Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 152 (2019) 274–281

Procedia Computer Science

www.elsevier.com/locate/procedia

International Conference on Pervasive Computing Advances and Applications – PerCAA 2019

Energy Efficient Face Recognition in Mobile-Fog Environment

Dhritiman Mukherjee[a],Rik Das[b],Souvik Majumdar[c],Sourav Ghosh[d],Sudeep Thepade[e], Abhishek Basu[f] *

[a]*Cloud computing and Information Security,Inurture(Centurian University),Bhubaneswar, India*
[b]*Dept. of Information Technology,Xavier Institude od Social Service,Ranchi,India*
[c]*Dept. of Computer Science and Engineering,RCC Institute of Information Technology,Kolkata,India*
[d]*A.K Choudhury School of Information Technology,University Of Calcutta,Kolkata,India*
[e]*Dept of Computer Engineering,Pimpri Chinchwad College of Engineering,Pune,India*
[f]*Dept of Electronics and Communication Engineering, RCC Institute of Information Technology,Kolkata,India*

Abstract

Proliferation in technological advancements has leveraged the evolution of cell-phones as powerful smart-phones with high computing competence. Integration of sensors and high definition camera has empowered the smart-phones as a tool to solve real world problems. Multiple applications of face recognition have been developed to address the security challenges using the computation power of handheld (mobile) devices. However, one important limitations of mobile devices is energy efficiency. This work has proposed an energy efficient fog-based face recognition using mobile platform. The architecture in proposed system initially performs the facial part detection on mobile device and uploads the selected part by discarding the rest. This reduces the network traffic and attempts to reduce energy consumption. Preliminary results have indicated a saving of approximately 93% of energy in mobile devices.

Keywords: Face Recognition, Fog Computing, Mobile Computing, Feature extraction, Energy Efficiency;

1. Introduction

Facial recognition is not an ordinary task and it's a delightful research area with lots of powerful use cases that helps society across various dimensions. It's use cases can be successfully implemented as a regular administrative

Corresponding author. Tel.: +91 912-304-7193;
E-mail address: dhritiman59@gmail.com

1877-0509
10.1016/j.procs.2019.05.016

feature in industry and academia. For example, a service company can adopt this technology for regular attendance for their employees as well as it can be also used for students' attendance in academics. This kind of technology is facilitated by different techniques of image processing. Modern advancements in deep learning or machine learning have significantly bettered accuracy in image processing results. On the other hand, smartphones are becoming immensely powerful devices. The inbuilt cores and smart machine learning libraries have extended huge support to perform complex machine learning tasks easily. But the tasks are computation intensive and require high computing resources, namely, processing power and battery life.

In paper [2] automated process for identifying significant region of an image for feature extraction has been mentioned. This work presents a Fog based facial recognition with the help of smartphones. This allows a smartphone to take an image and identify the facial part of the image. Further it crops and sends only the facial portion of the image instead of sending the whole image to the fog controller or cloudlet. On the other hand, after receiving the processed image the fog controller or cloudlet encodes the feature and compares it to other images in the dataset and sends back the result to the client. The proposed system achieved 93% of energy gain in performance after sending only the processed part instead of sending complete image. The remaining paper is organized as follows. Related study is presented in section II. The methodology of the system is described in section III. Result and discussion is given in section IV. Future work and conclusion are discussed in section V and VI respectively.

