Design Novel CNN Architecture to Protect Personal Devices from Unauthorized Access Using Face Recognition

Sarah Kareem Salim[1], Mohammed Majid Msallam[2,3], Huda Ismail Olewi[4]

[1]Electrical Engineering Department, University of Misan, Misan, Iraq.
[2]Computer Engineering Department, Ankara University, Ankara, Turkey.
[3]Control and Systems Engineering Department, University of Technology, Baghdad, Iraq.
[4]Civil Engineering Department, University of Misan, Misan, Iraq

With the advancement of technology and the development of the Internet of Things (IoT), the importance of protecting devices is becoming increasingly important to secure them from unauthorized access. The emergence of personal devices such as Personal Computers (PC) and smartphones increases the need for advanced security techniques because these devices contain important and personal information. Thus, technology companies tend to use biometrics to protect devices. This study proposes a novel Convolutional Neural Networks (CNN) architecture designed to use biometrics to protect personal devices from unauthorized access. Face image is one of the biometrics used to secure the device using the proposed CNN architecture. The device allows access to data if the proposed CNN architecture makes a high match rate. The proposed CNN architecture had trained on the AT&T standard database where it had high accuracy.

Keywords: Machine learning, Convolutional Neural Networks, Face recognition, Biometric, Cyber security..

1. INTRODUCTION

Biometric systems utilize physiological or behavioral characteristics to automatically identify and authenticate individuals [1]. Among these systems, face recognition plays a crucial role and finds applications in security, access control, and entertainment [2]. Companies like Apple and Samsung have implemented face detection technology to recognize users to secure data in their devices [3].

Face recognition, as a biometric technology, analyzes features in images or video frames to identify individuals [4]. The process of face recognition involves two main steps: feature extraction and classification [5]. Classification, which predicts the class of variables, is often performed using intelligent methods such as machine learning, a branch of artificial intelligence [6]. Machine learning focuses on developing techniques that enable computers to learn and improve their performance on specific tasks using data [7]. The machine learning process consists of training and testing stages [8], as shown in Figure 1. Researchers have utilized artificial intelligence and machine learning to enhance the performance of face recognition and protect devices and data. Therefore, the aim of this study is to employ Convolutional Neural Networks (CNN) for face detection and classification and propose a new architecture to improve face recognition performance based on previous research.

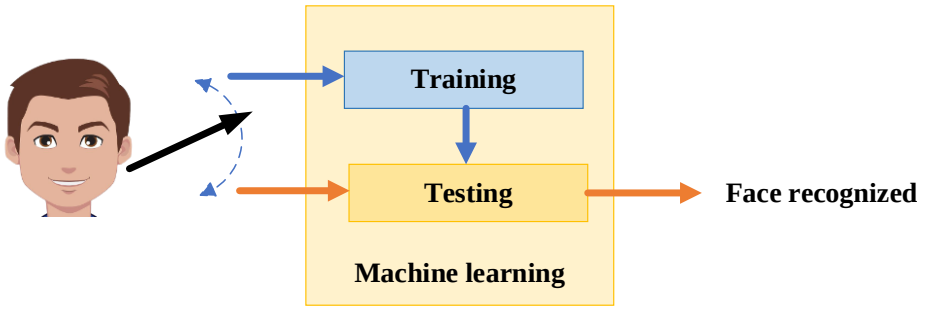


Fig. 1 General scheme of machine learning.

Previous studies have predominantly focused on developing intelligent methods to meet the demands of the technological era. For instance, Pranav et al. [9] designed and evaluated a real-time face recognition system using a CNN. They tested their approach using standard AT&T datasets and further extended it to develop a real-time system. Their method included systematic parameter tuning to enhance system performance, achieving accuracies of 98.00% for real-time inputs and 98.75% for standard datasets. However, despite its high accuracy, their proposed CNN architecture was large and computationally intensive. It is worth mentioning that CNN architectures typically exhibit high accuracy with a small size [10]. In another context, deep learning techniques were applied to detect and recognize goat faces [11]. The researchers collected goat face datasets to train their model, achieving an accuracy of 96.4%. Additionally, Hsiao et al. [12] investigated the impact of mask usage on face recognition performance and eye movements, exploring potential correlations between changes in eye movement strategy and performance adjustments. Chatterjee et al. [13] employed the UNet architecture for thermal face recognition. They converted thermal spectrum images into visible spectrum images and classified them using a CNN. They developed a UNet architecture based on the residual network backbone to generate visible face images from thermal face images. However, their proposed UNet architecture was large and computationally resource-intensive [14]. To enhance security when accessing cloud computing services via iPhones, Pukdesree et al. [15] proposed a multi-biometric system that incorporates physical features such as voice and face recognition, and behavioral features such as passwords. Their research focused on integrating data from multiple biometric sources.

