

Fingerprint recognition based on collected images using deep learning technology

Ali Fadhil Yaseen Althabhawe¹, Bashra Kadhim Olewi Chabor Alwawi²

¹Directorate General of Education in Holy, Karbala, Iraq

²Control and Systems Engineering Department, University of Technology-Iraq, Baghdad, Iraq

Article Info

Article history:

Received Jan 1, 2021

Revised Nov 26, 2021

Accepted Dec 10, 2021

Keywords:

Collected images

Convolutional neural network

Deep learning

Fingerprint

Recognition

ABSTRACT

The fingerprint identification is the most widely used authentication system. The fingerprint uniqueness for each human being provides error-free identification. However, during the scanning process of the fingerprint, the generated image using the fingerprint scanner may differ slightly during each scan. This paper proposes an efficient matching model for fingerprint authentication using deep learning based deep convolutional neural network (CNN or ConvNet). The proposed deep CNN consists of fifteen layers and is classified into two stages. The first stage is preparation stage which includes the fingerprint images collection, augmentation and pre-processing steps, while the second stage is the features extraction and matching stage. Regarding the implantation results, the proposed system provided the best matching for the given fingerprint features. The obtained training accuracy of the proposed model is 100% for training dataset and 100% for validating dataset.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Bashra Kadhim Olewi Chabor Alwawi

Control and Systems Engineering Department, University of Technology Iraq

Al-Wehda neighborhood, Box Office 19006, Baghdad, Iraq

Email: 60010@uotechnology.edu.iq

1. INTRODUCTION

Biometrics identification technology are important in security systems of different applications to reduce security threats of military applications, government centers and airports, and criminal applications [1], [2]. Accordingly, the need of safe, reliable, and convenient a identify identification system. Many fingerprint recognition algorithms have been suggested. Park *et al.* [3] proposed a fingerprint recognition system using feature detection algorithm based on scale-invariant feature transform. Cappelli *et al.* [4] presented a new 3D data structure based on minutiae distances and angles named minutiae-cylinder-code. Patil and Suralkar [5] presented fingerprint classification using a neural network. The neural network performed matching step and was successfully used for identifying and classifying the fingerprint utilizing back propagation algorithm. Although many of the previous work has achieved high performance precision, but it involves a lot of pre-processing, which may not be suitable for data sets of different fingerprints under different conditions. recently, efforts have been made for developing recognition models for learning and features extraction automatically [6], [7]. Recently, the emergence of the deep learning technology has great successful achievement in different applications such as image processing, natural language processing and so on [7]. As a results, the fingerprint classification based deep neural network can be also achieved good results. Minaee *et al.* [8] proposed fingerprint recognition model utilizing convolutional neural network (CNN) by tuning trained CNN model on ImageNet and applied the proposed model of the PolyU fingerprint database. Das *et al.* [9] proposed deep learning model based on finger-vein identification and obtained highly

accurate and stable performance using different finger-vein images with different quality. Militello *et al.* [10] has presented fingerprint classification systems using deep learning based CNN using two datasets of fingerprint called PolyU and national institute of standards and technology (NIST) and introduced comparisons study with ResNet, AlexNet, and GoogLeNet. Pandya *et al.* [11] has proposed fingerprint classification systems using deep learning based deep CNN using thinning of fingerprint, and Gabor improvement, histogram equalization.

In [12], [13] proposed a new COVID-19 detection model based on histogram techniques and new CNN architecture for improving the images contrast and for COVID-19 detection in chest X-ray image, respectively. Oleiwi and Althabhawe [14] has presented new architecture of CNN for matching the human iris images in authentication system. In [15], [16] introduced hybrid approach based on deep learning and image preprocessing for noisy iris dataset. In [17], [18] have proposed sentiment analysis system using deep learning, for Arabic language [17] and for American sign language as static letters (ASL) in hand gesture recognition [18]. In [19], [20] has suggested real time assistive blind model for object detection using deep learning based YOLO algorithm and raspberry model B pi 3. Mahmood and Saud [21] presented monitoring System based detection and classification for moving cars in streaming video. In this paper, the deep learning model based deep CNN is suggested for matching between the samples of presented fingerprint images. The fingerprint database is collected using fingerprint sensor. The fingerprint image augmentation and preprocessing is applied in order to expand and enhance the images. Then, the features extracted using deep CNN and after that, the Softmax performed as a recognizer. The paper is structured as: section 2 introduces the proposed fingerprint recognition system. Section 3 explains the CNN overview and its architecture, and section 4 gives simulation results and discussion. Finally, section 5 presents the conclusion part for the study.

