Identity Verification through Face Recognition Implemented on Raspberry Pi Framework

Alvin Jason A. Virata[1] and Enrique D. Festijo[2]

[1,2]Graduate Programs, Technological Institute of the Philippines, Manila, Philippines

***Abstract*:** The influence of identity verification in mobile application have attracted significant attention and growth to mobile technology development. So far, there are various methods involving identity verification in mobile application development, yet, it was still considered challenging and have been a primary study due to its limitation to memory allocation and the computing power. Subsequently, to address these issues there are different algorithms that were designed for identity verification to adopt the mobile environment and to resolve the arising challenges. In this paper, we proposed a novel, cost-effective and energy-efficient framework by introducing a mobile-based identity verification on offline-mode using the Raspberry Pi framework. A Raspberry Pi device is wirelessly connected to mobile phone to process the face detection and face verification. The proposed method is implemented on the latest version of Raspberry Pi 3 model B+ version run in Python 3.7 where the datasets and training sets images were experimented and tested using LBP algorithms for face detection and face verification. With the experimental test result using the confusion matrix in a multiclass classification, the proposed method showed a results of 87.5% accuracy score, 88% in terms of precision, 88% recall score and 86% F1 score. In addition, the experimental test were done using 3000 images in a controlled/unconstrained environment, were 20% or 600 of the images were used as data sets. During the offline mode testing, the face detection and verification has resulted to an average timing of 4.98 seconds. Thus, it concludes the feasibility to implement face verification system in the Philippine government services such as the voting system, road check-point, driver's ID verification to name a few.

***Keywords*:** Raspberry Pi, Local Binary Pattern (LBP), Face Detection, Face Verification, Mobile Application, Identity Verification

1. Introduction

Nowadays, the strong influence of mobile phones plays a significant role to the development of biometrics, particularly in identity verification as part of the security. Not leaving behind the fast-paced development of embedded devices like Raspberry Pi it also helps to further expand the application of mobile technology. Noticeably, mobile phones are progressively becoming into a fundamental piece of human life as it plays strong contribution to an effective and convenient communication [1].

Face recognition is turning into a multi-disciplinary research agenda and it has been used to a variety of applications in the field of security, for instance identification and verification of an individual face in a video or image frames. Identifying or recognizing individual face is exactly a unique way to implement strong security. However, with regards to the implementation using computers, mobile phones or any other devices will be very challenging as there are several parameters to be considered and calculated to generate precise result on face recognition or verification [2].

Commonly, the method of digital processing and analyzing of an individual facial structure were identified by measuring the points of a certain parts of the face features distances such as the eyes, nose mouth, jaw, forehead and other part of the face, at the same time considering the face angles [7]. Thus, face detection happens when the process of locating faces in a given domain using different algorithms is applied in the design environment [8], while face recognition is mainly used for the tasks of identification or verification. Face identification is used to confirm the identity of an unknown individual and identification happens when a face data of an individual person was compared using the face or image data of various individual. However, with regards to verification the identified face data of an individual will be guaranteed genuine when the image match to the individual features or attributes. Hence, face recognition is performed in a wide range of scope based on facial features, emerging technologies and algorithms [9], [10]

Even though that mobile computing (MC) shows fast paced progress and having been recognized for its powerful trend in the community IT Technology environment development especially in the fields of commerce, still, mobile devices cannot hide the facts the many challenges in its resources in terms of battery life, storage and bandwidth that greatly affect the mobility and security in relation to communication effectiveness. This limitation or weaknesses, however, a primary dilemma on service qualities [1]. As mentioned, mobile devices being battery dependent have limited its capacity to perform computationally intensive application, as well as storage demanding processes. This obstacles have hindered the mobile application capability in which demands to consider cloud application to resolve the issue [4], [6], [5]. In spite of the challenges on mobile devices, the wide range application on mobile object detection systems have been consistently progressing due to demand on the criteria of portability [7], [12]. In which one of the examples of mobile object detection was implemented as an assistive application systems that is intended to benefit a persons with disability [12], [7]. The SmartVision prototype [16] this mobile-based assistive application helps the blind person to navigate to its desired location as it provides direction through the use of global positioning system, computer vision and geographical information for the object obstacle and path detection.