In the literary review, the proposed models consumed device resources because the used architecture in their models was large and computationally expensive. This motivated us to propose a new architecture for CNN that is computationally inexpensive, does not consume device resources, and is highly accurate. The contribution of our system is to use the new proposed architecture of CNN in cyber security to protect personal devices from unauthorized access.

2. Proposed System

The proposed system we are working on is represented by the system outlined in Figure 2. The process begins with the camera capturing a facial image, which then passes through a pre-trained neural network. This neural network provides a predicted value. If the predicted value exceeds the threshold, our system is permitted to access the device. If the value is lower than that threshold, the process is repeated to capture another image, and the above process continues.

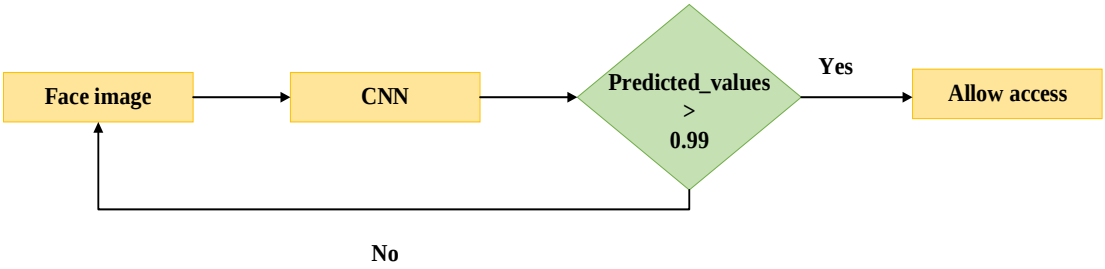


Fig. 2 Our proposal.

3. Convolutional Neural Network

A Convolutional Neural Network (CNN) is a network architecture for a deep learning algorithm and is very similar to ordinary neural networks that can take in an input image and classify an object inside it [16]. CNN is designed to learn spatial hierarchies of features automatically and adaptively from input data [17]. There are several architectures of CNN as PSPNet, DeeplabV3+, and UNet [18]. Figure 3 is shown the architecture of UNet. Many researchers tend to improve the accuracy of CNN by designing a new architecture for CNN [19].

The architecture of CNN is to organize the layers which consist of CNN in a specific style as the convolution layer, ReLU layer and, so on [21]. In this work, the used layers to build CNN architecture are the input layer, convolution layer, batch normalization layer, ReLU layer, Addition layer, fully connected layer, softmax layer, and output layer. The input layer is to hold the raw data which is the pixel value of the image. A convolutional layer uses slide convolutional filters as 2-D input. The layer convolutes the input by moving the filters vertically and horizontally along the input and computing the dot product of the weights and the input [22]. A batch normalization layer normalizes a mini batch of data across to make neural networks faster and more stable all observations for each channel independently [23]. A ReLU layer implements a threshold operation on each element of the data. ReLU is a non-

linear activation function as in Figure 4 that is defined mathematically as [24]:

$$F(t) = \max(0, t) \quad ..(1)$$

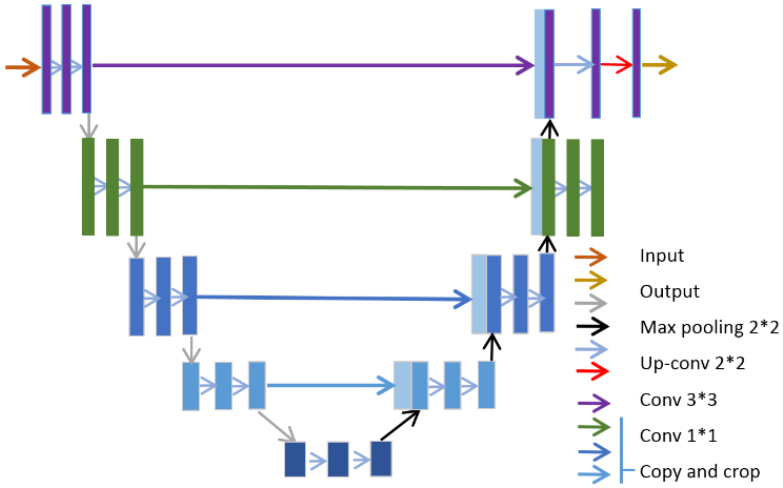


Fig. 3 The architecture of UNet [20].