2. THE PROPOSED FINGERPRINT RECOGNITION MODEL

Fingerprint image recognition plays an important role in automated fingerprint identification system, especially in the large dataset. With development of identification system of fingerprint based on big fingerprint database, the accuracy and fast processing time are required. The proposed model is comprised of the following phases.

2.1. Fingerprint dataset acquisition

There are different visual fingerprint patterns that unequally distributed. Thus, the fingerprint types can be divided into five types [2]; the right loop, left loop, the whorl arch and tented arch. In the initialization stage, the fingerprint database of the authorized users was collected using fingerprint sensor named ZKT ZK4500 as a USB fingerprint sensor [22]. The collected dataset includes 1,500 images for 15 authorized users with different ages, each one has 100 samples of his 10 fingers. Some images of the collected dataset are presented in Figure 1, as can be noticed in this figure, there are slightly different colour distributions and different sizes.



Figure 1. Eighteen fingerprint images using fingerprint sensor

2.2. Data augmentation and preprocessing

The fingerprint images augmentation and pre-processing step is important process in order to enlarge and enhance the dataset before feeding the dataset to the network. Essentially it applies random rotations, shifts, flips, crops, and sheers on fingerprint images. The fingerprint images in the dataset are converted from red, green, and blue (RGB) to grayscale images with pixel values ranging from 0 to 255 and resized to a uniform dimension images of size 360×280.

2.3. The matching process

The global and local features of the collected fingerprint images will be extracted using deep CNN. Hence, the representations of the structured feature template data stored in the database, as will be explained in the next section. The matching process is responsible for comparing the stored image and inputted image in order to achieve a verdict match or non-match. Hence, the entered fingerprint image scanned using fingerprint sensor and the feature extraction implemented through the same process in the initialization stage, as illustrated in Figure 2.

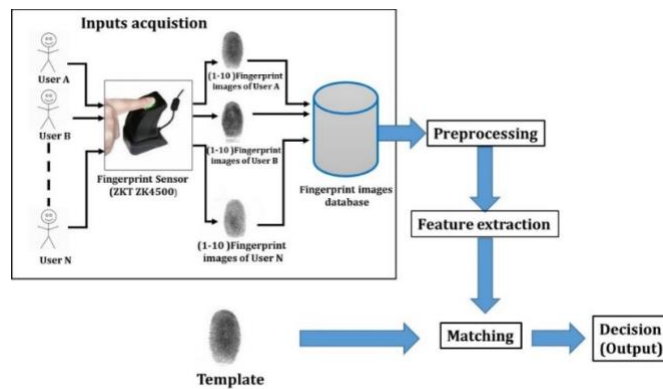


Figure 2. General block diagram of the proposed recognition system

3. OVERVIEW OF CNN

The deep learning field represents a subfield of machine learning concerning with algorithms that inspired by human brain structure and function which is named artificial neural networks. Deep-learning methods such as CNN consists of a multi convolutional layers with a fully connected layer, based a robust method of feature extraction and classification [9], [23], [24]. The CNN layers will be explained as:

- Convolutional layers: represent a several 2-D convolutional layers started from the input maps x_m^l , where l and m represent indexes of the level and map respectively. The filters based on the kernels $w_{n,m}^l$, where n is the index of the filter. The n th is the output map y_n^l of layer l which can be calculated as [9], [25]:

$$y_n^l = \sum_m^{M^{l-1}} w_{n,m}^l * x_m^l + b_n^l \quad (1)$$

Where M^{l-1} denotes input maps number, $*$ is convolution, and b_n^l , represents bias of n th map output in the l th level.

- The rectified linear unit (ReLU) layer: represents activation layer as nonlinear layer which is used directly after the convolutional layer mentioned by (1). This layer allows model to train faster with high accuracy. By applying activation function in (2).

$$\text{Softmax}[f(x) = \max(0, x)] \quad (2)$$

which will be increased as non-linearity.

- Pooling layer: This layer minimizes the spatial size, by reducing the required parameters number for describing the network, which led to reduce the computational process that are required for training the model.

