Decentralized Face Recognition Scheme for Distributed Video Surveillance in IoT-Cloud Infrastructure

Anang Hudaya Muhamad Amin, Nazrul Muhaimin Ahmad, Afiq Muzakkir Mat Ali
Thundercloud Research Lab
Faculty of Info. Science & Technology (FIST)
Multimedia University
Jalan Ayer Keroh Lama, Melaka 75450
Malaysia
Email: {anang.amin,nazrul.muhaimin}@ mmu.edu.my

***Abstract*—People monitoring and tracking activities in surveillance system usually generate massive amount of data from Internet-of-Things (IoT) devices such as cameras. Several issues need to be addressed, including data migration over limited bandwidth and high latency in communication networks. This paper presents an initiative to develop a decentralized face recognition scheme for distributed surveillance system that make use of an integrated framework of Internet-of-Things (IoT) and cloud computing. The decentralized face recognition approach implements a two-stage procedure, including face detection and extraction and face matching. Face detection and extraction are performed on a cloudlet that is located close to the surveillance cameras, hence minimizing the need for massive data transfer to the remote processing center. On the other hand, face matching process is carried out on the face feature vector within a private cloud environment. A case study conducted on the effectiveness of the proposed scheme in detecting "missing" person indicates that the procedure works effectively in the IoT-Cloud infrastructure.**

I. Introduction

With the advancements in existing Internet technology, the ability to perform continuous and seamless monitoring in a distributed manner has been made possible. Nowadays, different kind of monitoring devices can be interconnected to create an ecosystem that is commonly known as Internet-of-Things (IoTs). Monitoring applications such as distributed video surveillance could be deployed over such environment. Nevertheless, there are several issues to be addressed including massive data processing and storage, as well as communication latency.

Video surveillance has become increasingly important in everyday use, such as in the activities involving identification of individuals or objects and detection and prevention of abnormal activities. The existing technology has become more sophisticated with its ability to perform continuous monitoring and detection using distributed devices, as in the IoT ecosystem. Distributed video surveillance on IoT infrastructure requires an effective and efficient recognition system for detection and tracking of individuals or objects. Current implementations of such system involves complex computations of massive data and requires high communication bandwidth, due to massive data transfers between detection and processing devices. In this paper, we present a decentralized face recognition scheme for distributed video surveillance that can be deployed in the IoT system. In addition, we propose an integration of IoT and cloud-based system for massive data processing and storage. We also introduce a scheme that utilizes *cloudlet: "a data center in a box"*, that is able to perform feature extraction and processing on-site for lowering the requirement for high bandwidth in distributed video surveillance application.

The proposed face recognition scheme implements a decentralized two-level procedure: feature extraction on the cloudlet using Haar feature-based cascade classifier [6], [8] and face recognition on the private cloud system using Local Binary Patterns Histogram (LBPH) face recognition scheme [1]. The proposed approach minimizes network bandwidth requirements as the cloudlet only need to send extracted features to the private cloud, rather than the entire scene images. Furthermore, this scheme provides a scalable storage and processing facilities for all extracted data using a private cloud infrastructure.

This paper is organized as follows. Section II describes some related works in recognition applications within the IoT infrastructure. Our proposed system infrastructure that utilizes IoT-cloud integration is presented in Section III. Section IV provides a detailed description of our proposed decentralized face recognition scheme for distributed video surveillance application. A discussion on our works and some preliminary results are presented in Section V. Finally, Section VI concludes the paper.

II. Related Works

Nowadays, surveillance video cameras have been widely used in both public and private places for the security and safety purposes such as monitoring, tracking and access control. It is often desirable to use specific traits such as face to identify a person. In the area of video surveillance, a cloud-based system can provide a scalable and effective video

