International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-13 Issue-5, April 2024

Enhancing Arabic Sign Language Recognition using Deep Learning

Noor S. Sagheer, Faezah Hamad Almasoudy, Manar Hamza Bashaa

Abstract: ***In the present time, Sign language is essential for people who suffer from hearing loss or who cannot speak. Normal humans tend to overlook the significance of sign language, which is a vital means of communication for deaf and mute individuals. This study proposes a developed model for sign language recognition in Arabic using the Deep learning Convolutional Neural Network (CNN) algorithm. Then, set up the algorithm by developing a program using OpenCV and the Python language. The dataset contains 54049 snapshots of Arabic signal language alphabets. The 32 folders were created, each containing 1,500 images that incorporated hand gestures in various environments. The dataset was divided into a training set (70%), a testing set (20%), and a validation set (10%). The results show that the suggested model achieved an accuracy rate of 94.8%, demonstrating its effectiveness and success, particularly after being tried and tested by several users and receiving their comments and feedback.***

Keywords: ***Arabic Language, CNN classification, Deep learning, Image Classification.***

I. INTRODUCTION

Sign language is a language that uses visible manual signs to convey meaning, rather than relying solely on spoken words. Sign language is expressed through a combination of manual and non-manual signs. Unlike different natural languages, it utilises large physical motions to convey messages, known as markers or gestures. To speak a message, hand and finger gestures, head nodding, shoulder gestures, and facial expressions are employed [1]. So, the planned paintings might help deaf individuals interact with others who are deaf, as well as with hearing individuals. Once the deaf or hard-of-hearing person attempts to communicate, they are using signs to convey their thoughts and ideas. Each image signifies an extraordinary letter, word, or emotion. Every image represents a remarkable word, letter, or sign. The word is fashioned with the aid of a signal mixture, just as the string of phrases consists of phrases in the spoken languages. Thus, the language of signal is a fashionably everyday language with the structure of a sentence and grammar [2]. The CNN algorithm can be applied in various fields and performs basic tasks such as document analysis, face recognition, climate understanding, image recognition, object identification, and other features. Deep learning has revolutionised the learning algorithms category, enabling the explanation of complex systems by combining several nonlinear modifications [3]. The critical aspect of constructing deep learning blocks using neural networks is related to building deep neural networks. These strategies have assisted in giant development in the processing of sound and photo, computer vision, processing of computerized language, encompassing face identification, recognition of the voice, and various fields such as genomics and drug prognosis [4]. The applications of deep learning are numerous; one of them is the ability to utilise computational algorithms with multiple processing layers to accumulate representations of various abstracted dimensions. Based on that, a real-time Sign Language Recognition (SLR) system for the Arabic alphabet, utilising a camera, will be built. Then, set up the algorithm by developing a program using OpenCV and the Python language.

II. RELATED WORKS

Sign language is a natural and effective method for interaction between non-English-speaking people and the hearing-impaired [5]. In [6] The author developed a system using a vision-based CNN to recognise Arabic letters based on hand signs and translate them into speech. For each sign of the hand, a training set of 100 images and a test set of 25 images are also created for the ArSL of 31 letters, and the system achieves an accuracy higher than 90%. In [7] The author introduced a new system of ArSL recognition using faster R-CNN, which can recognise and localise the ArSL alphabet. Specifically, Faster R-CNN is designed to map and extract features from the image and learn the hand position in the assumed image. The models of ResNet-18 and VGG-16 are exploited, and the accuracy of the projected architecture is 93%. In article [8] Researchers proposed a framework based on the Self-Organising Map (SOM), combining DeepLabv3 semantic segmentation with a network of Bidirectional extended short-range memory for ArSL recognition. The problem of hand segmentation is explained using a model of DeepLabv3C, which relies on a ResNet-50-like backbone encoder and an atrous localisation pyramid.