- Fully connected layer: fully connected of all neurons as in ordinary neural systems. It represents the last part of the network, for yielding the probabilities of the class. The linear combination O_n is expressed in (3), where x_m is the m th map input in the output layer [9], [26]. The high level features will be extracted in this layer by calculating probabilities of the available classes. The probability distribution of the input data over C different classes will be predicted by using the Softmax function, as defined in (4):

$$O_n = \sum_{m=1}^M (w_{n,m} * x_m + b_n) \quad (3)$$

$$p_u = \frac{\exp(O_u)}{\sum_{n=1}^C \exp(O_n)} \quad (4)$$

3.1. Designed CNN architecture

The structure of the designed CNN for performing fingerprint recognition system consists mainly of convolutional layer, pooling layer, fully connected layer and the activation functions as illustrated in Figure 3. The defined 2 D CNN structure has fifteen Layers: image input as gray image with size 360, 280, b1, then 3 convolution layers for extracting the features based on different sizes, strides and padding, followed by 3 ReLU layers as activation functions, after that 3-max pooling layers for reducing the parameters, fully connected layers, softmax layer and classification layer as introduced in Figure 4. Figure 5 represents the options set chosen for training the proposed CNN.

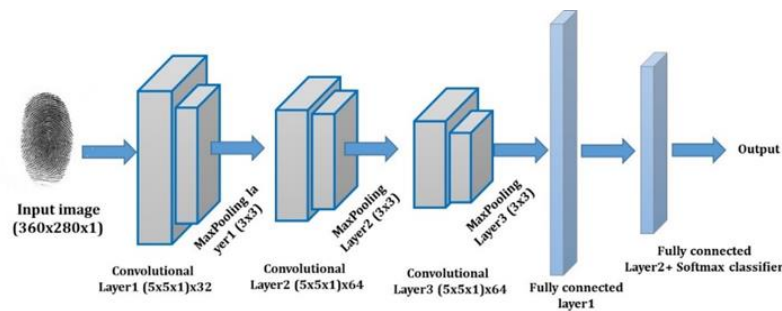


Figure 3. Employed CNN architecture

```

layers =
15x1 Layer array with layers:
 1 ** Image Input          360x280x1 images with 'zerocenter' normalization
 2 ** Convolution          32 5x5 convolutions with stride [1 1] and padding [2 2 2 2]
 3 ** ReLU
 4 ** Max Pooling          3x3 max pooling with stride [2 2] and padding [0 0 0 0]
 5 ** Convolution          32 5x5 convolutions with stride [1 1] and padding [2 2 2 2]
 6 ** ReLU
 7 ** Max Pooling          3x3 max pooling with stride [2 2] and padding [0 0 0 0]
 8 ** Convolution          64 5x5 convolutions with stride [1 1] and padding [2 2 2 2]
 9 ** ReLU
10 ** Max Pooling          3x3 max pooling with stride [2 2] and padding [0 0 0 0]
11 ** Fully Connected      64 fully connected layer
12 ** ReLU
13 ** Fully Connected      3 fully connected layer
14 ** Softmax
15 ** Classification Output crossentropy

```

Figure 4. The defined CNN fifteen layers

3.2. Software part of the CNN

The basic steps of the software part of the fingerprint identification system using CNN based multi layers are presented in flowchart in Figure 6. In the first step of this flowchart, the inputted image was recorded and read, then converted it from a color image to a gray image, and then CNN is applied for the purpose of detecting and classifying the inputted fingerprint if it matches the stored fingerprints or not. In the last stage, the detected image is classified into which class belongs of the stored fingerprints with rating ratio for each fingerprint stored in the dataset.

```

options =
  TrainingOptionsSGDM with properties:|
      Momentum: 0.9000
      InitialLearnRate: 1.0000e-04
      LearnRateScheduleSettings: [1x1 struct]
      L2Regularization: 1.0000e-04
      MaxEpochs: 300
      MiniBatchSize: 128
      Verbose: 0
      VerboseFrequency: 50
      ValidationData: [1x1 matlab.io.datastore.ImageDatastore]
      ValidationFrequency: 30
      ValidationPatience: 5
      Shuffle: 'once'
      CheckpointPath: ''
      ExecutionEnvironment: 'cpu'
      WorkerLoad: []
      OutputFcn: []
      Plots: 'training-progress'
      SequenceLength: 'longest'
      SequencePaddingValue: 0
    
```