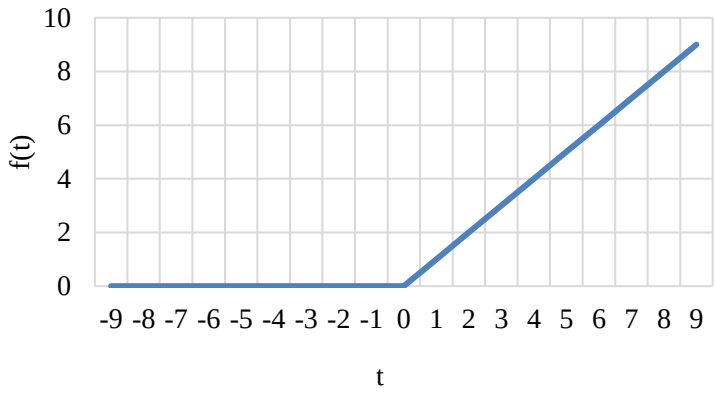


Fig. 4 The plot of the ReLU activation function.

An additional layer is also used in this work that uses to concatenate the output of layers. A fully connected layer is the ordinary neural network that multiplies the input by a weight matrix and then adds a bias vector [25]. A softmax layer is to apply a softmax function to the data and an output layer classifies the data. Softmax also has the benefit of converting results into a normalized probability distribution that may be shown to users or fed into other systems [26]. Table 1 is shown the size and count of layers used in this work.

Table 1 The number of layers and sizes in this work.

Layer	Size
Convolution layer 1	80 X 80
Convolution layer 2	2 X 40
Convolution layer 3	2 X 40

Convolution layer 4	2 X 40
Convolution layer 5	2 X 40
Convolution layer 6	2 X 40
Convolution layer 7	1 X 40
ReLU layer 1	80 X 80
ReLU layer 2	2 X 40
ReLU layer 3	2 X 40
ReLU layer 4	2 X 40
ReLU layer 5	2 X 40
ReLU layer 6	2 X 40
ReLU layer 7	1 X 40
Batch normalization layer	80 X 80
Input layer	112 X 92
Softmax layer	1 X 40
Output layer	1 X 40
additional layer 1	2 X 40
additional layer 2	2 X 40

If each layer receives information from the previous layer and the previous layer drops some information that is relevant to the classification problem, the effect of bottleneck will appear [27]. One of the solutions to this problem is to concatenate the output of the previous layer and the current layer. So, this work suggests the new architecture of the CNN to improve accuracy using common solution of bottleneck problem. The new architecture of the CNN in this paper is derived from the architecture of UNet after many attempts to get perfect accuracy. In the new architecture of the CNN, it organizes the layers in a new style such as shown in Figure 5. This style designed to work on the AT&T standard database [9] contains to detect the face and classify the category of the face.

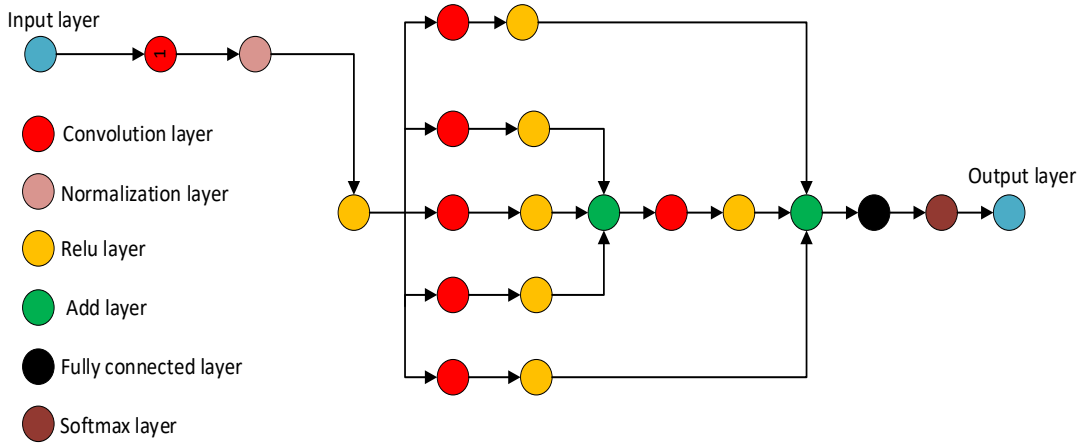


Fig. 5 The proposed architecture of CNN for face recognition.

This work will go through two phases which are the training phase and testing phase. During the training phase, the CNA will train to detect and classify the face in parts of images in the AT&T standard database. The training of CNN in the paper is supervision. Figure 6 is shown the training phase in proposed architecture of CNN for face recognition.