Manuscript received on 16 March 2024 | Revised Manuscript received on 15 April 2024 | Manuscript Accepted on 15 April 2024 | Manuscript published on 30 April 2024.

**Correspondence Author(s)*
Noor S. Sagheer*, Department of Computer Science, College of Computer Science and Information Technology, Kerbala University, Kerbala, Iraq. Email: noor.sabah@uokerbala.edu.iq, ORCID ID: 0000-0003-1167-592X.
Faezah Hamad Almasoudy, Department of Animals Production, College of Agriculture, Kerbala University, Kerbala, Iraq. Email: feazah.h@uokerbala.edu.iq, ORCID ID: 0009-0002-7739-3327.
Manar Hamza Bashaa, Department of Computer Science, College of Computer Science and Information Technology, Kerbala University, Kerbala, Iraq. Email: manar.hb83@gmail.com, ORCID ID 0000-0002-8824-9112.

The framework accuracy is 89.5% using DeepLabv3C segmentation, but it falls by 69.0% without the hand segmentation module. In article [9], they proposed an offline system to recognise Arabic signs for both letters and numbers based on a deep CNN, fed with the original data. They utilised an image dataset comprising 5,839 images for 28 letters and 2,030 images for numbers, and the proposed system achieved an accuracy of approximately 90.02%. In article [10], researchers introduced a model that utilises deep-CNN and transfer learning by using ResNet152 and VGG16 network models to recognise 32 hand gestures from ArSL, which are already trained, and then apply fine-tuning using the ArSL dataset. The model yielded a result with approximately 99% accuracy. In [11], researchers proposed a model using AdaBoosting to recognise signs in the Arabic language. They utilised two algorithms (KNN and SVM) and AdaBoosting to enhance the accuracy of these algorithms. Also, they used the DTW technique to compare the results with AdaBoosting. The model applied to the dataset included 20 gestures performed with a single hand and 10 gestures performed with both hands. The introduced model achieves an accuracy of 92.3% for single-hand gestures and 93% for double-hand gestures, while the recognition rate of DTW gestures is 88% for single-hand gestures and 86% for double-hand gestures.

III. MACHINE LEARNING CLASSIFICATION

Classification is a type of unsupervised learning [12]. Machine learning is a sophisticated division of computational algorithms designed to simulate human intelligence through the study of the environment. Classification is a technique that identifies unfamiliar elements [13]. Unknown elements are identified by comparing them with previously recorded patterns. If the comparison is successful, the unknown object will be recognized [14].

IV. CONVOLUTIONAL NEURAL NETWORK ALGORITHM (CNN)

CNN is a type of special feed-forward neural network in artificial intelligence. It is typically used in analysing visual scenes and computer vision applications, as it is considered a solution for various computer vision problems, such as image and video processing. CNNs are primarily used for image recognition and classification to discover and process objects, and are specifically designed to process pixel data [15].

CNNs are primarily used to classify images and categorise them based on their similarity, and then perform object recognition. CNN employs a system similar to a multi-layered perspective, designed to reduce processor requirements. CNN consists of several layers, including the output layer, the input layer, the hidden layer that includes multiple convolutional layers, fully connected layers, an activation function, and finally a pooling layer [16]-[17].

V. PROPOSED MODEL AND WORKING FOLLOW

The main aim is to recognise sign language for Arabic using a Deep learning algorithm, specifically a CNN. In Figure 1, the steps of the proposed study are given. We will call the dataset and do a set of pre-processing operations. Then, we elicit the features from the images and split the dataset into training and testing sets. Then, we use a training set to train a model to classify the images, use the test set to calculate accuracy, and then evaluate the model.

