TELKOMNIKA Telecommunication, Computing, Electronics and Control
Vol. 18, No. 3, June 2020, pp. 1671~1677
ISSN: 1693-6930, accredited First Grade by Kemenristekdikti, Decree No: 21/E/KPT/2018
DOI: 10.12928/TELKOMNIKA.v18i3.14787

Deep hypersphere embedding for real-time face recognition

Ryann Alimuin, Elmer Dadios, Jonathan Dayao, Shearyl Arenas
Technological Institute of the Philippines-Quezon City, De La Salle University, Philippines

Article Info

Article history:

Received Aug 31, 2019
Revised Dec 7, 2019
Accepted Dec 12, 2019

Keywords:

Deep hypersphere embedding
Real-time face detection
Real-time face recognition
Robotic visions
Security surveillance system

ABSTRACT

With the advancement of human-computer interaction capabilities of robots, computer vision surveillance systems involving security yields a large impact in the research industry by helping in digitalization of certain security processes. Recognizing a face in the computer vision involves identification and classification of which faces belongs to the same person by means of comparing face embedding vectors. In an organization that has a large and diverse labelled dataset on a large number of epoch, oftentimes, creates a training difficulties involving incompatibility in different versions of face embedding that leads to poor face recognition accuracy. In this paper, we will design and implement robotic vision security surveillance system incorporating hybrid combination of MTCNN for face detection, and FaceNet as the unified embedding for face recognition and clustering.

Corresponding Author:

Ryann Alimuin,
Technological Institute of the Philippines-Quezon City,
De La Salle University, Manila, Philippines.
Email: ryann.alimuin@tip.edu.ph

1. INTRODUCTION

With the emerging development of robots capable of human-computer interaction, recognizing people in computer vision and pattern recognition has attracted immense attention as it provides huge applications in terms of finance, military, public security and daily life. Among various biometrics used for person recognition, the face is one of the most popular, since this ubiquitous biometric can be acquired in unconstrained environments while providing strong discriminative features for recognition [1]. Over the years, there are many breakthroughs that contributed to the success of face recognition technology. This is with the help of advanced network architectures [2-5], discriminative approach [2]. Face recognition begins with extracting the coordinates of features such as width of mouth, width of eyes, pupil, and compare the result with the measurements stored in the database and return the closest record (facial metrics) [3]. There have been a huge number of research on ways of improving the local descriptors, feature transformations and pre-processing in face recognition such as linear subspace [4], in manifolds [5, 6], and sparse representation [5]. But these approaches targets only an aspect of constraints in facial feature and improved face recognition accuracy slowly [1]. Furthermore, challenges in terms of illumination, expression and pose are the three most known problems in face recognition (FR).

In the recent years, research landscape in face recognition significantly reshaped into the breakthrough of deep learning such as deepface method. Deep learning applies multiple processing layers to learn representations of data with multiple levels of feature extraction [7]. The most popular deep learning architecture is the convolutional neural network (CNN) that combats significant problems in computer visions such as image classification, segmentation, object detection, etc. [8]. Many face recognition applications seek a desirable low-dimensional representation that generalizes well to new faces that the neural network wasn't trained on but the representation is a consequence of training a network for high-accuracy classification on their training data.

One of the challenges of this kind of approach is that the representation is difficult to use because faces of the same person aren't necessarily clustered, that the classification algorithms can take advantage of [9].

This paper discussed about the computer vision of robots involving face recognition process incorporating FaceNet as the unified embedding for face recognition and clustering that learns how to cluster representations of the same person and that can alleviate training difficulties that can significantly improve FR accuracy utilizing Python as the programming language for the surveillance system. In this paper, we proposed a system and a method for target identification using artificial neural networks integrated in robotic vision. The contributions of this paper summarizes as follows:

a. We present a security surveillance system that authenticates a person in the robotic camera.
b. A method to provide an equivalent virtual instrument that has the same capability and functionality that contains the following: (1) a digital filter that is used for image processing (2) a machine learning algorithm that uses artificial neural networks by means of face vector identification for target identification.
c. We utilizes a method having a unified face image representation necessary for better recognition of face images.
d. The system that can be adapted to any existing surveillance systems, provides low cost memory storage, has data logging features and low maintenance.