2. Related Study

Face recognition is a significant research domain. It has gained a immense popularity because of the wide ranges of applications such as entertainment, smart cards, security of informations and law enforcement. The main constraints of face recognition are intensity, illumination, pose, and large occlusion. A slight variations of light intensity or change of face coordinates may increase the error rates in face detection.Modern changes in machine learning or deep learning significantly improved the accuracy of face recognition techniques. However, facial recognition techniques are computationally very high and required resources beyond the capacity of mobile devices, posing limitations on contribution capacities of those devices. From past several years, diverse algorithms are proposed for face detection, as in [3]. Human face is one the objects of interest to the researchers because of the various critical applications related to the human face ranging from surveillance systems to entertainment applications [4]. One of the more critical applications is used in Criminal detection using smartphone [5]. Confirming the proper face identity or reducing face spoofing [6] is also a very important issue in biometric authentication. In mobile environments, resource allocations in cloud extends beyond the public cloud. A smartphone may also offload the complex computations to cloudlets [6] and mobile cloud [7], [8]. A mobile user exploits virtual machine (VM) technology to rapidly instantiate customized service software on a nearby *cloudlet* and then uses the resultant service over a wireless LAN; the mobile device typically functions as a thin client with respect to the service. A cloudlet is a trusted, resource-rich computer or cluster of computers that is well-connected to the Internet and available for use by nearby mobile devices. Paper [9] describes attendance recording system which follows Viola-Jones algorithm in both controlled and uncontrolled environments and proved 100% and 60% accurate in controlled or uncontrolled environments respectively. Paper [10] focus on the different real time case of face recognition which is dry and wet face condition author named this type of case as wet face recognition (WFR). Modified bilateral framework is proposed in the paper to improve recognition performance. Paper [11] describes how computational overhead can be offloaded to network edge devices in order to improve performance.

In 1960, machine based(automated)human face recognition was first invented. The first semi-automated system for face recognition required the system to locate features of different human sense organ and mouth on the photographs before it calculates the distances and the ratios to a common reference point, which were then compared to the store data(reference). In 1970s, the problem with both of these early solutions was that the measurements and locations were manually computed. In the year of 1990, Sirovich and Kirvy used PCA, which is standard linear algebraic technique, on face recognition problem. This invention considered as a pioneer for face recognition.Nowadays,extensive used of smartphones, different biometric authentication, such as face and fingerprint recognition, is becoming widely popular for secure user identity.

3. Methodology

The proposed system consists of two parts, namely, edge device and server. Edge devices are android smart-phones used for face data collection and servers are those which comprise the Fog environment and process the data collected by smart-phones. Android smartphones consist of two parts, namely, image collection part and face detection part. Using first part (image collection part) of user interface in edge device the user or client can contribute images to the fog server. User or client can click an image and send the entire to the fog server also. Using second part (face detection) of user interface user can recognize the faces. Instead of sending the entire image to the fog server, only some part of the image will go to the server. After identifying the ROI (Region of Interest) of the image, the portion of the face will be detected automatically and then it will send to the server for facial recognition. Local server doing several tasks like feature extraction, facial feature matching and sends the response.

A. ROI detection in Mobile App

ROI in image processing is identified as the portion of the image which has significant contribution in robust feature extraction. In our case only facial portion is needed for detection of the human identity. So, the proposed algorithm aims to detect the facial portion of the image and crop the same from the rest of the image.

The algorithm operates in the user interface of the mobile device to enable capturing image from user for detection of ROI in the facial portion of the image. The identified ROI portion is cropped and forwarded to the server for facial detection. For processing purpose the app is solely developed for android devices using android studio IDE and JAVA programming

language. In this work, the Mobile Vision APIs has been used for new Face Detection APIs to locate human faces in image and video for better and faster detection. It is available with the release of Google Play service 7.8.

For detecting face, a class named FaceDetector needs to be created. FaceDetector.Builder are used to define multiple parameters for quickly identifying faces. Following codes snippet shows basic face detection in Fig 1:

```
FaceDetector deector = new FaceDetector.Builder(getContext())
.setTrackingEnabled(false)
.setLandmarkType(FaceDetector.ALL_LAND
MARKS)
.setMode(FaceDetector.FAST_MODE)
.build();
```

Fig. 1

Fig. 2 and Fig. 3 show the overall flow diagram and overview of the proposed system during the image processing in mobile devices.