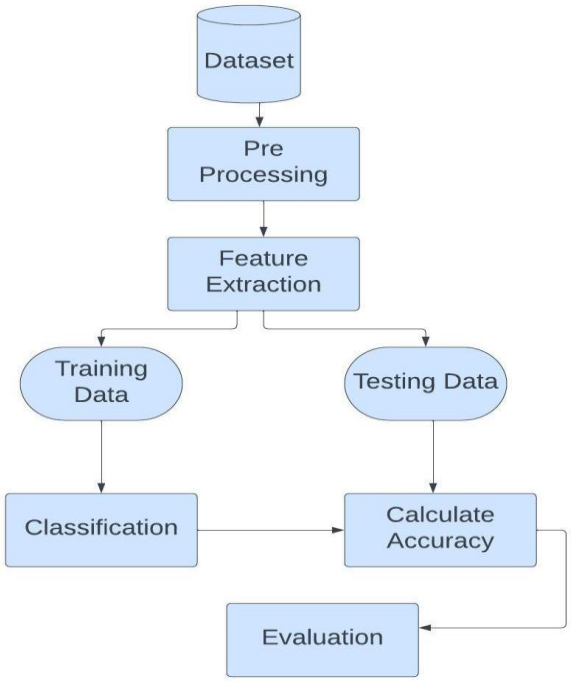


Fig. 1. Proposed Model

A. Dataset

The dataset comprises 54,049 snapshots of Arabic sign language alphabets, collected with the assistance of over 40 individuals for 32 general Arabic signs and alphabets. The dataset is available at ArSL2018. [18], released with the aid of Prince Mohammad Bin Fahd University, to be open to researchers. Each wonderful hand gesture shows little significant information. There are approximately 1,500 snapshots per class, and each class represents an exceptional means by using its hand gesture or signal. The pattern photograph of every class and its label is explained in Figure (2).

The number of folders created was 32, and each one consisted of 1,500 images incorporating hand gestures in different environments. A directory is treated as a training and validation dataset for the model.

Fig. 2. Dataset Overview

Retrieval Number: 100.1/ijitee.E984413050424
DOI: 10.35940/ijitee.E9844.13050424
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

B. Implementation Requirements

- Colab is a product from Google Research that allows arbitrary Python code to be executed and written through the browser, particularly suitable for machine learning, education, and data analysis [19].
- Python is one of the most well-known and high-level programming languages. This language is being used in web development, Machine Learning applications, and all cutting-edge technology in the Software Industry. The Python Programming Language is very well suited for Beginners, as well as for experienced programmers with other programming languages like C++ and Java [20].

C. Implementation Steps

- Call the needed libraries to work
- Load the dataset containing the image ID and the path of each image
- Execute the pre-processing operation to check if the dataset contains empty or missing values.
- Load class labels in Arabic and English
- Show the unique values, which are the names of signs in Arabic and English
- Dispersion and shape of the dataset
- Distribution using a bar plot.
- The data set was split into training, testing, and validation sets, with the training set comprising 70% of the data, the testing set containing 20% of the data, and the validation set containing 10% of the data.

D. Building the Model

In the CNN model layers, every layer consists of a positive number of neurons. There are max-pooling layers with a size of 2x2. The photos are 64 x 64 pixels in size. We reshape it so that it is of length 64 x 64 x 1, and feed this as enter to the network. Additionally, we utilised the activation function' relu' for the primary layers. The output layer is changed to 32 classes. For the output layer, we employed the softmax activation function to classify 32 classes. The neurons are equal to the courses in the output layer. We used the Adam optimiser for hand gesture recognition. The Adam optimiser is used after every epoch to reduce the loss calculated during training.

Additionally, we employed a loss function called 'categorical-cross-entropy'. The aim of using the loss function is to calculate the losses of training and/or validation after every epoch in the training process. A model's performance is degraded by a boom in loss and optimised by a reduction in loss. A class "categorical-cross-entropy" computes the loss of cross-entropy between actual and predicted values as a result of model predictions when the image type is greater than the classes to predict.

Training call-backs save the weights of a model after specified periods during the training process. The instance of a model is fitted using the function fit() to initiate the training process. The benefit of this function is that it trains the model over a fixed number of epochs using both training and validation data. The model had been trained for 35 epochs and is shown in Figure 3.

