International Journal of Emerging Science and Engineering (IJESE)
ISSN: 2319–6378 (Online), Volume-12 Issue-10, September 2024

CNN Algorithm with SIFT to Enhance the Arabic Sign Language Recognition

Manar Hamza Bashaa, Faezah Hamad Almasoudy, Noor S. Sagheer, Wasan Mueti Hadi

***Abstract:** Sign language is used as a primary means of communication by millions of people who suffer from hearing problems. The unhearing people used visual language to interact with each other, Represented in sign language. There are features that the hearing impaired use to understand each other, which are difficult for normal people to understand. Therefore, deaf people will struggle to interact with society. This research aims to introduce a system for recognizing hand gestures in Arabic Sign Language (ArSL) through training the Convolutional Neural Network (CNN) on the images of ArSL gestures launched by the University of Prince Mohammad Bin Fahd, Saudi Arabia. A Scale Invariant Feature Transform (SIFT) algorithm is used for creating the feature vectors that contain shape, finger position, size, center points of palm, and hand margin by extracting the Important features for images of ArSL and transforming them to points of the vector. The accuracy of the proposed system is 97% using the SIFT with CNN, and equal to 94.8% nearly without SIFT. Finally, the proposed system was tried and tested on a group of persons and its effectiveness was proven after considering their observations.*

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

I. INTRODUCTION

The gestures are the names that refer to the signs' non-verbal and body language. Their use is widely and common as the communication means in the world. Even using mobile phones, some people continue to use gestures in their daily lives. Most of the inhabitants of the Earth today depend on technology for most of their affairs, even when communicating with each other. This has increased the spread of knowledge of gestures, which has contributed to the interaction of deaf and mute people with society.

The increasing proportion of deaf and mute people in the world; and the increasing use of virtual reality and video games have led to an increase in demand for designing systems to identify gestures [1]. In the world for each language, there are many sign languages. These languages utilized more than 200 languages today like American, British, Spanish, Chinese, and ArSL. Finding a way to enable hearing and non-hearing communities to communicate with each other is so important. The automatic recognition technique is the latest method used to understand the signs of the deaf, replacing the help of an expert. This method allows signals to be interpreted into text and audio to meet the user's needs [2]. Artificial Intelligence (AI) methods such as Machine Learning (ML) and data mining (clustering and classification) are developing and widespread [3, 4]. ML expresses the system's capacity for learning from training data of the problem-specific to automating analytical system-building processes and solving related tasks. Deep learning is a concept of ML established on networks of artificial neural [5]. In the field of deep learning, the most common network is CNN. CNN has achievements in most areas such as processing of the natural language and computer vision [6]. To increase the accuracy of the recognition and reduce the time of computing, this paper uses an algorithm of feature extraction for allocating the letters in the images called the SIFT algorithm. The regions that are applicants discover distinct points in images and are then placed into a system of CNN recognition [7]. The paper's rest structure is as follows: Section II contains the information on related works. Section III gives details about the framework of the proposed model. Section IV presents the results and finally, section V explains the conclusion.

II. RELATED WORKS

In this section, we introduce previous methods used in sign language recognition. In [8], they proposed a method to recognize the word in ArSL utilizing a device of Leap Motion which helps by using infrared to build a 3D human hand model. They derived the mathematical features from the controller of Leap Motion and focused on analyzing them. The results show for gestures of the one hand, 89% had the highest rate of classification. In [9] a hybrid model for capturing (letters and words). This model includes Long Short-Term Memory (LSTM) for extracting temporal and spatial characteristics for handling the movements of the hand and CNN for extracting from the data of sign language of the spatial features. CNN classifier achieved an accuracy rate of 94.40%, while the LSTM classifier achieved an accuracy rate of 82.70%.