2. System Design and Development

2.1 System Configuration

In this study, the mobile phone is wirelessly connected to Raspberry Pi device which was used to store the trained data sets to process the face detection and verification. The Raspberry Pi device used has the following specification and configuration and hardware details presented in Table 1.

Table 1. Raspberry Pi Specifications for the Proposed Framework

Name	Configuration
Processor	Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC @ 1.4GHz
RAM	1GB LPDDR2 SDRAM
Connectivity	2.4GHz and 5GHz IEEE 802.11.b/g/n/ac wireless LAN, Bluetooth 4.2, BLE
Power	5V/2.5A DC power input

The Raspberry Pi has a Broadcom BCM2837B0 system on a chip which includes a Cortex-A53 (ARMv8) 64-bit SoC, with 1.4GHz 64-bit quad-core processor, dual-band wireless LAN, Bluetooth 4.2/BLE, faster Ethernet, and Power-over-Ethernet support (with separate PoE HAT). Also, to facilitate the offline face detection and face verification the training of data was executed in an ordinary laptop with the following hardware and software specifications shown in Table 2.

Table 2. Hardware and software specification

Name	Configuration
Display	AMD Radeon R7 Graphics
Processor	AMD A12-9720P Radeon R7, 12 Compute Cores 4C+8G 2.70GHz
RAM	8 GB (6.97 GB Usable)
System	64-bit Operating System, x64-based processor

During the training process, most researches recommend to use the Graphical Processing Unit (GPU) to improve the process of training datasets. However, this has become a breakthrough in this study that improve the performance of face detection and verification without primary using the GPU but instead re-configuring the memory allocation by maximizing a page file that will help the primary storage memory to maximize the processing of training data sets.

2.2 Face Images Collection

During the face images collection, the proponents administer the following:

- Provide a letter of consent in compliance with the Philippine Republic Act 10173- Data Privacy Act of 2012
- The collected images were participated by different people who were students, employees from private and NGO sectors and religious group.
- In the actual collection of images (3024x4032) pixel, the proponents considered the controlled/unconstrained environment.

2.3 Experimental Test – Setup

As presented in Figure 1 is the system framework of the proposed project. In the framework, it showed how the experimental test setup was conducted to validate the prediction accuracy, the timing as well as the training speed. The proponents gathered 3000 images to be used as sampling of the proposed project. Out of the 3000 images collected 80% of the images or 2400 images of the data were used to be the training set while the remaining 20% or 600 images was used as test data.

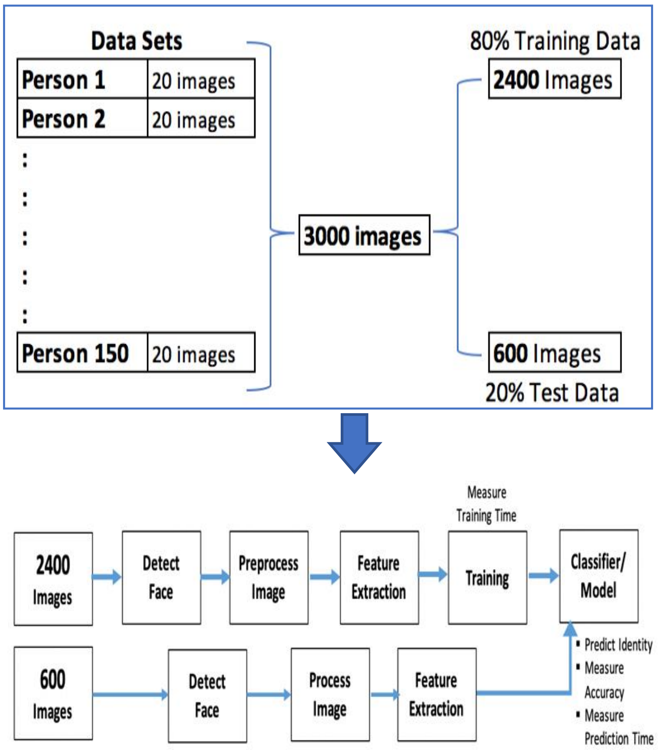

Figure 1. System Framework

During the testing process, the proponents aims to measure the prediction(sensitivity), precision(prediction accuracy), recall (sensitivity of actual prediction), and the F1 score (prediction consistency). Approximately, the time consumed in preparing the training set with 2400 uncompressed images took about 2 hours and 11 minutes or equivalent to 3.28 seconds per image tested in the machine with the following specifications: AMD A12-9720P Radeon R7, 12 Compute Cores 4C+8G 2.70GHz, 8 GB (6.97 GB Usable), 64-bit Operating System, x64-based processor.