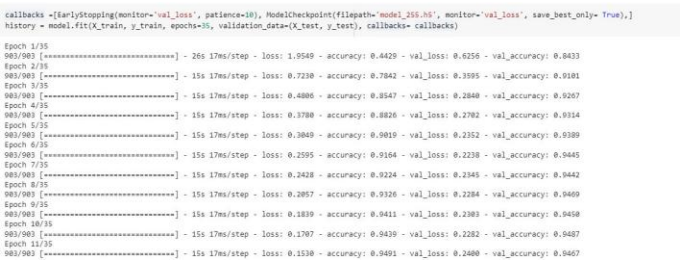


Fig. 3. Training Model

E. Model Evaluation

- **Training and Validation Accuracy Chart**: From the Keras model history, you can obtain each plotted curve, which computes both loss and accuracy for each epoch that the network went through. Accuracy is computed by comparing the predicted class to the actual class. Figure 4. The cross-entropy value between the predicted class and the actual class calculates the loss. After training the model, we achieved an accuracy of 94.8%. The following chart represents the accuracy obtained from the trained model. The blue line denotes the accuracy of the training set during each epoch. The Brown-line indicates the accuracy of the verification set during each epoch. We can see that the model is trained very well because the two lines are close to each other.

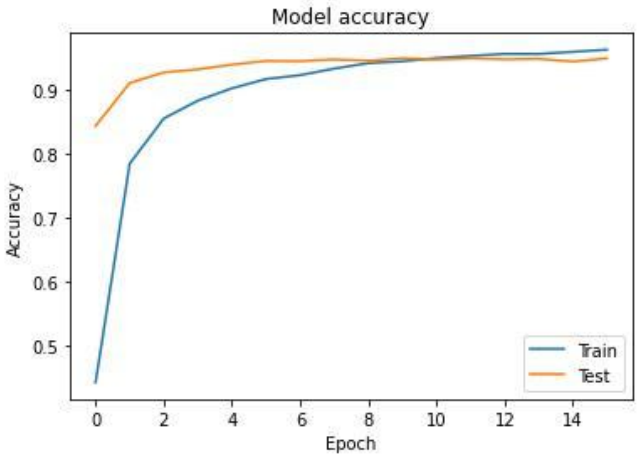


Fig. 4. Accuracy Chart

- **Training and Validation Chart:** In Figure 5, the blue line indicates the loss of the training set during each epoch, and the Brown line represents the loss of the validation set during each epoch. From the loss plot, we observe that the model performs similarly on both the training dataset and the validation dataset (labelled as test). The training may be stopped at a previous epoch if the parallel plots start departing continuously. It is evident that the accuracy shown in Figure 4 and the loss shown in Figure 5 exhibit a trend of increasing difference up to the 15th epoch. Both accuracy and loss diverge after the 14th epoch, indicating that the model has fully learned the weights. Consequently, at the sixteenth epoch, a model stops to avoid overfitting because a model has recognised the features of the input to classify the unexpected gestures of the hand better.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

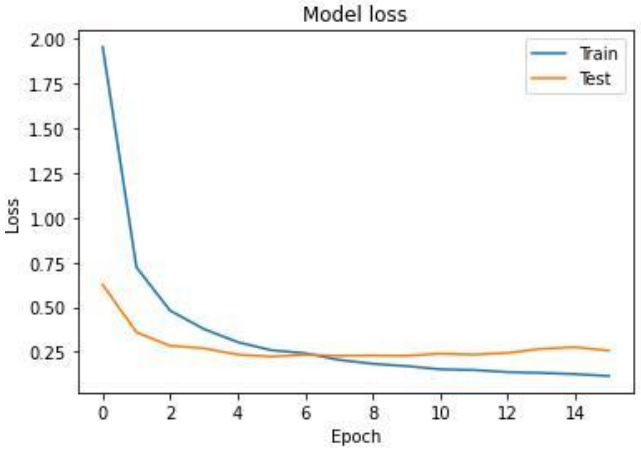


Fig. 5. Loss Chart

▪ **Confusion Matrix:** After training the model, a test is necessary to determine the model's actual performance on unseen data that has not been encountered by the model yet. The statistical evaluation is accomplished through the usage of a matrix of confusion; the matrix of confusion is represented in Figure (6).