Figure 5. The options set chosen for training the CNN network

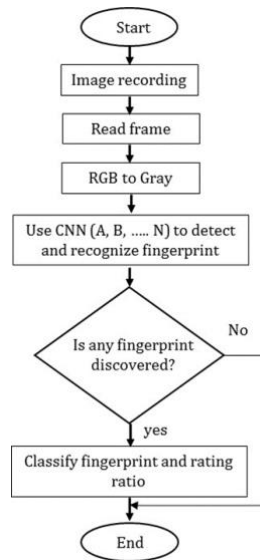


Figure 6. Flowchart of the fingerprint identification system using CNN

4. RESULTS AND DISCUSSION

The implementatin of the introduced model required two parts, software and hardware. The software part using MATLAB version 2020b [25], [27], and the hardware part using a PC assembled specially for the work with core i5 CPU composed of four 3.2 GHz working cores, a 10 GB RAM, and an advanced GPU of 6.1 computation capability measure. The proposed model are implemented in parallel computing fashion in which the four CPU cores and the GPU cooperate together.

In the initialization stage, the fingerprint images of the authorized users will be collected by fingerprint sensor. The features of the collected fingerprint image will be extracted, and then the comparison step in the matching stage. The template or inputted image will be compared with the stored dataset in order to achieve the matching outcome process. Hence, the query fingerprint image is scanned using fingerprint sensor. The feature extraction of the query fingerprint image implement using the same procedures in the initialization stage. The simulation has been carried out to evaluate the performance of the proposed system based on special fingerprint database. The fingerprint images have been acquired and quantized according to 360×280 and 256 gray. Figure 7 includes of patterns of the database that has been used for training process. While Figure 8 represents the performance of the proposed system, Figure 8(a). Training on single GPU, Figure 8(b). Training progress. Finally, the Fingerprint recognition has been represented in Figure 9. The obtained results indicated the success of the proposed model in achieving its intended in classification and identification of the fingerprint. Hence, the proposed system managed to achieve complex computations in relatively short times.

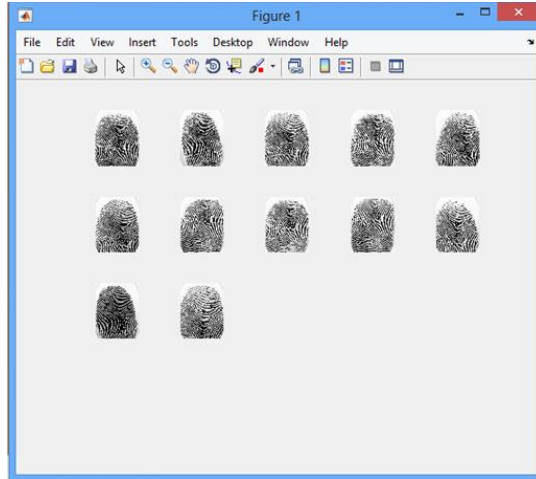
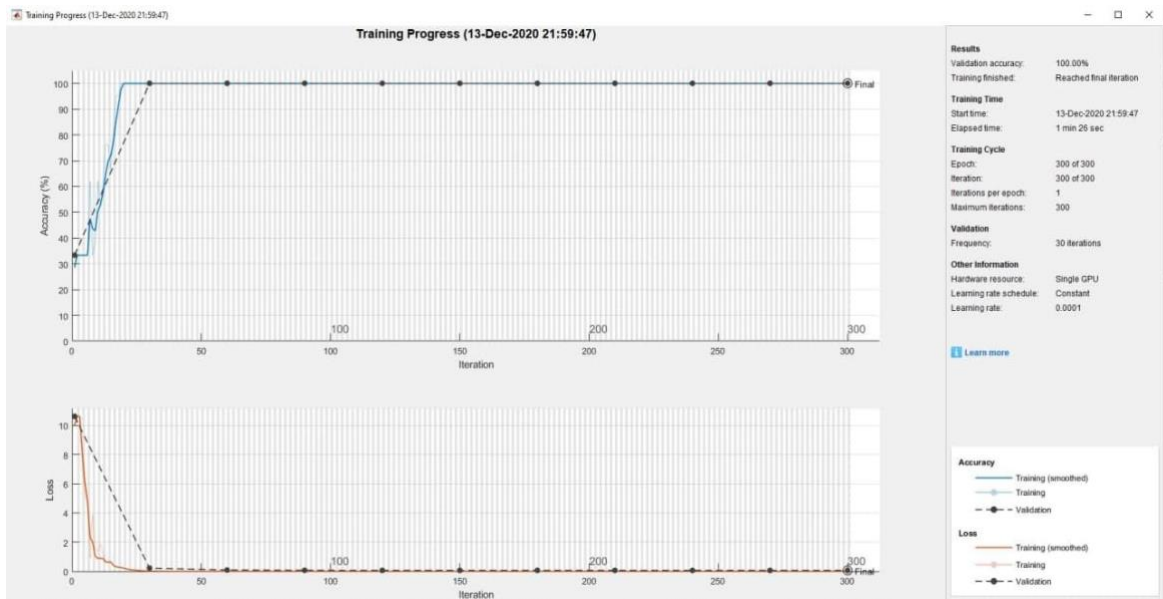