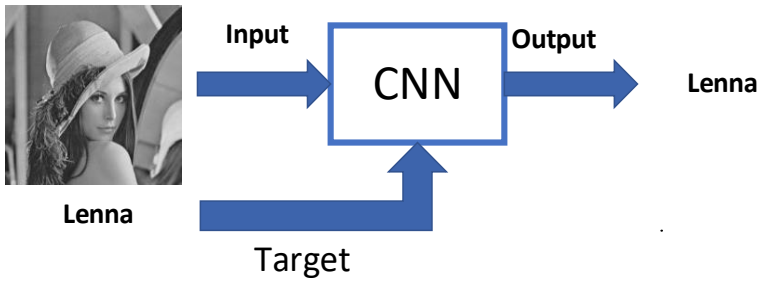


Fig. 6 The training phase in proposed architecture of CNN.

While during the testing phase, this model will test to evaluate its performance on the other part of the images in the AT&T standard database. In this article, the performance which evaluates this work is accuracy the detecting and classifying the face and determining an identity of a person. Figure 7 is shown the testing phase in proposed architecture of CNN for face recognition.

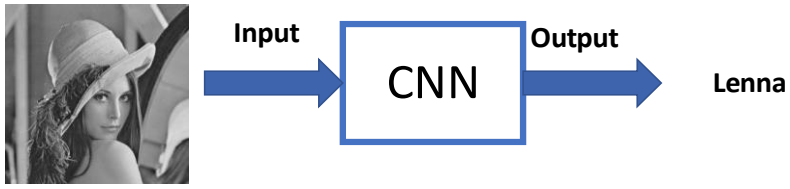
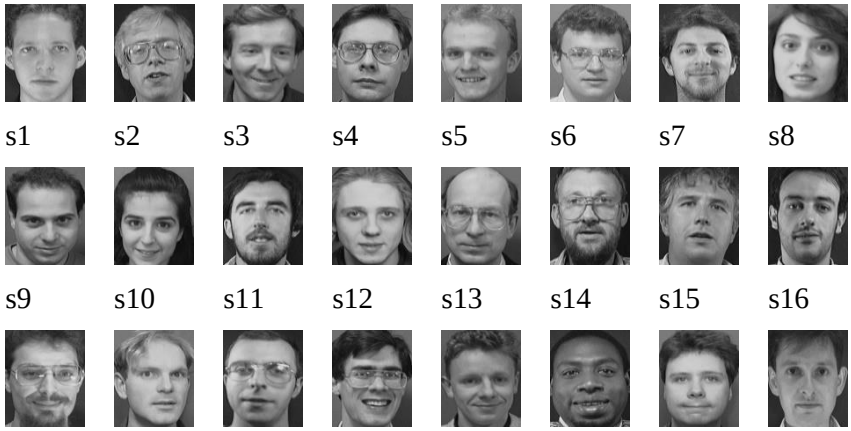


Fig. 7 The testing phase in proposed architecture of CNN.

4. Results

The evaluation of the architecture for CNN was done using the AT&T standard database which is a collection of facial images from the AT&T Laboratories Cambridge obtained between April 1992 and April 1994 and are available in the AT&T database of faces (formerly known as the "ORLDatabase of Faces") [28]. The AT&T standard database contains four hundred images with size 112*92 for forty individuals. The samples of images and their label from the AT&T database are shown in Figure 8.



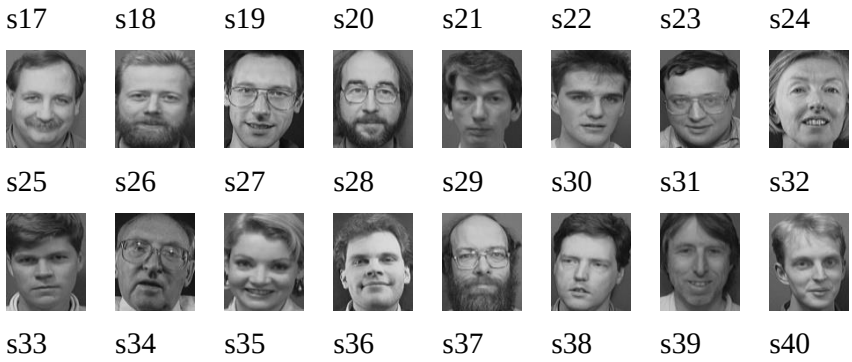


Fig. 8 The image with their label in AT&T database.

