, Jekyll Island, Georgia, USA, 2013.

[6] X. Wang, X. Zhao, V. Prakash, W. Shi, and O. Gnawali, “Computerized-Eyewear Based Face Recognition System for Improving Social Lives of Prosopagnosics,” in Proc. 7th PervasiveHealth, 2013, pp.77–80.

[7] A.M. Martines and A.C. Kak, “PCA versus LDA”, in IEEE TPAMI, Special Issue on Face Recognition, vol.23/2, 2001.

[8] P. N. Bellhumer, J. Hespanha, and D. Kriegman, “Eigenfaces vs. fisherfaces: Recognition using class specific linear projection,”. in IEEE TPAMI, Special Issue on Face Recognition, vol.17/7, 1997, pp. 711–720.

[9] A. Georghiades, P. Belhumeur, and D. Kriegman, “From few to many: illumination cone models for face recognition under variable lighting and pose,” in IEEE TPAMI, vol.23/6, 2001, pp. 643–660.

[10] K.C. Lee, J. Ho, and D. Kriegman, “Acquiring linear subspaces for face recognition under variable lighting”, in IEEE TPAMI, vol.27/5, 2005, pp. 684–698.