Manuscript received on 24 August 2024 | Revised Manuscript received on 02 September 2024 | Manuscript Accepted on 15 September 2024 | Manuscript published on 30 September 2024.
*Correspondence Author(s)

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

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

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

Wasan Mueti Hadi, Department of Computer Science, College of Computer Science and Information Technology, Kerbala University, Kerbala, Iraq. Email: wasan.m@uokerbala.edu.iq, ORCID ID: 0009-0007-5275-6431

In [10] researchers design ASODCAE-SLR as a technique for helping deaf and unhearing people through the process of SLR. This technique employed CapsNet to produce a feature vector collection. They furthermore used a DCAE model to recognize sign language and an ASO algorithm to increase the DCAE model efficacy. The researcher [7] presented the SIFT algorithm which locates objects in the satellite image. The image is cropped to get the region of the object from the features of the image. Then, this image is put into the recognition system CNN. The accuracy of the SIFIT algorithm is 70% for object recognition as ships and aircraft. The researcher has proposed a model based on the CNN algorithm to recognize the ArSL [11] [23]. The algorithm is set using Python language. The dataset includes images of hand gestures in different environments. The accuracy was 94.8% with good feedback from several users. In [12] they designed a model containing LSTM units for encoding the sequential data incorporated with Media Pipe to deal with the input sequence of recorded gestures. The dataset includes fifty numbers and words frequently used, that are obtained from the five volunteers. The accuracy of this model for individual volunteers was 85% while for combined data 83%.

III. FRAMEWORK FOR ARSL MODEL

There are four stages in the framework of the proposed model. The first stage includes the details of acquired image data. The second stage involves two tasks in the image data: the preprocessing and the segmentation. The algorithm of the SIFT and CNN model is applied in the third stage. Finally, the fourth stage applies the trained model. Figure (1) shows the framework of the proposed model with phases declared as follows:

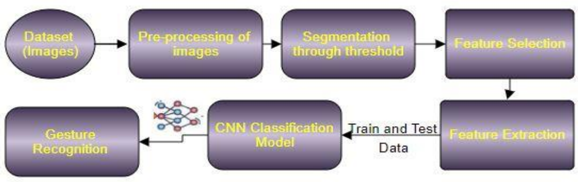


Fig. 1: The Framework of the Proposed Model

A. Dataset

There are 54049 snap-shots of the ArSL alphabet from forty volunteers including 32 alphabet and symptoms. The datasets are available at ArSL2018 [13], which was issued in cooperation with the University of Prince Mohammad Bin Fahd to be available to researchers. Each gesture of the hand shows important information. Figure (2) explains a pattern photograph with its label of every class. The 32 folders were created, and everyone included 1500 images of hand gestures in variant environments.

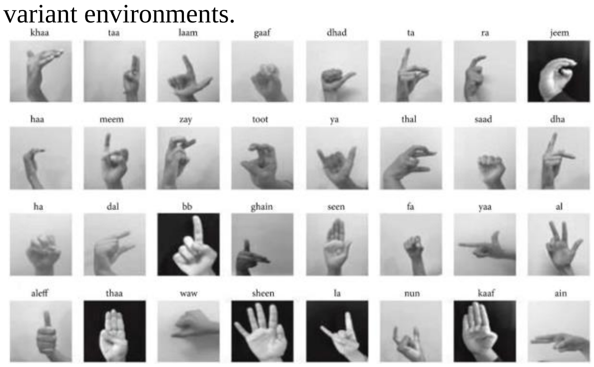


Fig. 2: ArSL Alphabetic and Their Identical Hand Sign Shape

B. Preprocessing and Segmentation

The significant stage in recognizing the hand gesture is a preprocessing of the hand and its segmentation from the image. The segmentation purpose is to isolate the hand signer from a backdrop. Colored image segmentation is a difficult process for the reason that it is difficult to locate the correct values of thresholding for both color saturation and light sensitivity. For simplification, we convert the images from color to grayscale. Grayscale images eliminate the complex computational requirements related to the algorithms. The luminosity method is also named the grayscale conversion [14]. Equation (1) explains the conversion method from the RGB images to grayscale.

$$Gs = 0.299R + 0.587G + 0.114B \quad (1)$$

Gs represent Grayscale, and R, G, and B represent (red, green, and blue) colors respectively.