On a personal computer (processor: Intel® CoreTm 7-8550U CPU @ 1.80GHz 2.00 GHz; memory: 16 Gb; operating system: Windows 10 64 bits), the complete model training and validation process was conducted. The NVIDIA GeForce MX130 2 Gb graphics processing unit (GPU) mode was used to optimize training speed. Additionally, MATLAB R2018b was used to implement the new architecture of CNN for this work. Table 2 presents the detailed used parameters in proposed modeling such as input batch size, learning rate, and so on. The reason to choose MATLAB to implement this work is very reliable and stable to implement a proposal model [29]. MATLAB R2018b also provides tools to implement CNN.

Table 2 The used parameters in proposed models.

Modeling Parameters	Values
Number of training samples	290
Number of validation samples	30
Number of test samples	80
Input size	112 X 92
Training number of epochs	100
learning rate	0.96e-4
Image input batch size	100
Number of classes	40
Maximum iterations	300

The accuracy of training and validation in the training phase is shown in Figure 9. The proposed model of the CNN obtains the perfect accuracy early, which is the percentage of labels that the network properly predicts, at the fiftieth iteration approximately. The elapsed time to train was five minutes for the whole training phase.

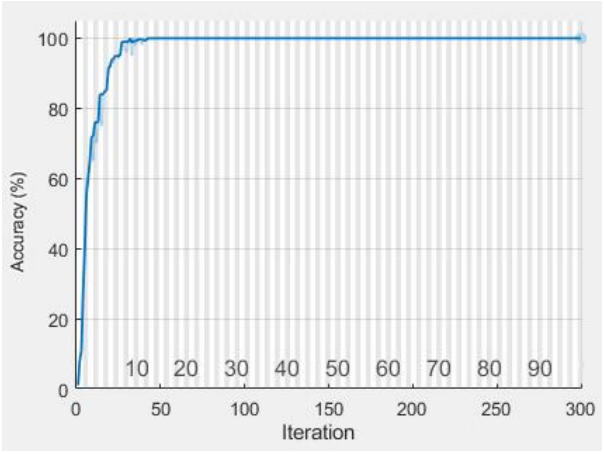


Fig. 9 The accuracy of training and validation of the new architecture of CNN.

The loss function for the training phase is cross-entropy function which calculates the difference between two probability distributions [30]. The cross-entropy function awards a small score for small differences tending to 0 and a large score for large differences close to 1 [31]. The cross-entropy loss (£) is defined as the following [32]:

$$£(\boldsymbol{y},\boldsymbol{p}) = -\frac{\mathbf{1}}{\boldsymbol{N}}\sum_{\boldsymbol{i=1}}^{\boldsymbol{N}}\sum_{\boldsymbol{c=1}}^{\boldsymbol{C}} \boldsymbol{y_{i,c}} * \boldsymbol{log}(\boldsymbol{p_{i,c}}) \qquad ..(\mathbf{2})$$

Where $p_{i,c}$ is a matrix of predicted values for each class, $y_{i,c}$ uses a one-hot encoding scheme of ground truth labels, and where indices c and i iterate over all classes and pixels, respectively. The loss of training and validation in the training phase illustrates the model of the CNS obtains the perfect error at the fiftieth iteration approximately as shown in Figure 10.

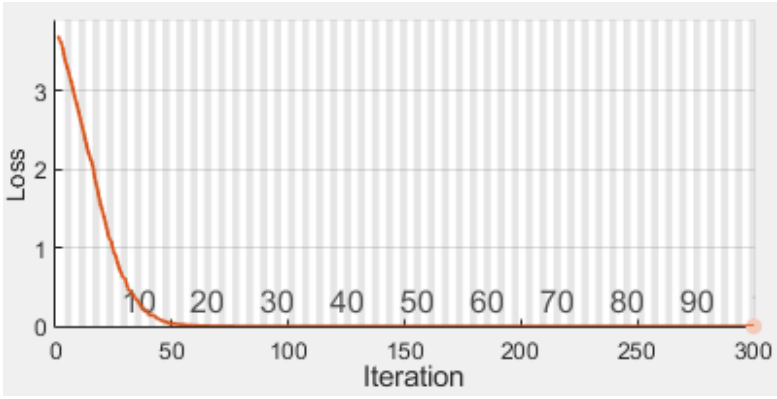
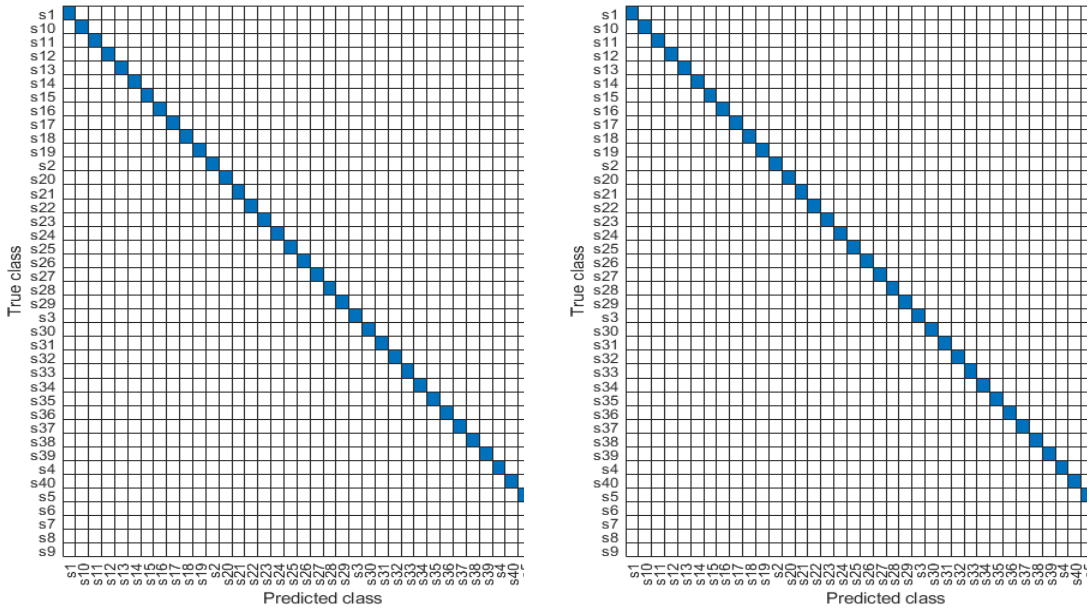