978-1-5090-0931-2/16/$31.00 ©2016 IEEE

management and user can access the video recording from any IP camera through Internet. With the advantages in processing and storage, cloud-based service are expected to drive more growth in video surveillance. The applications of object recognition in IoT-Cloud infrastructure can be seen in the work of Soyata et al. [11] on face recognition within mobile-cloudlet-cloud architecture (MOCHA). The work basically addresses the communication latency through task partitioning within the proposed architecture. This architecture was designed to run on the hostile environment such as battle zone. When running such complex applications, mobile devices will capture the observed image and send it back to the running servers in the cloud to match with the database. And it will return back the match result to the mobile devices. Both of the result and images are being transferred using satellite link with a long latency problem which is the major limitations and can caused a delay. In order to address this problem, cloudlet is being introduced. Pre-processing and caching is accomplished inside the cloudlet. With this cloudlet, the mobile devices will sent the observed images over a high speed bandwidth connection to the cloudlet instead of to the main cloud. The cloudlet will perform either all the processing or maybe some of it first. If the cloudlet does not have enough resources such as lacked of necessary data from database, cloudlet will send the remaining of the processes to the cloud. One possible limitation of this implementation is such that the bandwidth requirement is still considerably high due to large amount of images need to be transferred to the cloud.

Another example of recognition scheme being deployed in the IoT infrastructure is eyeDentify. It is a software developed by Kemp et al. [7] that performs cyber-foraging for improving the efficiency of the object recognition processes. Cyber-foraging is a pervasive computing technique that allows resource-constrained devices to offload computational intensive processes to more resourceful devices. eyeDentify application perform an object recognition process on high-performance distributed computing middleware that is being used in the mobile devices. The intensive recognition algorithm can be executed either in a single compute node or a collection of distributed multiple devices. This criteria make it a good choice to evaluate the cyberforaging. Cyber-foraging being describe as an effective method on how to handle the complex processing on the limited resource computation such as mobile devices [10].

Other recent works related to object recognition within the IoT and cloud infrastructure can be seen in the works of Hong et al. [5] on programming model involving IoT and mobile devices and Xiao et al. [13] on mobile crowd-sensing.

III. IoT-Cloud Infrastructure

Our design for distributed surveillance system is taking into account the infrastructure for deployment. We propose an integrated IoT-cloud infrastructure with cloudlet implementation for cyber-foraging purposes. The proposed design is similar to MOCHA [11] implementation, with an exception that we introduce IoT-cloudlet-cloud integration, rather than mobile-cloudlet-cloud. The proposed scheme follows the five-layer architecture from the work of Tao et al. [12] with minor modifications as described in [3]. Fig. 1 illustrates this system architecture. Note that the proposed scheme utilizes both IoT and cloud for processing, since some processing services are being done within the IoT sub-system.In general, IoT components can comprise of any devices. In this work, video surveillance camera is acting as an IoT entity. High latency between camera networks and the cloud server is the biggest challenge for using cloud as a server. Given this challenge, our main focus is on designing a cloudlet framework as a bridge between surveillance camera and public cloud convergence as shown in Fig. 2. Our specific objective is to design the formation of cloudlet and the integration of surveillance camera-cloudlet-cloud. Having this form of system architecture may increase the effectiveness of our proposed scheme by migrating large computation and complex processing from the cloudlet to cloud, while maintaining simple and time-critical operations on the cloudlet, which is closer to the surveillance camera.

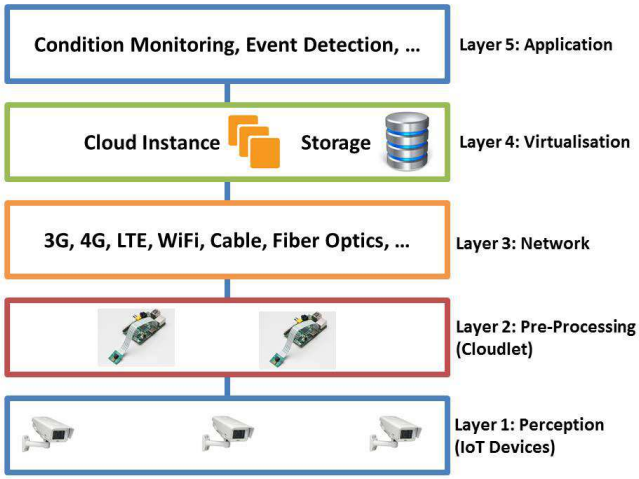

Fig. 1. IoT-Cloud layered system architecture for distributed surveillance application.