Also, to enhance the contrast values, the histogram equalization pre-processing technique is practical for the images. Furthermore, segmentation is applied to the data of the image by thresholding the image to locate the pixels that form the region of the hand [15].

C. Selection and Extraction of Features

The feature extraction stage is necessary for several reasons. First, in the recognition process, some features do not participate in recognition accuracy because they are irrelevant. Second, the accuracy and precision are affected by using fewer features. Therefore, this stage is important and requires careful study to achieve optimal system performance. The SIFT algorithm extracts relevant features (size of hand, finger position, Margin or edge, palm center) from a dataset. Figure (3) shows the points of these features.

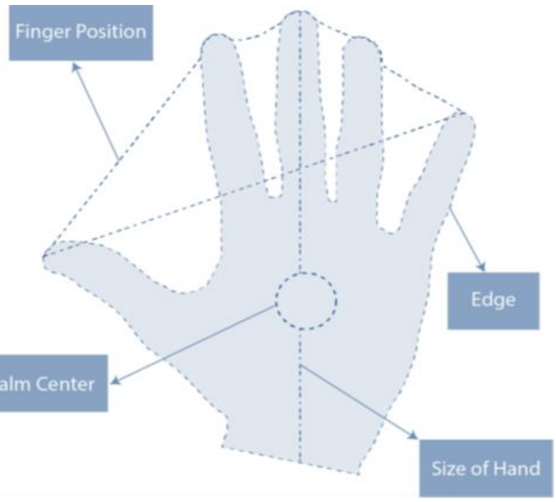


Fig. 3: Feature Points of the Hand

We choose the four features from the hand because each is important to provide information for the recognition system. Margin value is computed by the technique of canny edge detection which is used to select the human hand boundaries and the edges. Hand size is calculated using the method of the bounding box that can be computed by the maximum and minimum values of the coordinates of the x and y of the pixels of the hand [16].

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

The finger position and palm center are calculated by functions. The following steps represent the implementation of the algorithm SIFT:

1. Read the input image ID with its path.
2. Calculate d by the function of the difference of Gaussian.
3. Compute the initial key points.
4. Eliminating inaccurately and unstable localized key points.
5. Generate feature descriptor.
6. Output is a list of key points.

D. Convolutional Neural Network (CNN)

CNN is a deep-learning algorithm; it can train big datasets including millions of factors. Also, it is a feature extraction model that has been applied to image recognition and has given amazing results, and on data from the video. Furthermore, CNNs are in use by several leaders of the industry such as Amazon, Facebook, and Google [17][24][25][26][27]. CNN is mainly utilized to classify images and gather them depending on their similarity. It plays an important role in reducing processor requirements by acting as a multi-layer system. CNNs contain three layers; input, output, and hidden layer. Whereas, the hidden layer consists of a fully connected layer, a pooling layer, and finally convolutional layer [18]-[19]. The convolutional layer is the first layer of the CNN network and is considered the basic building block that performs most computational heavy lifting. A so-called filter or kernel is applied to it. It is the question of determining the presence of certain patterns or features in the original image. Also, many filters can be used when there is a need to extract different features. The filter must be small to scan the picture completely and apply the appropriate calculations. In the pooling layer, the size of the activation maps is reduced, and the number of necessary calculations is reduced preventing falling into a condition called overfit tinting. When reducing the size of large arrays, two functions are used: max: and here the maximum values are specified in each window. Average: Calculates the arithmetic average of the values in a single window. The first technique is the most popular [20].

There is a method called Max-pooling that scans the activation map to a small window and leaves the larger values within each window thus reducing the size of the map. The fully connected layer is the last in the convolutional network, and the reason for its presence at the end of the process is that the final classification process takes place in it. In this layer, the neurons are fully connected to all the ganglia of the previous layer.

Finally, we combine the features to create a form. We have a Soft-Max or sigmoid activate function to classify the outputs.