2.4 System Architecture

Illustrated in Figure 2, the system architecture of the project that describe the process of identity verification. Whereby, using a mobile phone an image will be captured to process the face detection, and face extraction in the Raspberry Pi device since the classifier/model were already installed. The classifier that processed the identity verification were established using the Local Binary Pattern (LBP) Algorithm. Also, the application of LBP in the recent study was found successful in terms of face authentication and face detection recognition [17] . Hence, in spite of the existing challenges mentioned pertaining to the implementation of face recognition in mobile application development, the proponents opt to pursue the proposed project to address the challenges on the limitation to memory allocation and computation power by adopting the offline identity verification. The offline identity verification was made possible since the process of preparing the training set were done separately using a desktop computer.

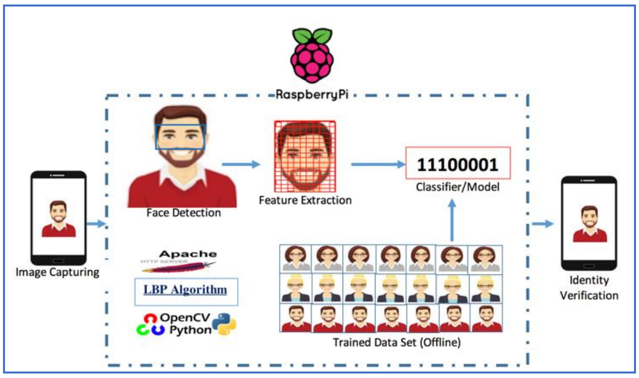

Figure 2. System Architecture

2.5 System Output

To test the proposed project, a prototype was created. There are four (4) menus that can be selected from the prototype, these are (1) capture menu, where image can be collected or gathered; (2) detect menu, this will compare the image from the taken photos; (3) Identify, is an offline mode process of verifying the person's identity; and lastly (4) person's information menu, a database where the person's information are registered.

Figure 3. System Prototype

Shown in figure 3 is the Graphical User Interface (GUI) system prototype that is used during the testing process. The system has initial stored data of the person's basic information that is essential for verification of the person's identification.

3. System Evaluation Results

Using the collected 3000 controlled images, the proponents come up with different methodologies to test the algorithm being used, specifically, local binary pattern (LBP). First, the proponents used a compressed image as data set and training set. However, during the process of testing the impact of accuracy, prediction and timing the result is not successful. It was observed that the more the images were compressed the algorithm have difficulty to show accuracy on face recognition. However, the testing process became successful when the original resolution of the image (3024x4032 pixel) is used as data set and training set. The challenge on timing prediction was also concluded with remarkable result even though a wireless communication were used between the mobile phone and raspberry pi device.

As presented in table 3, the summary results of confusion matrix in a multiscale described the prediction accuracy in terms of precision, recall and F1 score. Hence, based on the data presented it shows significant results with 12% error rate.

Table 3. Confusion Matrix Summary Result

Description	Result
Precision	88%
Recall	88%
F1 Score	86%
Accuracy	87.5%
Error Rate	12%

Further, the data was also tested using a rapid miner to compare the result using different model. As presented in figure 4 below using Generalize Linear Model (GLM) as simulator, it shows significant relationship of the prediction result using confusion matrix.

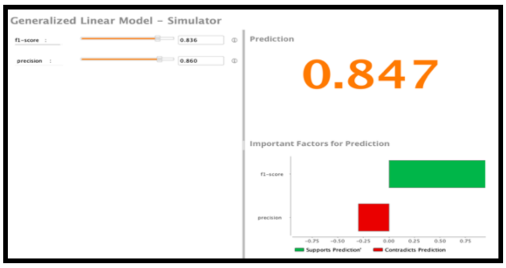

Figure 4. Prediction Simulator Result using GL Model

Illustrated in figure 5 the comparative correlation of the precision recall and f1-score. In the visualization it presented the contrast between the model-specific weights, showing which columns have in general most influence on the predictions for each specific model. Thus, the attributes described positive outcome result in the area of f1-score, where, an f1-score measures the accuracy of the precision and recall result score.