2. RESEARCH METHOD

The input of the system will came from the wireless camera embedded in Robots feeds that will be processed and examined by the system. Once a face image is detected in the camera feeds, then it will decide whether the face detected is recognized or not, if the face is recognized then the system logs the date, time and the camera number otherwise the system still logs the date, time, and activates the alarm and notification system.

Algorithm 1. Algorithm of the system

```
                         Algorithm of the system
Input: Camera's real-time image data $I_{data}$
Step 1: While $I_{data} \neq$ face_image
Step 2:       Process Video feeds
Step 3: If $I_{data} =$ face_image, then
Step 4:     If $I_{data} =$ face_authorized, then
Step 5:           System logs date, time and camera number
Step 6:     Else
Step 7:           System logs date, time and camera number
Step 8:           Alarm and Notification is activated
Step 9:     End if
Step 10:End if
```

2.1. Signal conditioning

Figure 1 shows the general overview of the proposed system. To analyze, measure and manipulate data feeds from camera footage, analog signals should be converted into digital signal utilizing the theory of digital signal processing. Analog to digital converter (ADC) is the one responsible in sampling, quantizing and encoding the continuous-amplitude analog signal into discrete time and amplitude digital signal.

$$\boldsymbol{p(t) = \sum_{k=-\infty}^{\infty}[u_s(t-kT) - u_s(t-kT-p)]} \tag{1}$$

A number of variable bit-rate data streams of input signals from different wireless cameras will be integrated into a constant capacity signal through time division multiplexer (TDM) used for a higher bit-rate flow of data [10]. Subsequently, the signals were being digitally filtered through digital signal processing (DSP) to process the image for the integration of face detection and data logging technology using artificial neural networking.

2.2. Image processing

Multithreading and GPU based processing technologies were used to perform the image processing. Architecture of the image processing in this research is shown in Figure 2. Detailed processing will be explained in the below section.

2.2.1. Multi-task cascaded CNN

In face detection phase, our method is based on multi-task cascaded CNN used for joint face detection and face alignment [11] in detecting faces within the vicinity of camera footage in real-time. It initially resizes the images into a different scale building an image pyramid. The process came with 3 stages MTCNN namely: proposal network (PNet) used to obtain candidate facial windows, as well as their bounding box regression vectors. Refine network (RNet) that refines huge amount of false candidate as well as performs calibration with bounding box regression, and conducts NMSa and lastly, output network (O-Net) that is used to produce the final huge box and facial landmarks position, respectively [12]. This stage aims to identify face regions with more supervision Additionally, MTCNN uses a complex algorithm in multiple threads wherein it can detect faces effectively even in ranges of distance from the camera that makes it a good fit for our application.

a. Training

In training for the CNN detector, it leverages three tasks as follows.

- Face Classification. It utilizes the cross-entropy loss in each samplex_i.

$$L_i^{det} = -(y_i^{det}\log(p_i) + (1 - y_i^{det})(1 - log(p_i))) \tag{2}$$

- Bounding box regression. The learning objective is formulated as a regression problem and the method utilizes the Euclidean loss for each sample x_i.

$$L_i^{box} = \left\|\hat{y}_i^{box} - y_i^{box}\right\|_2^2 \tag{3}$$

- Facial landmark localization. The same with the bounding box, the facial landmark detection is formulated as a regression problem and utilizes minimizes the Euclidian loss.

$$L_i^{landmark} = \left\|\hat{y}_i^{landmark} - y_i^{landmark}\right\|_2^2 \tag{4}$$