To implement the CNN algorithm, we used two requirements:

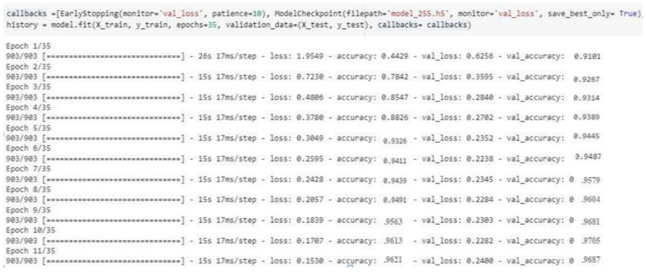
1. Python is a language of high-level programming [21].
2. Colab is a Google Research product for writing and executing Python code via browser [22].

E. CNN Algorithm Implementation Steps

1. Input is a list of key points (get from the SIFT algorithm).
2. Load the label of the class in English and Arabic.
3. Show the sign names in English and Arabic (unique values).
4. Descriptive statistics review the dataset's distribution shape, dispersion, and central tendency.
5. A plot bar is used for the distribution.
6. Data is split into three parts; training, testing, and prediction (70%, 20%, 10%) respectively.

F. Model Building and Evaluation

In model layers of CNNs, each layer contains neurons in positive quantity. The layer size of max-pooling is 2*2. The image length is 64*64. We reshape the image length to 64 x 64 x 1. For a primary layer, we use "relu" which is a function of activation. At the output layer, for classifying 32 classes; we use the activation function "soft-max". On the output layer; the quantity of the neurons is equivalent to the number of the classes. In training phenomena, we use the Loos Function (LF) for calculating the training losses and validation after each epoch. The model can prove its performance through lower loss and vice versa. The "categorical-cross-entropy" class calculates cross-entropy loss between predicted and actual values as the model predictions result. During the training operation, we used a call-back to keep the model weight after several periods. A function fit () fits the model instance to start the training operation. This function's benefit is model training in the specified periods using validation and training. Figure (4) shows the model after training it 35 epochs.



```
callbacks =[EarlyStopping(monitor='val_loss', patience=10), ModelCheckpoint(filepath='model_255.h5', monitor='val_loss', save_best_only= True)
history = model.fit(X_train, y_train, epochs=35, validation_data=(X_test, y_test), callbacks= callbacks)

Epoch 1/35
903/903 [==============================] - 26s 17ms/step - loss: 1.9549 - accuracy: 0.4429 - val_loss: 0.6256 - val_accuracy: 0.9101
Epoch 2/35
903/903 [==============================] - 15s 17ms/step - loss: 0.7230 - accuracy: 0.7842 - val_loss: 0.3595 - val_accuracy: 0.9267
Epoch 3/35
903/903 [==============================] - 15s 17ms/step - loss: 0.4806 - accuracy: 0.8547 - val_loss: 0.2840 - val_accuracy: 0.9314
Epoch 4/35
903/903 [==============================] - 15s 17ms/step - loss: 0.3780 - accuracy: 0.8826 - val_loss: 0.2702 - val_accuracy: 0.9389
Epoch 5/35
903/903 [==============================] - 15s 17ms/step - loss: 0.3049 - accuracy: 0.9326 - val_loss: 0.2352 - val_accuracy: 0.9445
Epoch 6/35
903/903 [==============================] - 15s 17ms/step - loss: 0.2595 - accuracy: 0.9411 - val_loss: 0.2238 - val_accuracy: 0.9487
Epoch 7/35
903/903 [==============================] - 15s 17ms/step - loss: 0.2428 - accuracy: 0.9439 - val_loss: 0.2345 - val_accuracy: 0 .9579
Epoch 8/35
903/903 [==============================] - 15s 17ms/step - loss: 0.2057 - accuracy: 0.9491 - val_loss: 0.2284 - val_accuracy: 0 .9604
Epoch 9/35
903/903 [==============================] - 15s 17ms/step - loss: 0.1839 - accuracy: .9563 - val_loss: 0.2303 - val_accuracy: 0 .9681
Epoch 10/35
903/903 [==============================] - 15s 17ms/step - loss: 0.1707 - accuracy: .9613 - val_loss: 0.2282 - val_accuracy: 0 .9705
Epoch 11/35
903/903 [==============================] - 15s 17ms/step - loss: 0.1530 - accuracy: .9621 - val_loss: 0.2400 - val_accuracy: 0 .9687
```