In the proposed IoT-cloud scheme, surveillance cameras are connecting with the cloudlet on the high-speed local area network (LAN). Images captured by the cameras will be sent to the cloudlet for pre-processing. Inside the cloudlet, the raw images will undergo face detection process and also face vector extraction process. On the other hand, cloudlet is connected with public or private cloud on low-speed wide area network (WAN). After the completion of pre-processing stage within the cloudlet, the face vector will be sent to the cloud for matching processes with existing image vectors on the database. After getting the result, the user will be notified. In this implementation, cloudlet system is being deployed in a *Raspberry Pi* System-on-Chip (SoC) device. This Raspberry Pi is an on-board processing device, capable of performing simple and low-level computations.

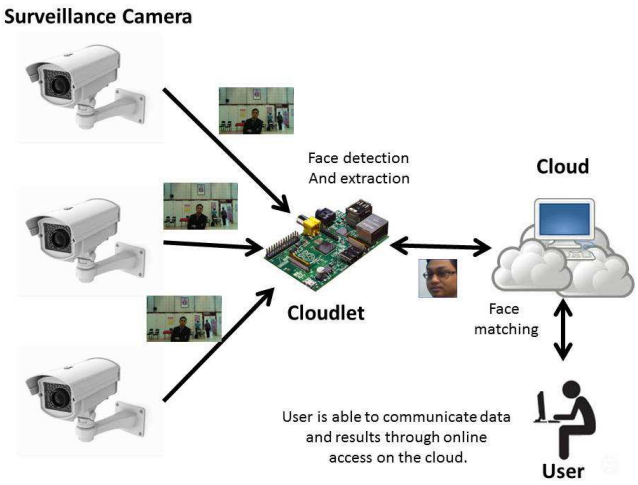

Fig. 2. Surveillance Camera-Cloudlet-Cloud infrastructure for distributed surveillance scheme.

IV. Decentralized Face Recognition Scheme

In our distributed surveillance scheme, face recognition is carried out in a decentralized manner, in which feature extraction and recognition are carried out in two separate environments, namely cloudlet and public or private cloud. As described in the previous section, cloudlet is responsible for extracting face and face vectors from a set of images captured using the video surveillance cameras, while the cloud system is responsible for matching and recognition of captured images with reference to the images in the database

In this section, we will describe both stages in details. This will include a description on feature extraction and matching algorithms being deployed at each stage.

A. Pre-Processing on the Cloudlet

In this decentralized face recognition scheme, cloudlets are responsible for feature extraction from the raw image data, prior to their transmission to the cloud. In normal IoT implementations, the raw images will be sent straight away to the central processing service that may reside in a private data center. As shown in Fig. 2, under this pre-processing stage, cloudlet are assigned to extract the face feature vector on the raw images. Only face feature vector will be transmitted to the cloud system for the matching process with the stored images in database.

The camera surveillance send the raw images to the cloudlet. Within the cloudlet, face vectors are extracted from these raw images. The process involves face feature identification and extraction.Within a single image, there is a possibility of more than one faces exist. Therefore, the proposed detection and extraction scheme must be able to fulfill such condition.

In this work, we performed face feature vector detection and extraction using the Haar feature-based cascade classifier [8]. This algorithm utilizes positive and negative images for training the classifier. Haar features, as shown in Fig. 3 are used to extract features from these images. Haar-like feature can be calculated as the difference of the sum of pixels of areas inside a rectangle, which can be at any position and scale within the original image.

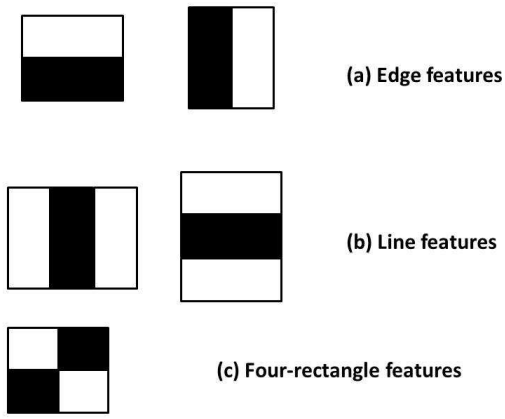

Fig. 3. Haar features used to classify positive and negative images.