Attributes	f1-score	precision	recall
f1-score	1	0.894	0.940
precision	0.894	1	0.716
recall	0.940	0.716	1

Figure 5. Prediction Correlation Results of f1-score, precision and recall.

As shown in Figure 6, the result of precision distribution that describes the number of correct predicted images during the testing process. In the statistical summary, it shows that the average value of prediction is 87.6%.

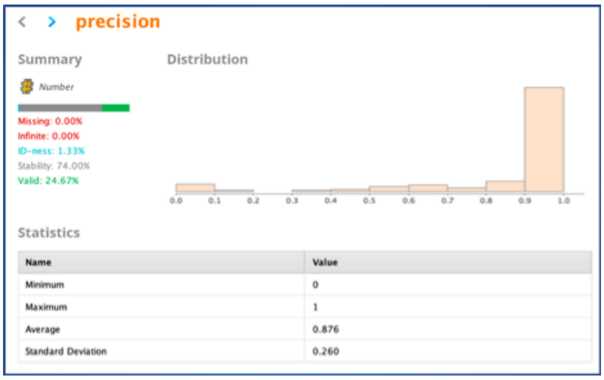

Figure 6. Precision Distribution Result

Considering figure 7, the recall distribution result average also showed 87.5% in which it illustrates the sensitivity of the actual prediction. Based on the visualization, it described that the actual prediction is high in relation to a true positive rate.

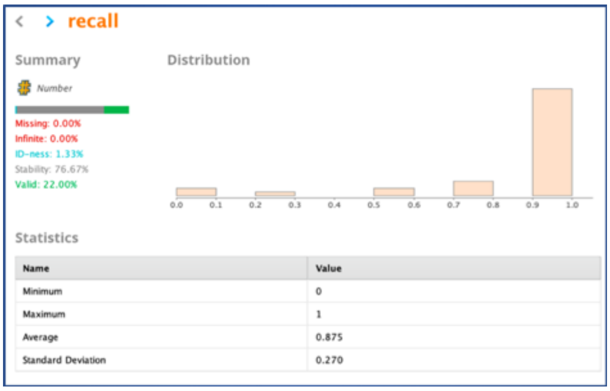

Figure 7. Recall Distribution Result

Lastly, the impact of F1 score is significant to the study as it measures the accuracy of the precision and recall result. Based on the illustration shown in figure 8, it proves the prediction consistency between precision and recall result as validated in an f1-score with an average percentage of 86.2%

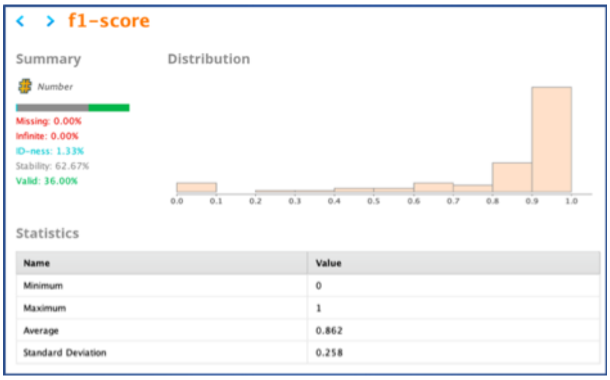

Figure 8. F1 Score Average and Summary of Distribution

4. Conclusions

In the study to implement identity verification through face recognition using raspberry pi as an embedded device, the following were concluded. The objective to address the issue on utilization in terms of limitation to storage and computation power consumption was addressed by embedding raspberry pi device to process face detection and face verification in an offline mode. In addition, the success of using LBP algorithm was noted for face detection and face verification accuracy due to uniform binary patters as LBP patterns contains two bitwise transitions from 0 to 1 or 1 to 0 in their binary notation. Hence, the higher the resolutions of the face images used in the training sets is an advantage to produce a better prediction results of face verification. Further, considering the vantage point positions when taking images is necessary, if it is intended to be used as part of the training sets. In addition, during the testing process the prediction timing speed has a relationship on the network frequency availability. Overall, the minimum requirements to evaluate the face verification through face recognition in terms of prediction, precision, recall and F1 score have exceeded more than 50% of the correct predicted images vis-à-vis with recall as it only requires not lower than 30% of the correct predicted images. For future study, the proposed project will be tested using a standard database available from the cloud to compare the accuracy of the algorithm. At the same time come up with different split tests among the existing images to further measure its efficiency.