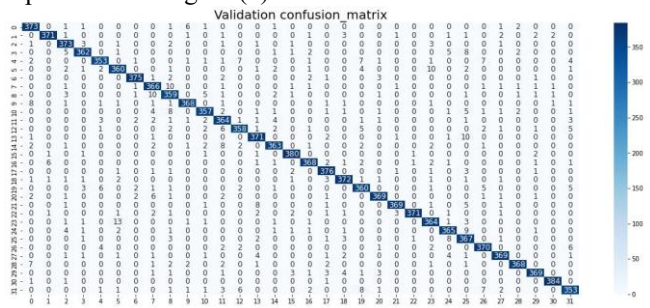


Fig. 6. Confusion Matrix

▪ **Show the Predicted Value:** The predicted value is the value that the model predicts will occur. To obtain it, we pre-processed the previous images and then entered them into the model to predict them, as in Figure (7) in English.

Fig. 7. Predicted Value English

The Figure (8) shows the value in Arabic.

Fig. 8. Predicted Value Arabic

VI. SYSTEM DESIGN IN REAL TIME

▪ **Initialise Media-Pipe:** The algorithm for hand recognition is performed by the mp.solution.hands module. The model is configured by using the mpHands to store the object after it is created. Additionally, Media-Pipe can identify multiple hands in a single frame. Detected key points will be drawn for us by MP Solutions. Drawing Utils. In this case, we do not need to pull them by hand.

▪ **Read Frames from a Webcam:** We create a Video-Capture object and pass an argument (0). The cap.read ( ) function reads each frame from the webcam. The cv2.imshow() function shows the frame in a new OpenCV window. The cv2.waitKey ( ) function keeps the window open until the key (q) is pressed. Media-Pipe handles images in RGB format, whereas OpenCV reads images in BGR format. So, we use the function cv2.cvt-COLOR ( ) to convert the frame into RGB format. The process function takes an RGB frame and returns the result class. After that, we used the result.The multi-hand_landmarks method is used to determine if any hands are identified or not. Then we circle through each disclosure and keep the coordinates on the list named landmarks. Since the model yields normalised outcomes, the result is multiplied by the height of the image (y) and the width of the image (x). This means the values in outcomes are between (0, 1). Ultimately, we utilise the mpDraw.draw-landmarks ( ) function for drawing all the landmarks in the frame.

VII. RESULTS

We notice that when someone makes their hand like in picture (a) in Figure (9), the proposed model directly assigns the letter (al). Also in the picture (b) in Figure (90, the proposed model directly assigns the letter (aleff). Figure (10.a) shows the letter (bb) result, and Figure (10.b) shows the letter (dal) result.

Retrieval Number: 100.1/ijitee.E984413050424
DOI: 10.35940/ijitee.E9844.13050424
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