The Haar feature pool computation used in this work has been inspired by the works of Papageorgiou et al. [9], and a fast computation scheme proposed by Viola et al. [6] and improved by Lienhart et al. [8]. In Haar feature representation, a rectangle of pixels, with top left corner (x, y), width w, height h and orientation $\alpha \in \{0°\}$. This rectangle is inside a window and specified by the tuple $r = (x, y, w, h, \alpha)$ with a pixel sum denoted by $Sum(r)$. The set of used features have the form:

$$f = \omega_1 \cdot Sum(r_1) + \omega_2 \cdot Sum(r_2) \quad (1)$$

where the weights $\omega_1, \omega_2 \in \mathbb{R}$ are used to compensate the difference in area size between the two rectangles r_1, and r_2.

The Haar features are used to classify positive and negative images. Nevertheless, the approach requires large amount of features to be applied on a single image window. Hence the use of cascade classifier on each stage of classification is introduced to minimize such amount of features. In this work, we use the Haar feature-based cascade classifier library provided by SimpleCV, a variation of OpenCV implementation for lightweight devices [4].

Fig. 4 shows a result of face extraction process from the raw image taken using a surveillance camera during an event.

The face feature vector obtained in this pre-processing stage are stored in an XML-structured tree format, as shown in the following code snippet:

```
<opencv_storage>
        <haarcascade_frontalface_tree_alt
        type_id="opencv-haar-classifier">
                <size>20 20</size>
                <stages>
</opencv_storage>
```

Fig. 4. Face feature detected and extracted from a video image.

One of the limitations of using Haar feature-based cascade classifier is such that it is not able to detect partial faces, i.e faces that have some parts of it being hidden or secluded. For instance, an image with a person standing in a sideway position.

B. Face matching on the cloud

The face matching process in the decentralized face recognition scheme is commonly being carried out in our private cloud environment. In certain cases involving tracking of person in different locations, this face matching process could also be carried out in a cloudlet.

The face matching process is conducted on face feature vector obtained through face detection and extraction as described in the previous subsection. We have implemented a local binary pattern histogram (LBPH) matching procedure, to match the incoming feature vector with existing vectors in the database which is resided on the private cloud. Our proposed LBPH recognition scheme follows the work that has been carried out by Ahonen et al. [1].

The LBP operator for this matching process follows a formal description in the following equation:

$$LBP(x_c, y_c) = \sum_{p=0}^{P-1} 2^P s(i_p - i_c) \tag{2}$$

given that (x_c, y_c) as central pixel with intensity i_c; and i_p being the intensity of the the neighbor pixel. s is the sign function defined as:

$$s(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{else} \end{cases} \tag{3}$$

LBP is able to capture very fine-grained details in images. Hence enables more accurate matching of face images. Nevertheless, an important key element in this algorithm is neighbourhood determination. As described in Ahonen et al. [2], the use of variable neighbourhood, i.e. to align an arbitrary number of neighbours on a circle with a variable radius as shown in Fig. 5.

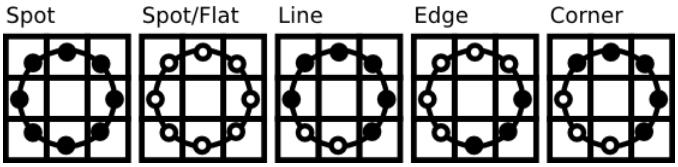

Fig. 5. Variable neighbourhood identification used in LBP algorithm (Adopted from [4]).

The LBP operator used in this face matching procedure implies the representation scheme proposed by Ahonen et al. [1], by dividing the LBP image into m local regions and extract a histogram from each of these regions. The spatially enhanced face feature vector is then obtained by concatenating these local histograms.

C. Virtualization

The face recognition, processing, and storage of the proposed scheme are carried out in a virtualized environment of cloud computing. Data (facial features) obtained from the IoT devices are sent to the cloud instance for recognition and storage. In this work, we have implemented our virtual instance on our private cloud, running on OpenStack cloud platform.