Fig. 4: Training Model

The plotted curve calculates the accuracy of each epoch and the loss which can be obtained from the history of the Keras model. A comparison between the class of predicted and actual gives the accuracy. The loss computes via the value of cross-entropy between the class of actual and predicted. Figure (5) shows a chart of the accuracy containing two lines the blue refers to training accuracy through each epoch. The second line is brown referring to verification accuracy. We note that both lines are near each other, this indicates that the model is well-trained.

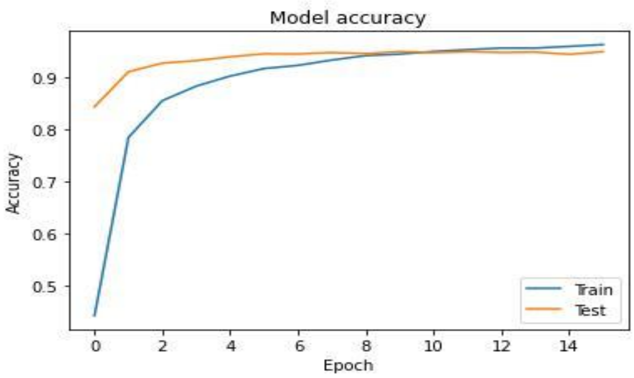


Fig. 5: Accuracy Chart

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

We note two lines in Figure (5) brown and blue, the brown refers to a set of the loss of validation while the blue indicates the loss of a training set via each epoch. So, on both datasets of train and validation, the model performance is similar. In both Figure (4) and Figure (5) the loss and accuracy splay at the 15th epoch indicates the model has learned the weights completely. So, the model terminates to get rid of overfitting at the 16th epoch because the model could recognize the input features for classifying unanticipated hand gestures better

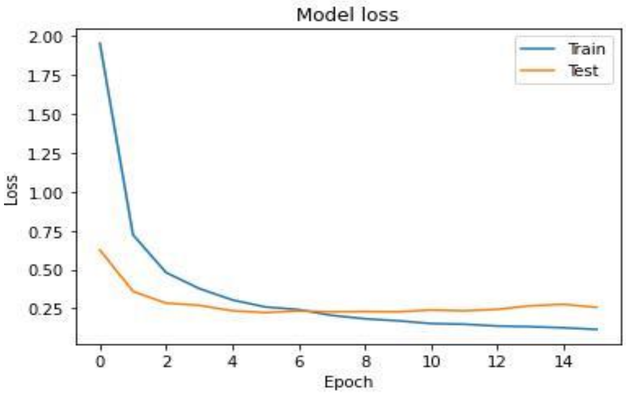


Fig. 6: Chart of Loss

Figure (7) shows the confusion matrix used to perform a statistical evaluation of the model to determine its true performance on unseen data that has not yet been encountered.

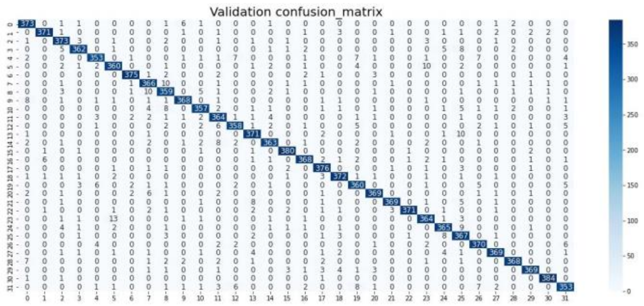


Fig. 7: Confusion Matrix

We processed the input images as a first step and then applied the SIFT algorithm and finally, to predict entered them into a model. Both Figure (8) and Figure (9) present the prophesy value in English and Arabic respectively.