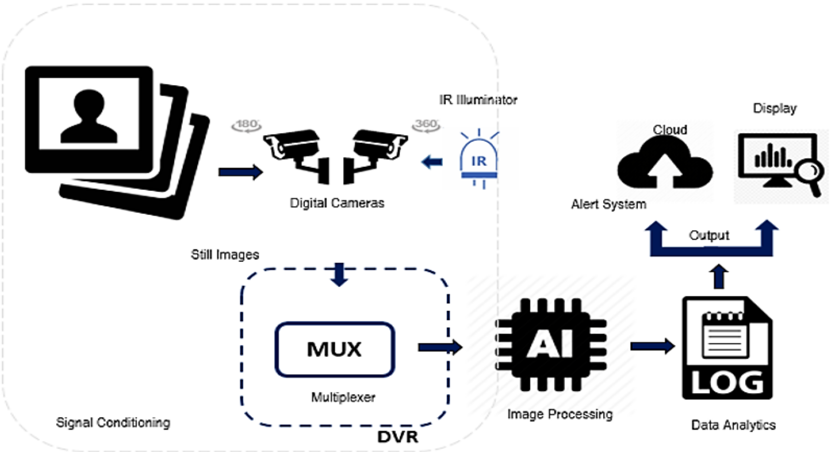


Figure 1. General overview of the system

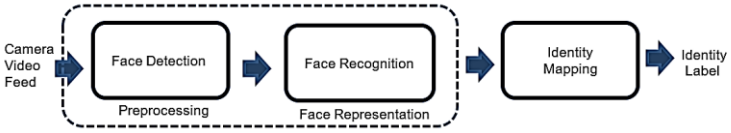


Figure 2. Image processing architecture

2.2.2. Face recognition

The above-mentioned problem involving invariance of face representation over a period of time, can be solved by the notions of finding packing asymptotic bounds, that are not overlapping, for which it can be fit within a face representation space or hypersphere. A representation of the geometrical structure can be can be describe wherein the lower bound represents the low-dimensional population manifold embedded in a high dimensional space located on the upper bound hyper-ellipsoid that is clustered into their own class specific hyper-ellipsoids [13]. The invariance of the face representation is determined by the number of identities that is packed per hyper-ellipsoid.

a. FaceNet

In this paper, we integrated a method called FaceNet for the face recognition phase. FaceNet is a unified embedding for face recognition and clustering that directly learns a mapping from face images to a compact Euclidean space where distances directly correspond to a measure of face similarity [14] using the triplet loss function based on LMNN.

b. Triplet loss function

The system used FaceNet to map face features from the input images taken from the cameras into a 512-dimensional Euclidian space vector [15]. The embedding is represented by (5), having an input image x embedded into d-dimensional Euclidian space vector $\mathbb{R}^d$and is constraints in (6) [11].

$$f(x) \in \mathbb{R}^d \tag{5}$$

$$||f(x)||_2 = 1 \tag{6}$$

Furthermore, the system utilizes triplet loss used to effectively cluster face vector embedding between the same classes, and is represented by (7) [16].

$$\sum_i^n [||f(x_i^a) - f(x_i^p)||_2^2 - ||f(x_i^a) - f(x_i^n)||_2^2 + a] \tag{7}$$

where x_i^a represents an anchor of a specific person, x_i^p indicates positive representation of the same person. x_i^n is the negative representation of any other person and a being the margin between the positive and negative pairs. The triplet loss minimizes the square of the distance between the anchor and a positive while maximizing the square of the distance between the anchor and negative pairs [17].

c. Harmonic embedding and triplet loss

The system also provides a powerful function wherein it has the capability to compare a new training dataset to the existing datasets in the gallery. This function is ideal for large scale datasets that is divergent, and requires retraining the subject multiple times.

2.3. Serial communication

A USB to serial adapter also referred to as a USB serial converter or RS232 adapter was used for serial communication as the interface from the camera into the computer [18]. It is a small electronic device which can convert a USB signal to serial RS232 data signals [19]. It is the type of signal which is in many older PCs and is referred to as a serial COM port. A USB to serial adapter typically converts between USB and either RS232, RS485, RS422 or TCP signals, however some USB to serial adapters have other special conversion features such as custom baud rates, high-speed or other [20, 21].

3. IMPLEMENTATION

The following are the components of the whole system

a. Wireless camera
b. NVR
c. USB serial converter or RS232
d. Laptop Computer

This research uses camera specifications as shown in Table 1, and uses a computer as shown in Table 2.