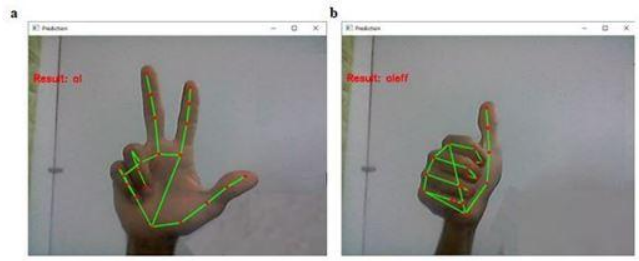


Fig. 9. (a) Show Result al; (b) Show Result Aleff

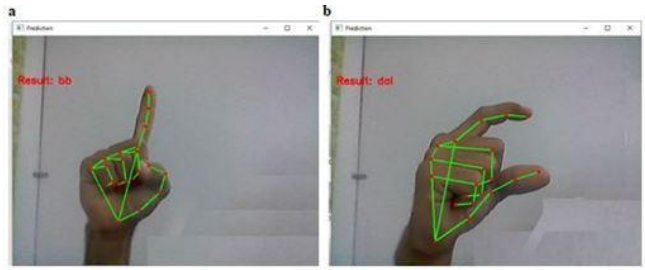


Fig. 10. (a) Show Result Bb; (b) Show Result Dal

VIII. CONCLUSION

Deaf and unspeaking individuals use sign language to connect with people who can hear. Therefore, we proposed a model that is both inexpensive and straightforward. The snapshots are about 54049, used to train, test, and validate the proposed model. A real-time sign language recognition system for the Arabic alphabet, utilising a camera, was developed. Then, set up the algorithm by creating a program using OpenCV and the Python language. The accuracy of the proposed model is 94.8%. After using the Deep Learning CNN algorithm for Sign Language Recognition of Arabic, we found that most users who tried it liked the system, its ease of use, and its appearance, with positive feedback.

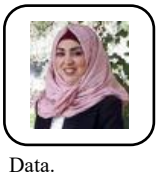
DECLARATION STATEMENT

Funding	No, I did not receive.
Conflicts of Interest	No conflicts of interest to the best of our knowledge.
Ethical Approval and Consent to Participate	No, the article does not require ethical approval or consent to participate, as it presents evidence that is not subject to interpretation.
Availability of Data and Materials	Not relevant.
Authors Contributions	All authors have equal participation in this article.