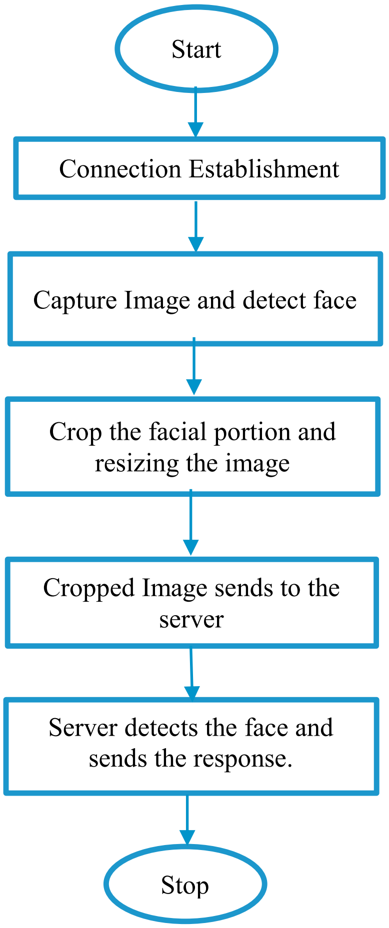

Fig. 2

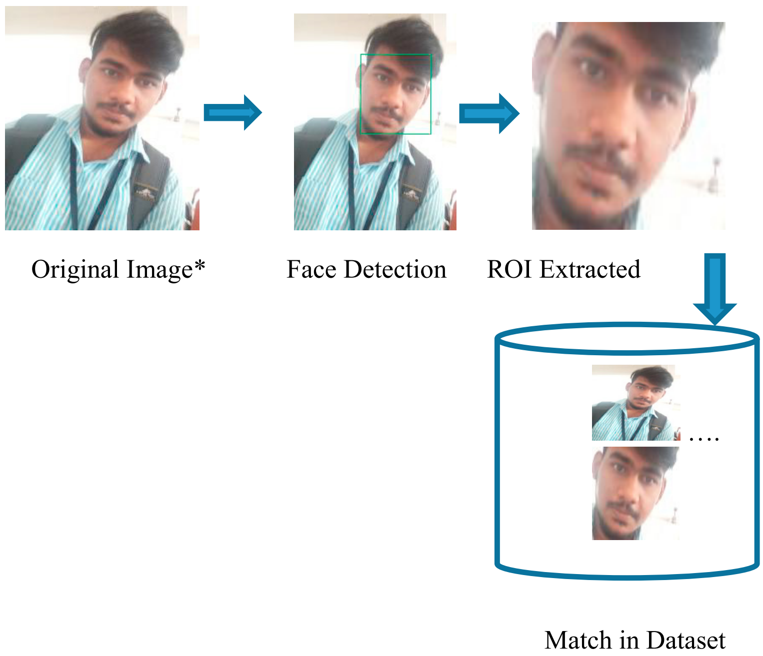

Fig. 3

B. Face Recognition in Server

Face recognition in server is the core of the system where the processed image from mobile device is compared with rest of the images from dataset. Here the sever performs the following task in four stages. The stages are enlisted as follows:

- feature vector extraction
- encoded feature vector
- comparison
- sending result to client

Face identification from a given image can be performed considering multiple parameters as a pattern to create signature. The image has several semantics out of which 5 are selected in this approach as enlisted below:

- height of the face (cm)
- width of the face

- average color of basic
- color components present in face (Red, Green, Blue)
- width of human lips(cm)
- height of human nose (cm)

The above features exhibit higher similarities for faces with related semantics and demonstrate a low score for un-related types. The numerical values of the above enlisted five features are considered as feature vector for each face.

Feature Vector Extraction: -During this stage server program reads the processed image sent from the mobile device and mapping of various feature and convert it into a feature vector as shown in Table 1.

Table 1. Feature Vector

Height of Face(cm)	Width of face(cm)	Average color face(R,G,B)	Width of lips(cm)	Height of nose(cm)
23.8	14.7	(225,223,170)	5.3	4.6

So, the image forms a vector which is represented as the format (23.8,14.7,225,223,170,5.3,4.6). Coding snippet in Table. 1 shows how processed image is loaded and feature vector is encoded.

Encoded Feature: - This is the second stage in server. During this stage each image in dataset encoded into their feature vector in runtime. Clearly, when we have images that is similar to query image the derived feature vector will be quite similar. Another way, we can also say that the two feature vector distances will be quite small.

```
# load image
    Cropped_image=load(face_recognition)
    Uploaded_image_path_
 #encoded the loaded image into feature vector
     encoded_cropped_image=face_recognit
     ion.face_encodings(cropped_image)[0]
# load your image
```

Fig. 4

Comparing: -During this stage encoded features in the dataset are compared with the feature in query image after looping all over the image. This comparison is done during runtime. Result send: - After the comparison result is sent to the smartphones.