Table 1. Camera specifications

Specifications	Value
Image Sensor	½.8'2.4 MP CMOS
Effective Pixels	1984 (H) x 1225 (V)
Electronic Shutter	1/3s – 1/100,000s
Minimum Illumination	0.05 lux/F1.4, lux IR on

Table 2. Laptop computer specs

Specifications	Value
Processor	At least 4GHz
Operating System	Windows XP or later
Internal Storage	Minimum of 1TB
Random Access Memory (RAM)	At least 8 GB

4. TECHNICAL RESULT

In this section we will evaluate the effectiveness and the performance of the proposed system.

4.1. Graphic user interface (GUI)

The user interface of the system indicates the portion where the 4 cameras will be shown. It also displays preview, database, cloud storage, local storage, about and lastly, data logging. In the section of data logging, it displays the detection of the cameras where the recognized faces are authorized or unauthorized. It also indicates the identity of the detected person. Sample 12x12 pixel of face datasets as shown in Figure 3.

Figure 3. Sample 12x12 pixel of face datasets

4.2. Training Data

The training set used in the experiment are face images taken from sample specimens, where facial features will be taken multiple times. To minimize face variations, each will be taken without expression, and then asked to tilt their faces to the left and slowly to the right, and move their faces slowly upward and downward position. The result of the training set acquisition process will produce 200 sets of 12x12 pixelated face images for the training set per person and additional 10 images for the samples necessary for the testing.

4.3. Detection, Extraction ad Recognition time

Detection is the process wherein the system searches for the faces within an image and returns its coordinates [22]. Extraction, on the other hand is the process where the system filters the detected face to filter out the unnecessary details of the image [23]. Lastly, recognition is the process of the system where it identifies the detected face based from the datasets [24]. After 10 iterations, we were able to measure the detection, extractions and recognition time. The average detection time is 100.8ms, extraction time is 91.7 ms and recognition time is 0/8 ms.

4.4. Percent accuracy per person

Figure 4 shows the result of the accuracy of the system wherein 5 different people were tested one at a time. The system shows highly accurate classification having an average of 86%.

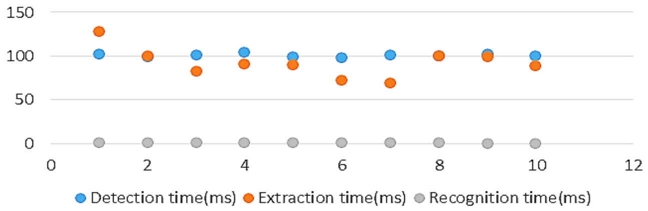


Figure 4. Detection, extraction and recognition time (ms)

4.5. Evaluation of the systems performance

We tested the system's performance and limitations by varying distance of the subjects from the camera that ranges from 1-7ft incremented with 1foot and at the same time varying the number of people being recognized simultaneously. Table 3 shows the numerical value of the accuracy. The experiment shows that adding the number of subjects being recognized by the system simultaneously can greatly affect the performance of the system while increasing the distance of the subject from the camera creates a minimal effect to the performance of the system. Figure 5 shows the average accuracy of 50% from the multiple faces with varied distance.