Fig. 10 The values of loss function of the new architecture of CNN.

To evaluate the model of a neural network, a confusion matrix is a matrix that displays the distribution of predicted and actual class instances [33]. The confusion matrix is used for the performance evaluation of the proposed model of the new architecture. Figure 11 is

confusion matrices in the training phase and testing phase.



a) Training phase. b) Testing phase

Fig. 11 Confusion matrix.

Table 3 Predicted Probability.

Truth class	Predicted class	Predicted values
s1	s1	0.9995
s1	s1	0.9990
s10	s10	0.9997
s10	s10	0.9996
s11	s11	0.9996
s11	s11	0.9999
s12	s12	0.9997
s12	s12	0.9998
s13	s13	0.9993
s13	s13	1.0000
s14	s14	0.9997
s14	s14	1.0000
s15	s15	0.9998
s15	s15	0.9997
s16	s16	0.9999
s16	s16	0.9990
s17	s17	0.9998
s17	s17	0.9999
s18	s18	0.9999
s18	s18	0.9998
s19	s19	0.9994
s19	s19	0.9998

s2	s2	0.9998
s2	s2	0.9999

Predicted scores in Table 3 refer to the predicted values of a model for a given input data. Predicted values are the probability that predicted classes is truth class that are given by a number between 0 and 1. Where zero indicates that a given sample does not completely belong targeted class, while one indicates that a given sample completely belongs targeted class. Table 3 contains examples of predicted value with truth class and predicted class for the proposed architecture of CNN.

In Table 4, the accuracy recognition of the proposed work is compared with the results reported in the literature which uses the same dataset for face recognition. It has been shown that the proposed approach and architecture of CNN are comparable to the research described in the literature. The enhancement in the accuracy of the recognition face system of the proposed work is obtained due to rearranging the layers to get a new architecture.

Table 4 A comparison of face recognition system results.

Reference	Accuracy
[34]	95.00%
[35]	98.30%
[9]	98.75%
Work proposal	100%

5. Conclusion and Future Work

This article proposed the new architecture of CNN to protect personal devices from unauthorized access. The proposed architecture of CNN was twenty layers to extract features from the face person image. The proposed system allowed to access the device data if the predicted value was higher than 0.99. The proposed architecture of CNN trained and tested on the AT&T database. The proposed CNN architecture achieved a high accuracy rate in the recognition of faces. The predicted values of the proposed architecture of CNN were higher than 0.99. This work can be extended as future work by the implementation in real-time.

References

1. E. N. Uchenna, O. O. Raphael, and A. T. Leonard, “Overview of Technologies and Fingerprint Scanner Used for Biometric Capturing,” Innovation, vol. 1, no. 1, pp. 1–5, 2020, doi: 10.11648/j.innov.20200101.11.
2. D. K. Verma and S. Ojha, “Performance Analysis of Biometric Systems: A Security Perspective,” International Journal of Advanced Research in Computer and Communication Engineering, vol. 8, no. 4, pp. 104–110, 2019, doi: 10.17148/ijarcce.2019.8417.
3. H. Baqeel and S. Saeed, “Face detection authentication on Smartphones: End Users Usability Assessment Experiences,” 2019 International Conference on Computer and Information Sciences, pp. 1–6, 2019. doi: 10.1109/ICCISci.2019.8716452.
4. T. C. Hlongwane, T. E. Mathonsi, D. D. Plessis, and T. Muchenje, “A Review Paper on Facial Recognition Techniques in E-business,” 2021 International Conference on Computational Science and Computational Intelligence, pp. 1702–1707, 2021. doi: 10.1109/CSCI54926.2021.00323.