C. Dataset in Server:

In this research, dataset in server are composed of by the contribution of the user using the application running in the smartphones. Users who are willing to participate for the contribution for the dataset can take image and send it to the server.

D. Database in Server

The database component is designed with MySQL, the database table consists with details of faces collected from the different sources. Database table consists with image id, image name, image file location. As soon as client uploads the image for contribution it stores into the database and new image id will create for that image. Images in database can be fetched or identified by its unique image id.

4. Result and Discussion

The client part is designed in android and tested with Samsung J7 (Nougat) octa-core with 3300 mAh battery capacity and Motorola e^5 octa-core (Oreo) smartphones with 4000mAh battery capacity. Fog controller or local server is implemented using 8 GB of RAM and i3 processor. Measurement of the time is calculated by comparing timestamps after completing each operation. Energy efficiency is calculated programmatically after preforming the tasks. In this experimentation the client app sends the image to the fog controller after pre-processing it. This pre-processing means detecting and cropping only facial part of the image. For identifying image into the server only facial portion are needed. So, we can eliminate the rest of the part of the image. By eliminating or cropping the rest of the part from image we get approx. 93% gain in energy. The gain is achieved by reducing 59.9 KB of image to a size of 20.4 KB by means of pre-processing technique. Different computative study for energy consumption is shown in Fig 5. In first case, the complete image is captured by the user and is sent to the fog controller for further processing. For uploading 59.9 KB of image to the fog server energy consumption is 2.4 Joules. In second case the image is pre-processed after being captured by the user and is further processed to reduce it to 20.4 KB. Thus, energy consumption is 1.25 Joules for uploading to the server. The overall response time comparison is showing in Fig. 6.

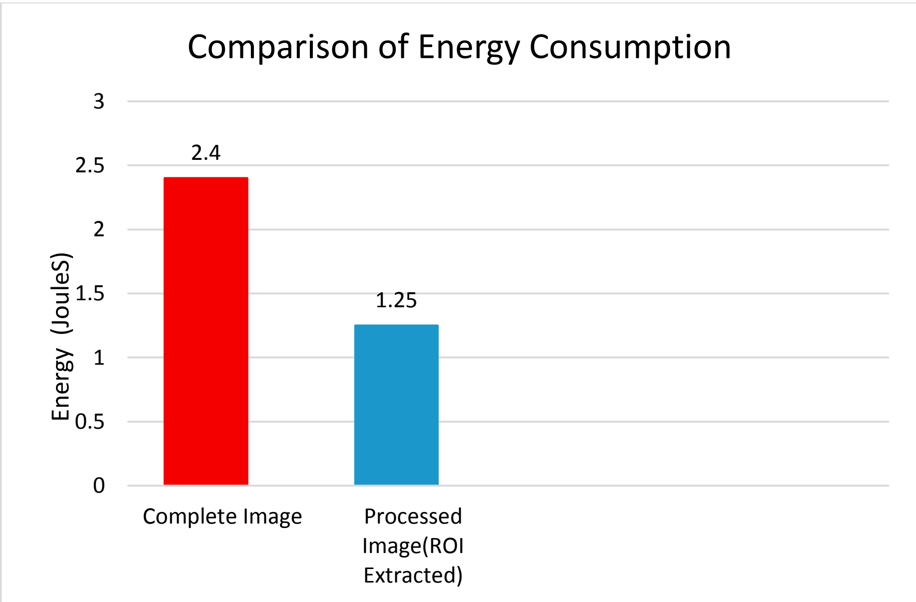

Fig. 5

The comparisons shown in Fig. 5 and 6 have evidently established the efficiency of our proposed technique. It has clearly revealed energy savings by deploying the processing power of the mobile device to the fog controllers.