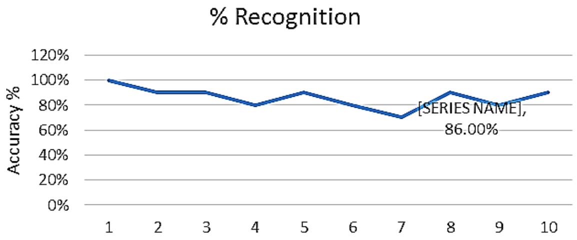


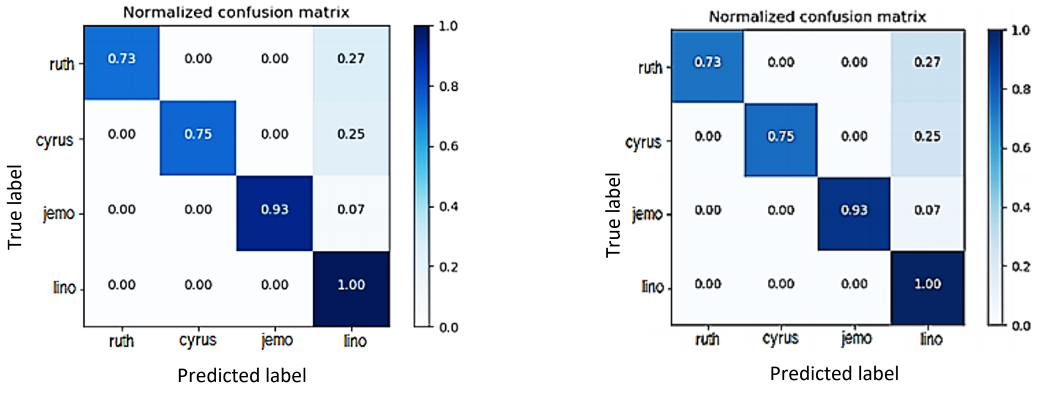
Figure 5. Accuracy per person (ms)

Table 3. Accuracy testing

Distance (ft)	1 Person	2 Persons	3 Persons	4 Persons	5 Persons	Average
1	100	27	44.44	50	40	52.44
2	90	50	44.44	25	40	49.88
3	90	40	55.55	25	40	50.11
4	80	50	22.22	25	40	43.44
5	90	60	33.33	50	50	56.66
6	80	60	44.44	37.5	40	52.38
7	70	40	22.22	37.5	40	41.94

4.6. Confusion matrix

To evaluate the performance of the classifier, confusion matrix was used. It shows a visualization in which the classifier is confused when making a prediction in dealing with adding training subjects to the data set. Figure 6 shows the result of classifier's true positive rate and misclassification rate by means of dividing temporarily the dataset which is composed of 33% test sets and 67% of train sets [25]. It shows normalized confusion matrix from 3 and 4 persons known, respectively.



(a)

(b)

Figure 6. Confusion matrices with, (a) 3 persons, (b) 4 persons (normalized)

4.7. Comparisons of the systems performance with other deep learning algorithm

Varying the algorithm on the system provides minimal variations to the performance of the system. Figure 7 shows the system's performance in terms of its accuracy, sensitivity and specificity using the DeepFace, SphereFace and MTCNN. The MTCNN and FaceNet adapts to the systems performance by having a stand out result in compared with the two other algorithm.

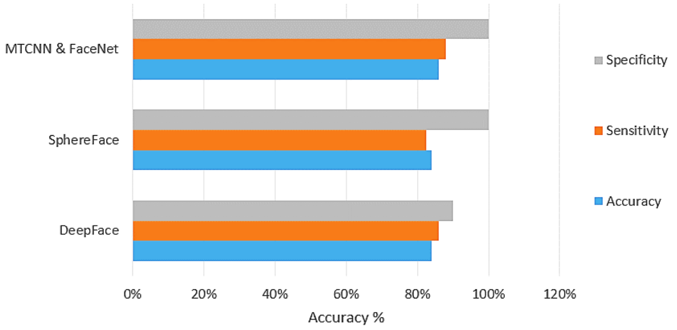


Figure 7. System's performance test with other deep learning algorithm

5. CONCLUSION

Identifying a person from a surveillance system embedded in robots offers significant advantages in terms of security in compared with the traditional surveillance system. It can save huge amount storage and its corresponding costs by only storing frames of face images that was detected by the system. It offers more secure environment, as it will send alerts regarding burglary that is happening in real time. It serves as biometrics of a person's identity logging in and out from an establishment and can be very useful in locating and identifying criminals around the city. The experiments uncover the system's limitation in detecting and identifying multiple person at a specified distance simultaneously. The system resulted to only 50% of the average accuracy when dealing with multiple person in compared with 86% in accurately identifying different faces at a time. For future designers who wishes to venture into this study. We highly recommend to further improve the performance of the system's accuracy by trying other types of algorithm knowing that there are a lot of options. We also recommend to use better specifications of camera and computer mentioned in section 3.

ACKNOWLEDGMENT

The proponents would like to thank De La Salle University-Manila and Technological Institute of the Philippines-Quezon City for the research collaboration and to the following students namely R. Enorme, L. Estrada, C. Montealto, J. Oreta.

REFERENCES

[1] I. Masi, et al., "Deep Face Recognition: a Survey," *2018 31st SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, pp. 471-478, 2018.