References

[1] Dinh, H.T., Lee, C., Niyato, D. And Wang, P., “A Survey Of Mobile Cloud Computing: Architecture, Applications, And Approaches,” Wireless Communications And Mobile Computing, 13(18), Pp.1587-1611, 2013.

[2] Nguyen, A., Yosinski, J. And Clune, J., “Deep Neural Networks Are Easily Fooled: High Confidence Predictions For Unrecognizable Images,” In Proceedings Of The IEEE Conference On Computer Vision And Pattern Recognition (Pp. 427-436), 2015.

[3] Kumar, K. And Lu, Y.H., “Cloud Computing For Mobile Users: Can Offloading Computation Save Energy?,” Computer, (4), Pp.51-56, 2010.

[4] Khan, A.N., Kiah, M.M., Khan, S.U. And Madani, S.A., “Towards Secure Mobile Cloud Computing: A Survey,” Future Generation Computer Systems, 29(5), Pp.1278-1299, 2013.

[5] Rimal, B.P., Choi, E. And Lumb, I., “A Taxonomy And Survey Of Cloud Computing Systems,” In 2009 Fifth International Joint Conference On INC, IMS And IDC (Pp. 44-51), IEEE, 2009, August.

[6] Huerta-Canepa, G. And Lee, D., June. “A Virtual Cloud Computing Provider For Mobile Devices.“ In Proceedings Of The 1st ACM Workshop On Mobile Cloud Computing & Services: Social Networks And Beyond (P. 6). ACM, 2010,

[7] Al-Shayea, Q. And Al-Ani, M., “Biometric Face Recognition Based On Enhanced Histogram Approach,” International Journal Of Communication Networks And Information Security, 10(1), Pp.148-154, 2018.

[8] Sun, Y., Wang, X. And Tang, X., “Deep Convolutional Network Cascade For Facial Point Detection.” In *Proceedings Of The IEEE Conference On Computer Vision And Pattern Recognition* (Pp. 3476-3483), 2013.

[9] Jafri, R., & Arabnia, H. R. “A Survey Of Face Recognition Techniques,” Jips, 5(2), 41-68, (2009).

[10] Ahonen, T., Pietikainen, M. And Hadid, A., “Face Recognition With Local Binary Pattern,” Lecture Notes in Computer Science, Springer Berlin Heidelberg, Germany, 2004, pp. 469 – 481. ECCV, 2004.

[11] Sun, Y., Wang, X. And Tang, X., 2013. “Deep Convolutional Network Cascade For Facial Point Detection,” In Proceedings Of The IEEE Conference On Computer Vision And Pattern Recognition (Pp. 3476-3483).

[12] Latman, N.S. And Herb, E., “A Field Study Of The Accuracy And Reliability Of A Biometric Iris Recognition System,” *Science & Justice*, *53*(2), Pp.98-102, 2013.
[13] Moreno, M., Shahrabadi, S., José, J., Du Buf, J.H. And Rodrigues, J.M., “Realtime Local Navigation For The Blind: Detection Of Lateral Doors And Sound Interface” proceedings of the 4th International Conference on Software Development for Enhancing Accessibility and Fighting Info-exclusion (DSAI) and Procedia Computer Science, 14, Pp.74-82, 2012.
[14] Ess, A., Leibe, B., Schindler, K. And Van Gool, L., “Moving Obstacle Detection In Highly Dynamic Scenes,” In 2009 IEEE International Conference On Robotics And Automation(Pp. 56-63). IEEE, 2009 May.
[15] Viola, P. And Jones, M., “Rapid Object Detection Using A Boosted Cascade Of Simple Features,” *Cvpr (1)*, *1*(511-518), pp.3. 2001.
[16] Guresen, E. And Kayakutlu, G., “Definition Of Artificial Neural Networks With Comparison To Other Networks,” Procedia Computer Science, 3, Pp.426-433, 2011.
[17] Bereta, M., Pedrycz, W. And Reformat, M., ”Local Descriptors And Similarity Measures For Frontal Face Recognition: A Comparative Analysis,” Journal Of Visual Communication And Image Representation, 24(8), Pp.1213-1231, 2013.