Figure 7. Patterns of the database that has been used for training process

Training on single GPU.
Initializing input data normalization.

Epoch	Iteration	Time Elapsed (hh:mm:ss)	Mini-batch Accuracy	Validation Accuracy	Mini-batch Loss	Validation Loss	Base Learning Rate
1	1	00:00:01	28.57%	33.33%	10.0100	10.6283	1.0000e-04
30	30	00:00:10	100.00%	100.00%	0.0136	0.2199	1.0000e-04
50	50	00:00:15	100.00%	100.00%	0.0009		1.0000e-04
60	60	00:00:18	100.00%	100.00%	0.0006	0.0929	1.0000e-04
90	90	00:00:27	100.00%	100.00%	0.0004	0.0709	1.0000e-04
100	100	00:00:30	100.00%	100.00%	0.0004		1.0000e-04
120	120	00:00:36	100.00%	100.00%	0.0003	0.0674	1.0000e-04
150	150	00:00:44	100.00%	100.00%	0.0003	0.0669	1.0000e-04
180	180	00:00:52	100.00%	100.00%	0.0003	0.0664	1.0000e-04
200	200	00:00:58	100.00%	100.00%	0.0002		1.0000e-04
210	210	00:01:01	100.00%	100.00%	0.0002	0.0659	1.0000e-04
240	240	00:01:09	100.00%	100.00%	0.0002	0.0656	1.0000e-04
250	250	00:01:12	100.00%	100.00%	0.0002		1.0000e-04
270	270	00:01:18	100.00%	100.00%	0.0002	0.0654	1.0000e-04
300	300	00:01:26	100.00%	100.00%	0.0002	0.0651	1.0000e-04

(a)



(b)

Figure 8. The performance of the proposed system for (a) training on single GPU and (b) training progress