5. S. Sharma, M. Bhatt, and P. Sharma, "Face recognition system using machine learning algorithm," Proceedings of the Fifth International Conference on Communication and Electronics Systems, pp. 1162–1168, 2020. doi: 10.1109/ICCES48766.2020.9137850.
6. I. S. Stafford, M. Kellermann, E. Mossotto, R. M. Beattie, B. D. MacArthur and S. Ennis, "A systematic review of the applications of artificial intelligence and machine learning in autoimmune diseases," NPJ digital medicine, vol. 3, no. 1, 2020. doi: 10.1038/s41746-020-0229-3.
7. K V R Pranav, K J Sarma, "Origin, Development and Uses of Machine Learning," International Journal for Multidisciplinary Research, vol. 5, no. 1, pp. 1–18, 2023. doi: 10.36948/ijfmr.2023.v05i01.1367.
8. S. Ren , K. Sun , C. Tan, and F. Dong, "A Two-Stage Deep Learning Method for Robust Shape Reconstruction with Electrical Impedance Tomography," IEEE Transactions on Instrumentation and Measurement, vol. 69, no. 7, pp. 4887–4897, 2020. doi: 10.1109/TIM.2019.2954722.
9. K. B. Pranav, and J. Manikandan, "Design and Evaluation of a Real-Time Face Recognition System using Convolutional Neural Networks," Procedia Computer Science, vol. 171, pp. 1651–1659, 2020. doi: 10.1016/j.procs.2020.04.177.
10. A. P. Sunija, S. Kar, S. Gayathri, V. P. Gopi, and P. Palanisamy, "OctNET: A Lightweight CNN for Retinal Disease Classification from Optical Coherence Tomography Images," Computer Methods and Programs in Biomedicine, vol. 200, 2021. doi: 10.1016/j.cmpb.2020.105877.
11. M. Billah, X. Wang, J. Yu, and Y. Jiang, "Real-time goat face recognition using convolutional neural network," Computers and Electronics in Agriculture, vol. 194, 2022. doi: 10.1016/j.compag.2022.106730.
12. J. Hui, W. Liao, and R. V. Y. Tso, "Impact of mask use on face recognition : an eye - tracking study," Cognitive Research: Principles and Implications, vol. 7, no. 1, 2022. doi: 10.1186/s41235-022-00382-w.
13. S. Chatterjee and W. T. Chu, "Thermal face recognition based on transformation by residual U-net and pixel shuffle upsampling," International Conference on Multimedia Modeling, vol. 11961, pp. 679–689, 2020. doi: 10.1007/978-3-030-37731-1_55.
14. C. Jin, S. Li, T. D. Do, and H. Kim, "Real-time human action recognition using CNN over temporal images for static video surveillance cameras," Advances in Multimedia Information Processing--PCM 2015: 16th Pacific-Rim Conference on Multimedia, 2015., doi: 10.1007/978-3-319-24078-7.
15. S. Pukdesree and P. Netinant, "Reviewed : The Face Authentication Processes for Accessing Cloud Computing Services using iPhone," TEM Journal , vol. 7, no. 3, pp. 475–479, 2018. doi: 10.18421/TEM73-01.
16. V. R. Allugunti, "A machine learning model for skin disease classification using convolution neural network Healthcare View project A machine learning model for skin disease classification using convolution neural network," International Journal of Computing, Programming and Database Management, vol. 3, no. 1, pp. 141–147, 2022.
17. Z. Ye and J. Yu, "Multi-level features fusion network-based feature learning for machinery fault diagnosis," Applied Soft Computing, vol. 122, 2022. doi: 10.1016/j.asoc.2022.108900.
18. B. Y. Liu, K. J. Fan, W. H. Su, and Y. Peng, "Two-Stage Convolutional Neural Networks for Diagnosing the Severity of Alternaria Leaf Blotch Disease of the Apple Tree," Remote Sensing, vol. 14, no. 11, 2022. doi: 10.3390/rs14112519.
19. T. Kattenborn, J. Leitloff, F. Schiefer, and S. Hinz, "Review on Convolutional Neural Networks (CNN) in vegetation remote sensing," ISPRS Journal of Photogrammetry and Remote Sensing, vol. 173, pp. 24–49, 2021. doi: 10.1016/j.isprsjprs.2020.12.010.
20. Y. Ding, F. Chen, Y. Zhao, Z. Wu, C. Zhang, and D. Wu, "A Stacked Multi-Connection