Fig. 8: English Predicted Value

Fig. 9: Arabic Predicted Value

IV. RESULTS

In this section, we will show the results after applying the model by entering an image of the gesture. We notice that the resulting image is the predicted letter. When I make a certain gesture with my hand as in Figure (10) picture (a), the letter (bb) will be immediately assigned by the proposed model. Also in Figure (10) picture (b) presents the (dal) letter. Figure (11. a) presents the (al) letter. Finally, Figure (11. b) presents the (aleff) letter.

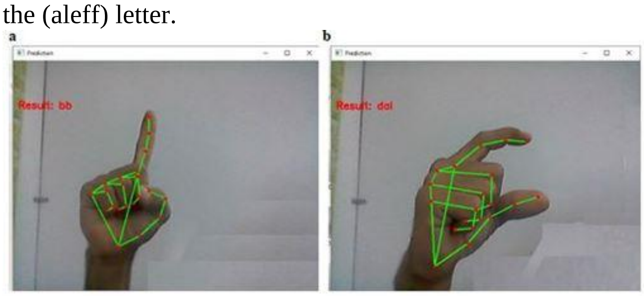


Fig. 10: (a) Show Result bb; (b) Show Result Dal

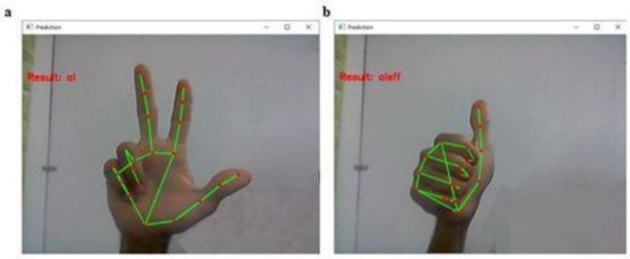


Fig. 11: (a) Show Result al; (b) Show Result Aleff

V. CONCLUSION

Sign language, a necessary tool is used to connect normal people with hearing hard people or deaf. Also, normal people suffer from understanding sign language. In this research, we introduce a system for ArSL recognition which includes four stages: dataset, pre-processing and segmentation, features extraction, and finally classification stage. This system is based on a camera for capturing gesture images. These images are taken in different environments and saved in a Portable Network Graphic(PNG).

Retrieval Number: 100.1/ijese.I258412100924
DOI: 10.35940/ijese.I2584.12100924
Journal Website: www.ijese.org

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

Images are transformed into grayscale to simplify them and apply the segmentation process. The segmentation is performed by utilizing thresholding. The SIFT is used in the third stage for feature extraction and selection which is considered a critical process because if the number of features increases the time of computational increases, and if the number of features decreases affects the system's accuracy. The images are entered into the SIFT algorithm, the output is a feature vector set. The fourth stage is a classification including the CNN use. The dataset was split into 70% and 30% for training and validation respectively. The accuracy was 97% which the proposed model achieves.

DECLARATION STATEMENT

After aggregating input from all authors, I must verify the accuracy of the following information as the article's author.

- **Conflicts of Interest/ Competing Interests:** Based on my understanding, this article has no conflicts of interest.
- **Funding Support:** This article has not been sponsored or funded by any organization or agency. The independence of this research is a crucial factor in affirming its impartiality, as it has been conducted without any external sway.
- **Ethical Approval and Consent to Participate:** The data provided in this article is exempt from the requirement for ethical approval or participant consent.
- **Data Access Statement and Material Availability:** The adequate resources of this article are publicly accessible.
- **Authors Contributions:** The authorship of this article is contributed equally to all participating individuals.