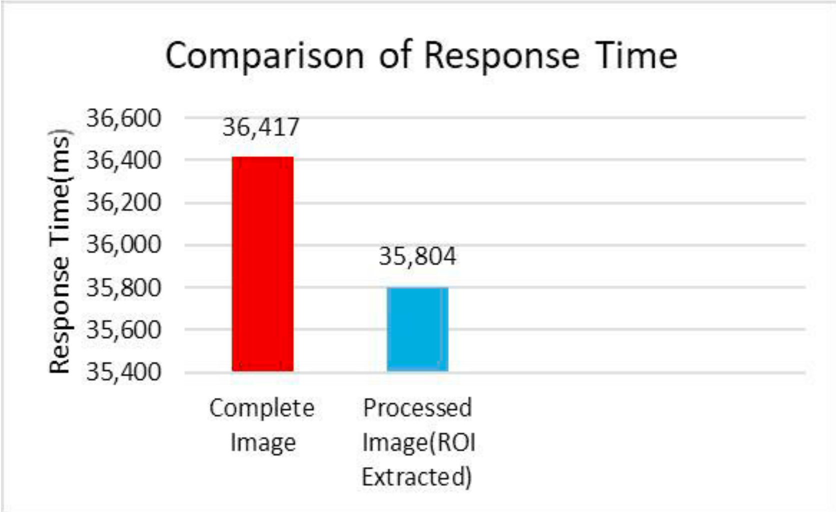

Fig.6

5. Conclusion

This paper presents a system that initially detects the face to identify the facial portion and uploads the selected part (ROI). Further, processing and feature extraction is carried out only on the selected portion which enables energy savings and efficient computation for processing of the face recognition algorithm. The research work has been able to portray a saving of 93% of energy consumption by eliminating the irrelevant image portions which are not significant for feature vector extraction. Feature extraction and image comparison are done dynamically at the Fog server end which has played a crucial role in energy savings. Implementation of mobile-fog environment in this approach has kindled further scopes to optimize and address problems related to limited energy resources by deploying fusion of efficient image processing techniques with Fog computing. This work can be extended with a distributed approach which will enable every client to contribute in fog dataset for training the system. We will be exploring implementations of fog controller in Raspberry Pi for more ubiquitous. The current work is based on a customized dataset may be extended towards experimentation using public dataset available in Kaggle, UCI machine learning repository and so on.

References

[1] FogComputing,http://www.cisco.com/web/sollution/trends/iot/fogcomputing.html.

[2] Das Rik, Walia Ekta, "Partition selection with autoencoders for content based image classification" *Springer London, Neural Computing and Application,* pp. 1-16 , 2017.

[3] R. Chellappa, C. Wilson, and S. Sirohey, "Human and Machine Recognition of Faces: A Survey," *Proc. IEEE*, vol. 83, no. 5, pp. 705-740, 1995.

[4] A. Samal and P. Iyengar, "Automatic Recognition and Analysis of Human Faces and Facial Expressions: A Survey," *Pattern Recognition*, vol. 25, pp. 65-77, 1992.

[5] Lamiaa A. Elrefaei1, Alaa Alharthi, Huda Alamoudi, Shatha Almutairi and Fatima Al-rammah," Real-time Face Detection and Tracking on Mobile",*IEEE International Conference on Anti-Cyber Crimes(ICACC)*,March 2017.

[6] Cuervo, E., Balasubramanian, A., Cho, D. K., Wolman, A., Saroiu, S., Chandra, R., & Bahl, P. (2010, June). MAUI: making smartphones last longer with code offload. In Proceedings of the 8th international conference on Mobile systems, applications, and services (pp. 49-62). ACM.

[7] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "The case for vm-based cloudlets in mobile computing," IEEE Pervasive Computing, vol. 8, pp. 14–23, Oct. 2009.

[8] E. E. Marinelli, "Hyrax: Cloud computing on mobile devices using mapreduce," Master's thesis, Carnegie Mellon University, 2009.

[9] Surekha, Borra, et al. "*Attendance recording system using partial face recognition algorithm*." Intelligent techniques in signal processing for multimedia security. Springer, Cham, 2017. 293-319.

[10] Dharavath, K., et al. "*Face recognition under dry and wet face conditions*." Intelligent techniques in signal processing for multimedia security. Springer, Cham, 2017. 253-271.

[11] Hu, P., Ning, H., Qiu, T., Zhang, Y. and Luo, X., 2017. "*Fog computing based face identification and resolution scheme in internet of things*". IEEE transactions on industrial informatics, 13(4), pp.1910-1920.

*__Declaration__: - Prior permission is granted by the person in the image (Pic.3) for using his face as a testbed for the experimentation process carried out in this research papers,