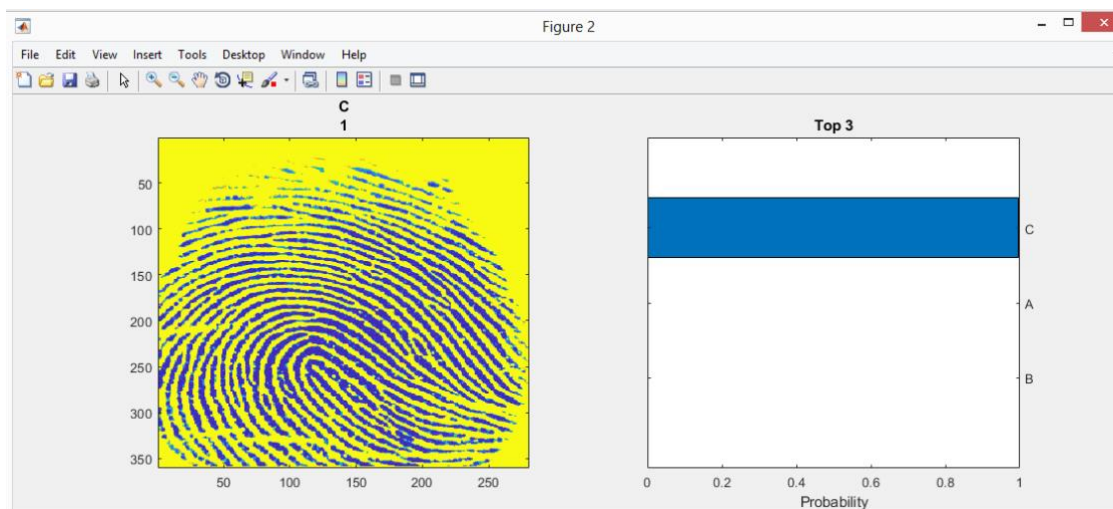


Figure 9. Fingerprint classification

5. CONCLUSION

In this paper, a deep learning framework for fingerprint recognition using CNN to learn the representation of the features and perform the recognition. The proposed model has been trained on a large-scale fingerprint recognition dataset. The successful simulation results have been indicated that the proposed system can be applied as a reliable system for developing artificial identification and can be adapted to any image contents of different characteristics. The accuracy of the proposed system according to training model was 100% for training dataset and 100% for validation dataset. Therefore, the proposed system is robust, reliable, efficiency. In future research, CNN structure optimization will be study to speed up the learning speed for fast matching and to improve the recognition performance of images with noise, to apply multi modal biometric recognition and implement in real time using Raspberry pi and fingerprint sensor.




REFERENCES

- [1] R. Wang, C. Han, Y. Wu, and T. Guo, "Fingerprint classification based on depth neural network," 2014, [Online]. Available: <http://arxiv.org/abs/1409.5188>.
- [2] D. Peralta, I. Triguero, S. García, Y. Saeys, J. M. Benitez, and F. Herrera, "On the use of convolutional neural networks for robust classification of multiple fingerprint captures," *International Journal of Intelligent Systems*, vol. 33, no. 1, pp. 213–230, 2018, doi: 10.1002/int.21948.
- [3] U. Park, S. Pankanti, and A. K. Jain, "Fingerprint verification using SIFT features," in *Biometric Technology for Human Identification V*, 2008, vol. 6944, doi: 10.1117/12.778804.
- [4] R. Cappelli, M. Ferrara, and D. Maltoni, "Minutia cylinder-code: a new representation and matching technique for fingerprint recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 12, pp. 2128–2141, 2010, doi: 10.1109/TPAMI.2010.52.
- [5] S. R. Patil and S. R. Suralkar, "Fingerprint classification using artificial neural network," *International Journal of Scientific and Advanced Engineering*, vol. 2, no. 12, pp. 1–4, 2012.
- [6] W. Yang, S. Wang, J. Hu, G. Zheng, and C. Valli, "Security and accuracy of fingerprint-based biometrics: a review," *Symmetry*, vol. 11, no. 2, Jan. 2019, doi: 10.3390/sym11020141.
- [7] S. Minaee, A. Abdolrashidi, H. Su, M. Bennamoun, and D. Zhang, "Biometrics recognition using deep learning: a survey," Nov. 2019, [Online]. Available: <http://arxiv.org/abs/1912.00271>.
- [8] S. Minaee, E. Azimi, and A. Abdolrashidi, "FingerNet: pushing the limits of fingerprint recognition using convolutional neural network," Jul. 2019, [Online]. Available: <http://arxiv.org/abs/1907.12956>.
- [9] R. Das, E. Piciuccio, E. Maiorana, and P. Campisi, "Convolutional neural network for finger-vein-based biometric identification," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 2, pp. 360–373, 2018, doi: 10.1109/TIFS.2018.2850320.
- [10] C. Militello, L. Rundo, S. Vitabile, and V. Conti, "Fingerprint classification based on deep learning approaches: experimental findings and comparisons," *Symmetry*, vol. 13, no. 5, p. 750, Apr. 2021, doi: 10.3390/sym13050750.
- [11] B. Pandya, G. Cosma, A. A. Alani, A. Taherkhani, V. Bharadi, and T. M. McGinnity, "Fingerprint classification using a deep convolutional neural network," in *2018 4th International Conference on Information Management, ICIM 2018*, 2018, pp. 86–91, doi: 10.1109/INFOMAN.2018.8392815.
- [12] B. K. O. C. Alwawi and L. H. Abood, "Convolution neural network and histogram equalization for COVID-19 diagnosis system," *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 24, no. 1, pp. 420–427, Oct. 2021, doi: 10.11591/ijeecs.v24.i1.pp420-427.
- [13] B. K. O. and M. I. AL Tameemi, "Visual-infrared video monitoring system for moving object detection, classification and tracking using deep learning technique," *8th Engineering and 2nd International Conference for College of Engineering*, 2021.
- [14] B. K. Oleivi and A. F. Althabhaawe, "Towards more accurate and efficient human iris recognition model using deep learning technology," *TELKOMNIKA Telecommunication, Computing, Electronics and Control*, 2021.
- [15] H. J. A. A. T. H. A. M. Hasan, "Segmentation approach for a noisy iris images sased on hybrid techniques," *Engineering and*