V. Results and Discussions

We have developed a prototype scheme for face recognition on video surveillance images that implements face extraction on cloudlet and matching on private cloud. This prototype has been designed for the application of finding "missing" person through the use of our surveillance camera. The database resided on the cloud is consists of train images of a particular "missing" person. These images are captured in three different positions, front, left, and right, as shown in Fig. 6.

Fig. 6. Stored images used for training purposes in face matching procedure.

Fig. 7 shows the scene images where the respective individual in Fig. 6 are found using the proposed decentralized face recognition scheme.

The prototype being developed has been integrated into our IoT-Cloud infrastructure known as *I-Awana*. *II-Awana* is a web-based system that interfaces between users and IoT-Cloud infrastructure for direct interaction. It allows users to

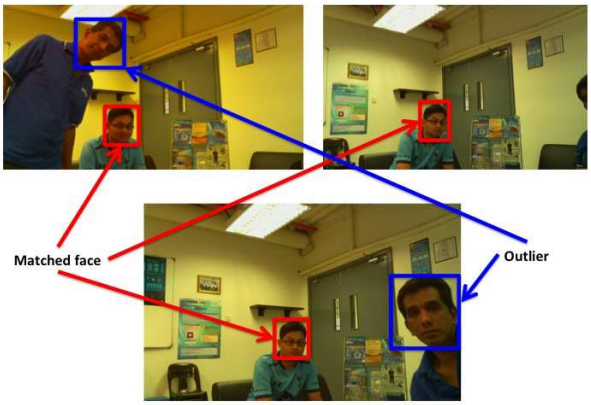

Fig. 7. Face matching process on video surveillance images. Note that the outlier is still detected. However, it does not match with the face image resided on the database.

upload images for surveillance purposes. For instance, user can use *I-Awana* to find missing person in the given surveillance areas covered by the IoT-Cloud infrastructure. Fig. 8 shows the interface of the developed system.

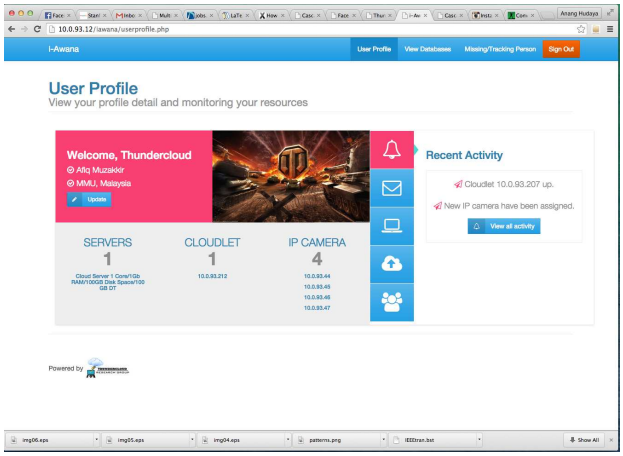

Fig. 8. *I-Awana* portal for user access to the distributed surveillance system.

Although at this prototyping stage, the face recognition scheme is able to extract faces and perform image matching in a decentralized manner, there are some limitations faced by the developed prototype in giving a full accuracy in detecting "missing" person. These include its inability to differentiate a "real" person with a portrait or photo within the video image. Fig. 9 shows how the system mistakenly detect a photo of a person inside the image as a "real" person. Hence, there are more works to be done in minimizing such errors. One possible way is to reexamine the neighbourhood determination function in LBP approach. More tests should be carried out in determing the reight neighbourhood sizes for more accurate results.

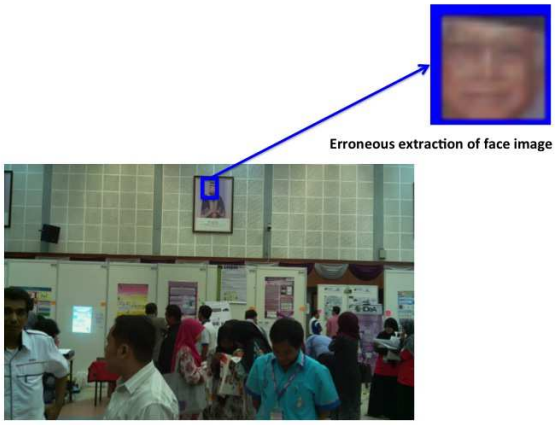

Fig. 9. A portrait being mistakenly extracted as face image.