Simple Reducing Net for Brain Tumor Segmentation," IEEE Access, vol. 7, 2019. doi: 10.1109/ACCESS.2019.2926448.
21. T. Huynh-The, C. H. Hua, Q. V. Pham, and D. S. Kim, "MCNet: An Efficient CNN Architecture for Robust Automatic Modulation Classification," IEEE Communications Letters, vol. 24, no. 4, pp. 811–815, 2020. doi: 10.1109/LCOMM.2020.2968030.
22. L. Zhou, H. Yu, and Y. Lan, "Deep Convolutional Neural Network-Based Robust Phase Gradient Estimation for Two-Dimensional Phase Unwrapping Using SAR Interferograms," IEEE Transactions on Geoscience and Remote Sensing, vol. 58, no. 7, pp. 4653–4665, 2020. doi: 10.1109/TGRS.2020.2965918.
23. S. Singh and S. Krishnan, "Filter Response Normalization Layer: Eliminating Batch Dependence in the Training of Deep Neural Networks," Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 11234–11243, 2020. doi: 10.1109/CVPR42600.2020.01125.
24. S. Sharma, S. Sharma, and A. Anidhya, "Activation functions in neural networks," Towards Data Sci, vol. 6, no. 12, pp. 310–316, 2017.
25. A. Novikov, D. Podoprikhin, A. Osokin, and D. Vetrov, "Tensorizing neural networks," Advances in neural information processing systems, vol. 28, pp. 442–450, 2015.
26. P. TS and P. Shrinivasacharya, "Evaluating neural networks using Bi-Directional LSTM for network IDS (intrusion detection systems) in cyber security," Global Transitions Proceedings, vol. 2, no. 2, pp. 448–454, 2021. doi: 10.1016/j.gltp.2021.08.017.
27. N. Ketkar and J. Moolayil, Deep Learning with Python. 2021. doi: 10.1007/978-1-4842-5364-9.
28. C. Cifarelli, G. Manfredi, and L. Nieddu, "Statistical face recognition and intruder detection via a fc-means iterative algorithm: A resampling approach," Proceedings of the 13th Multi-Conference on Systemics, Cybernetics and Informatics, Vol. 2. 2009.
29. J. Till, V. Aloi, and C. Rucker, "Real-time dynamics of soft and continuum robots based on Cosserat rod models," The International Journal of Robotics Research, vol. 38, no. 6, pp. 723–746, 2019. doi: 10.1177/0278364919842269.
30. L. Nieradzik, G. Scheuermann, D. Saur, and C. Gillmann, "Effect of the output activation function on the probabilities and errors in medical image segmentation," arXiv preprint arXiv, 2021.
31. P. K. Jain, N. Sharma, A. A. Giannopoulos, L. Saba, A. Nicolaides, and J. S. Suri, "Hybrid deep learning segmentation models for atherosclerotic plaque in internal carotid artery B-mode ultrasound," Computers in Biology and Medicine, vol. 136, 2021. doi: 10.1016/j.compbiomed.2021.104721.
32. M. Yeung, E. Sala, C. B. Schönlieb, and L. Rundo, "Unified Focal loss: Generalising Dice and cross entropy-based losses to handle class imbalanced medical image segmentation," Computerized Medical Imaging and Graphics, vol. 95, 2022. doi: 10.1016/j.compmedimag.2021.102026.
33. I. Markoulidakis, I. Rallis, I. Georgoulas, G. Kopsiaftis, A. Doulamis, and N. Doulamis, "Multi-Class Confusion Matrix Reduction method and its application on Net Promoter Score classification problem," The 14th pervasive technologies related to assistive environments conference, pp. 412–419, 2021. doi: 10.1145/3453892.3461323.
34. H. Hu, S. Afaq, A. Shah, M. Bennamoun, and M. Molton, "2D and 3D Face Recognition Using Convolutional Neural Network," IEEE Transactions on Geoscience and Remote Sensing, vol. 58, no. 7, pp. 4653-4665, July 2020. doi: 10.1109/TGRS.2020.2965918..
35. P. Kamencay, M. Benco, T. Mizdos, and R. Radil, "A New Method for Face Recognition Using Convolutional Neural Network Face Recognition System – State of the Art," Advances in Electrical and Electronic Engineering, vol. 15, no. 4, pp. 663–672, 2017. doi: 10.15598/aeee.v15i4.2389.