- Technology Journal*, vol. 38, pp. 1684–1691, 2020, doi: 10.30684/etj.v38i11A.450.
- [16] A. T. Hashim and D. A. Noori, “An approach of noisy color iris segmentation based on hybrid image processing techniques,” in *Proceedings - 2016 International Conference on Cyberworlds, CW 2016*, 2016, pp. 183–188, doi: 10.1109/CW.2016.39.
- [17] A. Nasser and H. Sever, “A concept-based sentiment analysis approach for Arabic,” *International Arab Journal of Information Technology*, vol. 17, no. 5, pp. 778–788, 2020, doi: 10.34028/iajit/17/5/11.
- [18] A. Abdulhussein and F. Raheem, “Hand gesture recognition of static letters american sign language (ASL) using deep learning,” *Engineering and Technology Journal*, vol. 38, no. 6, pp. 926–937, 2020, doi: 10.30684/etj.v38i6a.533.
- [19] B. K. O. and M. R. Kadhim, “Real time embedded system for object detection using deep learning,” *The 2nd International Conference on Electromechanical Engineering & its Applications, ICEMEA*, 2021.
- [20] M. R. K. and B. K. Oleiwi, “Blind assistive system based on real time object recognition using machine learning,” *Engineering and Technology Journal*, an accepted paper, 2021.
- [21] S. Mahmood and L. Saud, “An efficient approach for detecting and classifying moving vehicles in a video based monitoring system,” *Engineering and Technology Journal*, vol. 38, no. 6, pp. 832–845, 2020, doi: 10.30684/etj.v38i6a.438.
- [22] Zkteco.systems, “Zk4500.” <https://zkteco.systems/en/product/zk4500/>.
- [23] J. Murphy, “An overview of convolutional neural network architectures for deep learning,” *Microway*, 2016, doi: 10.20944/preprints201811.0546.v1.
- [24] M. Thoma, “Analysis and optimization of convolutional neural network architectures,” Jul. 2017, [Online]. Available: <http://arxiv.org/abs/1707.09725>.
- [25] Mathworks.com “Convolutional neural network,” *Mathworks*, 2021 [Online]. Available: <https://www.mathworks.com/discovery/convolutional-neural-network-matlab.html>.
- [26] R. Vidal, J. Bruna, R. Giryes, and S. Soatto, “Mathematics of deep learning,” 2017, [Online]. Available: <http://arxiv.org/abs/1712.04741>.
- [27] Mathworks.com “Deep learning toolbox,” *Mathworks.com*, 2021 [Online]. Available: <https://www.mathworks.com/products/deep-learning.html>.

BIOGRAPHIES OF AUTHORS



Ali Fadhil Yaseen Althabhawee    born in Karbala-Iraq, completed B.Sc. degree in Computer Engineering at Alhussain University college, Karbala-Iraq. He finished his M.Sc. degree in Computer Engineering at University of Technology, Bagdad-Iraq. His research interests are artificial intelligence systems and deep learning and machine learning and programming languages. He is currently working as computer engineer at the Ministry of Education Directorate in Iraq. He can be contacted at email: 61124@student.uotechnology.edu.iq



Bashra Kadhim Oleiwi Chabor Alwawi    completed a Master degree in Mechatronics Engineering/Control and Systems Engineering Department at University of Technology (UOT) Bagdad-Iraq. She finished Ph.D. student at Control Engineering Department (RST), Siegen University, Germany. Her research interests are mobile robot, path planning, multi objective optimization and artificial intelligence systems and deep learning and machine learning. She is currently working as a faculty member in control and systems engineering department at UOT. She can be contacted at email: 60010@uotechnology.edu.iq