VI. Conclusions

In this paper, we propose a decentralized face recognition scheme for video surveillance application on IoT-Cloud infrastructure. Our scheme minimizes the need for massive image transfers on low-bandwidth communication between surveillance cameras and data center in a common IoT systems. The recognition procedures are conducted in two stages: face feature detection and extraction, and face matching. The feature detection and extraction are performed in a cloudlet, while more complex face matching procedure is carried out on a private cloud. Preliminary results of the developed prototype indicate that the scheme is able to detect and match specific face image from a series of images captured using the video surveillance camera. Nevertheless, more investigations are still need to be carried out in order to improve its effectiveness and accuracy.

References

[1] T. Ahonen, A. Hadid, and M. Pietikäinen, "Face recognition with local binary patterns," in *Computer vision-eccv 2004*. Springer, 2004, pp. 469–481.

[2] T. Ahonen, A. Hadid, and M. Pietikainen, "Face description with local binary patterns: Application to face recognition," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 28, no. 12, pp. 2037–2041, 2006.

[3] A. M. M. Ali, N. M. Ahmad, and A. H. M. Amin, "Cloudlet-based cyber foraging framework for distributed video surveillance provisioning," in *Information and Communication Technologies (WICT), 2014 Fourth World Congress on*. IEEE, 2014, pp. 199–204.

[4] G. Bradski, "Opencv library," *Dr. Dobb's Journal of Software Tools*, 2000.

[5] K. Hong, D. Lillethun, U. Ramachandran, B. Ottenwälder, and B. Koldehofe, "Mobile fog: A programming model for large-scale applications on the internet of things," in *Proceedings of the second ACM SIGCOMM workshop on Mobile cloud computing*. ACM, 2013, pp. 15–20.

[6] P. Jones, P. Viola, and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *University of Rochester. Charles Rich*. Citeseer, 2001.

[7] R. Kemp, N. Palmer, T. Kielmann, F. Seinstra, N. Drost, J. Maassen, and H. Bal, "eyedentify: Multimedia cyber foraging from a smartphone," in *Multimedia, 2009. ISM'09. 11th IEEE International Symposium on*. IEEE, 2009, pp. 392–399.

[8] R. Lienhart and J. Maydt, "An extended set of haar-like features for rapid object detection," in *Image Processing. 2002. Proceedings. 2002 International Conference on*, vol. 1. IEEE, 2002, pp. I–900.

[9] C. P. Papageorgiou, M. Oren, and T. Poggio, "A general framework for object detection," in *Computer vision, 1998. sixth international conference on*. IEEE, 1998, pp. 555–562.

[10] M. Satyanarayanan, "Pervasive computing: Vision and challenges," *Personal Communications, IEEE*, vol. 8, no. 4, pp. 10–17, 2001.

[11] T. Soyata, R. Muraleedharan, C. Funai, M. Kwon, and W. Heinzelman, "Cloud-vision: Real-time face recognition using a mobile-cloudlet-cloud acceleration architecture," in *Computers and Communications (ISCC), 2012 IEEE Symposium on*. IEEE, 2012, pp. 000 059–000 066.

[12] F. Tao, Y. Zuo, L. Da Xu, and L. Zhang, "Iot-based intelligent perception and access of manufacturing resource toward cloud manufacturing," *Industrial Informatics, IEEE Transactions on*, vol. 10, no. 2, pp. 1547–1557, 2014.

[13] Y. Xiao, P. Simoens, P. Pillai, K. Ha, and M. Satyanarayanan, "Lowering the barriers to large-scale mobile crowdsensing," in *Proceedings of the 14th Workshop on Mobile Computing Systems and Applications*. ACM, 2013, p. 9.