REFERENCES

1. R. Rastgoo, K. Kiani, and S. Escalera, "Sign language recognition: A deep survey," *Expert Systems with Applications,* vol. 164, p. 113794, 2021. https://doi.org/10.1016/j.eswa.2020.113794
2. R. Kushalnagar, "Deafness and hearing loss," *Web Accessibility: A Foundation for Research,* pp. 35-47, 2019. https://doi.org/10.1007/978-1-4471-7440-0_3.
3. A. Mathew, P. Amudha, and S. Sivakumari, "Deep learning techniques: an overview," *Advanced Machine Learning Technologies and Applications: Proceedings of AMLTA 2020,* pp. 599-608, 2021. https://doi.org/10.1007/978-981-15-3383-9_54.
4. O. I. Abiodun, A. Jantan, A. E. Omolara, K. V. Dada, N. A. Mohamed, and H. Arshad, "State-of-the-art in artificial neural network applications: A survey," *Heliyon,* vol. 4, no. 11, 2018. https://doi:10.1016/j.heliyon.2018.e00938.
5. M. J. Cheok, Z. Omar, and M. H. Jaward, "A review of hand gesture and sign language recognition techniques," *International Journal of Machine Learning and Cybernetics,* vol. 10, pp. 131-153, 2019. https://doi.org/10.1007/s13042-017-0705-5.
6. M. Kamruzzaman, "Arabic sign language recognition and generating Arabic speech using a convolutional neural network," *Wireless Communications and Mobile Computing,* vol. 2020, 2020. https://doi.org/10.1155/2020/3685614.
7. R. A. Alawwad, O. Bchir, and M. M. B. Ismail, "Arabic sign language recognition using faster R-CNN," *International Journal of Advanced Computer Science and Applications,* vol. 12, no. 3, 2021. DOI: 10.14569/IJACSA.2021.0120380.
8. S. Aly and W. Aly, "DeepArSLR: A novel signer-independent deep learning framework for isolated Arabic sign language gestures recognition," *IEEE Access,* vol. 8, pp. 83199-83212, 2020. . http://doi:10.1109/ACCESS.2020.2990699.
9. S. Hayani, M. Benaddy, O. El Meslouhi, and M. Kardouchi, "Arab sign language recognition with convolutional neural networks," in *2019 International conference of computer science and renewable energies (ICCSRE)*, 2019, pp. 1-4: IEEE. http://doi: 10.1109/ICCSRE.2019.8807586
10. Y. Saleh and G. Issa, "Arabic sign language recognition through deep neural networks fine-tuning," 2020. https://doi.org/10.3991/ijoe.v16i05.13087
11. B. Hisham and A. Hamouda, "Arabic sign language recognition using Ada-Boosting based on a leap motion controller," *International Journal of Information Technology,* vol. 13, pp. 1221-1234, 2021. https://doi.org/10.1007/s41870-020-00518-5
12. F. H. Almasoudy, W. L. Al-Yaseen, and A. K. Idrees, "Differential evolution wrapper feature selection for intrusion detection system," *Procedia Computer Science,* vol. 167, pp. 1230-1239, 2020. https://doi.org/10.1016/j.procs.2020.03.438
13. N. S. Sagheer and S. A. Yousif, "Canopy with k-means clustering algorithm for big data analytics," in *AIP Conference Proceedings*, 2021, vol. 2334, no. 1: AIP Publishing. https://doi.org/10.1063/5.0042398
14. I. El Naqa and M. J. Murphy, *What is machine learning?* Springer, 2015. https://doi.org/10.1007/978-3-319-18305-3_1
15. S. Indolia, A. K. Goswami, S. P. Mishra, and P. Asopa, "Conceptual understanding of convolutional neural network-a deep learning approach," *Procedia computer science,* vol. 132, pp. 679-688, 2018. https://doi.org/10.1016/j.procs.2018.05.069
16. K. Shridhar, F. Laumann, and M. Liwicki, "A comprehensive guide to Bayesian convolutional neural network with variational inference," *arXiv preprint arXiv:1901.02731,* 2019. https://doi.org/10.48550/arXiv.1901.02731
17. M. K. Singh, V. Kekatos, and G. B. Giannakis, "Learning to solve the AC-OPF using sensitivity-informed deep neural networks," *IEEE Transactions on Power Systems,* vol. 37, no. 4, pp. 2833-2846, 2021. DOI:10.1109/TPWRS.2021.3127189
18. G. Latif, N. Mohammad, J. Alghazo, R. AlKhalaf, and R. AlKhalaf, "ArASL: Arabic alphabet sign language dataset," *Data in brief,* vol. 23, p. 103777, 2019. https://doi.org/10.1016/j.dib.2019.103777
19. W. Vallejo, C. Díaz-Uribe, and C. Fajardo, "Google colab and virtual simulations: practical e-learning tools to support the teaching of thermodynamics and to introduce coding to students," *ACS omega,* vol. 7, no. 8, pp. 7421-7429, 2022. https://doi.org/10.1021/acsomega.2c00362
20. J. Fontenrose, *Python: A Study of Delphic Myth and Its Origins*. Univ of California Press, 2023.

AUTHORS PROFILE



Noor Sabah Sagheer completed a B.Sc. in computer science at Kerbala University and a Master's degree in Artificial Intelligence at Al-Nahrain University. She has had a total of 14 years of Experience, including 11 years in teaching, with research interests in Artificial Intelligence, Machine Learning, Deep Learning, and Big Data.

Faezah Hamad Almasoudy completed a Bachelor's degree in Computer Science at Kerbala University and a Master's degree in Networks at Babylon University. She has had a total of 14 years of Experience, 6 years in teaching. Her research interests include computer networks, Artificial Intelligence, Machine Learning, and Deep Learning.

Manar Hamza Bashaa is a Computer Science student at Babylon University. She has a strong enthusiasm for computer networks and security and has delved into various domains, including computer vision and image analysis.