[2] Y. Sun, et al., "Deep learning face representation by joint identification-verification," *Advances in Neural Information Processing System 27 (NIPS 2014)*, pp. 1988-1996, 2014.

[3] A. R. S. Siswanto, et al., "Implementation of face recognition algorithm for biometrics-based time attendance system," *2014 International Conference on ICT for Smart Society (ICISS)*, pp. 149-154, 2014.

[4] W. Deng, et al., "Transform-Invariant PCA: A Unified Approach to Fully Automatic FaceAlignment, Representation, and Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 6, pp. 1275-1284, June 2014.

[5] J. Wright, et al., "Robust Face via Sparse Representation," *IEEE Transaction on Pattern Analysis and Machine Intelligent*, vol. 31, no. 2, pp. 210-227, February 2009.

[6] E. Klarreich, "Sphere Packing Solved in Higher Dimensions," *Quanta Magazine*, March 2016. [Online]. Available: https://www.quantamagazine.org/sphere-packing-solved-in-higher-dimensions-20160330/.

[7] M. Wang and W. Deng, "Deep Face Recognition: A Survey," arXiv: 1804.06655, 2019.

[8] W. Liu, et al., "Deep Hyperspherical Learning," arXiv: 1711.03189, 2018.

[9] B. Amos, et al., "OpenFace: A general-purpose face recognition library with mobile applications," CMU-CS-16-118, 2016. Available: http://elijah.cs.cmu.edu/DOCS/CMU-CS-16-118.pdf.

[10] M. Plonus, "CHAPTER 9-Digital Systems," *Electronics and Communications for Scientists and Engineers*, Academic Press, pp. 327-403, 2001.

[11] K. Zhang, et al., "Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks," *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1499-1503, Oct 2016.

[12] Programmer Sought, "Face key point detection algorithm-MTCNN," 2018. [Online]. Available: http://www.programmersought.com/article/90001188428/.

[13] S. Gong, et al., "On the Capacity of Face Representation," arXiv: 1709.10433, 2019.

[14] F. Schroff, et al., "FaceNet: A unified embedding for face recognition and clustering," *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 815-823, 2015.

[15] M. Wang and W. Deng, "Deep Face Recognition: A Survey," 2018. [Online]. Available: https: //www.researchgate.net/publication/324600003_Deep_Face_Recognition_A_Survey.

[16] F. Schroff, et al., "FaceNet: A Unified Embedding for Face Recognition and Clustering," arXiv: 1503.03832, 2015.

[17] T. T. Do, et al., "A Theoretically Sound Upper Bound on the Triplet Loss for Improving the Efficiency of Deep Distance Metric Learning," *CVPR 2019*, April 2019.

[18] R. OBrien, "How Does a USB to Serial Adapter Work?" 2019. [Online]. Available: https://itstillworks.com/usb-serial-adapter-work-4969162.html.

[19] Wikipedia, "RS-232," Available: https://en.wikipedia.org/wiki/RS-232.

[20] R. Alimuin, et al., "Design of a 4-Channel Virtual Instrument Video Adapter for Digital Data Multiplexing," *2018 IEEE 10th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM)*, pp. 1-5, 2018.

[21] Treichler J. and Larimore, "Theory and design of adaptive filters," Prentice-Hall of India, 2007.

[22] A. Sinha, "How To Detect and Extract Faces from an Image with OpenCV and Python," DigitalOcean, 2019. [Online]. Available: https://www.digitalocean.com/community/tutorials/how-to-detect-and-extract-faces-from-an-image-with-opencv-and-python.

[23] N. J. Pyun, "Extraction of an image in order to apply face recognition methods," *Artificial Intelligence*, Université Sorbonne Paris Cité, 2015.

[24] J. Brownlee, "A Gentle Introduction to Deep Learning for Face Recognition," Machine Learning Mastery, 2019. Available: https://machinelearningmastery.com/introduction-to-deep-learning-for-face-recognition/.

[25] K. V. Arya and A. Adarsh, "An Efficient Face Detection and Recognition Method for Surveillance," *2015 International Conference on Computational Intelligence and Communication Networks (CICN)*, pp. 262-267, 2015.