REFERENCES

1. S. Arooj, S. Altaf, S. Ahmad, H. Mahmoud, and A. S. N. Mohamed, "Enhancing sign language recognition using CNN and SIFT: A case study on Pakistan sign language," *Journal of King Saud University-Computer and Information Sciences*, vol. 36, no. 2, p. 101934, 2024. https://doi.org/10.1016/j.jksuci.2024.101934
2. R. A. Kadhim and M. Khamees, "A Real-Time American Sign Language Recognition System using Convolutional Neural Network for Real Datasets," *Tem Journal*, vol. 9, no. 3, 2020. https://doi.org/10.18421/TEM93-14
3. 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
4. N. S. Sagheer and S. A. Yousif, "A parallel clustering analysis based on Hadoop multi-node and Apache mahout," *Iraqi Journal of Science*, pp. 2431-2444, 2021. https://doi.org/10.24996/ijs.2021.62.7.32
5. C. Janiesch, P. Zschech, and K. Heinrich, "Machine learning and deep learning," *Electronic Markets*, vol. 31, no. 3, pp. 685-695, 2021. https://doi.org/10.1007/s12525-021-00475-2
6. Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, "A survey of convolutional neural networks: analysis, applications, and prospects," *IEEE Transactions on neural networks and learning systems*, vol. 33, no. 12, pp. 6999-7019, 2021. https://doi.org /10.1109/TNNLS.2021.3084827
7. K.-Z. Liu, P.-J. Lee, G.-C. Xu, and B.-H. Chang, "SIFT-enhanced CNN based objects recognition for satellite image," in *2020 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-Taiwan)*, 2020, pp. 1-2: IEEE. **DOI:** 10.1109/ICCE-Taiwan49838.2020.9258037
8. A. Alnahhas, B. Alkhatib, N. Al-Boukaee, N. Alhakim, O. Alzabibi, and N. Ajalyakeen, "Enhancing the recognition of Arabic sign language by using deep learning and leap motion controller," *Int. J. Sci. Technol. Res*, vol. 9, no. 4, pp. 1865-1870, 2020.
9. T. H. Noor *et al.*, "Real-Time Arabic Sign Language Recognition Using a Hybrid Deep Learning Model," *Sensors*, vol. 24, no. 11, p. 3683, 2024. https://drepo.sdl.edu.sa/handle/20.500.14154/2832
10. R. Marzouk, F. Alrowais, F. N. Al-Wesabi, and A. M. Hilal, "Atom search optimization with deep learning enabled Arabic sign language recognition for speaking and hearing disability persons," in *Healthcare*, 2022, vol. 10, no. 9, p. 1606: MDPI. https://doi.org/10.3390/healthcare10091606
11. F. H. A. Noor S. Sagheer, Manar Hamza Bashaa, "Enhancing Arabic Sign Language Recognition using Deep Learning," *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, vol. 13, no. 5, p. 6, 30l4l2024 2024. https://doi.org/10.35940/ijitee.E9844.13050424
12. R. Abdul Ameer, M. Ahmed, Z. Al-Qaysi, M. Salih, and M. L. Shuwandy, "Empowering Communication: A Deep Learning Framework for Arabic Sign Language Recognition with an Attention Mechanism," *Computers*, vol. 13, no. 6, p. 153, 2024. https://doi.org/10.3390/computers13060153
13. G. Latif, N. Mohammad, J. Alghazo, R. AlKhalaf, and R. AlKhalaf, "ArASL: Arabic alphabets sign language dataset," *Data in brief*, vol. 23, p. 103777, 2019. https://doi.org/10.1016/j.dib.2019.103777
14. M. Haroon, S. Altaf, S. Ahmad, M. Zaindin, S. Huda, and S. Iqbal, "Hand gesture recognition with a symmetric pattern under diverse illuminated conditions using artificial neural network," *Symmetry*, vol. 14, no. 10, p. 2045, 2022. https://doi.org/10.3390/sym14102045
15. M. S. Amin, S. T. H. Rizvi, and M. M. Hossain, "A comparative review on applications of different sensors for sign language recognition," *Journal of Imaging*, vol. 8, no. 4, p. 98, 2022. https://doi.org/10.3390/jimaging8040098
16. 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
17. R. Chauhan, K. K. Ghanshala, and R. Joshi, "Convolutional neural network (CNN) for image detection and recognition," in *2018 First International Conference on secure cyber computing and communication (ICSCCC)*, 2018, pp. 278-282: IEEE. **DOI:** 10.1109/ICSCCC.2018.8703316
18. 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
19. 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
20. R. Schmitz *et al.*, "Multi-scale fully convolutional neural networks for histopathology image segmentation: from nuclear aberrations to the global tissue architecture," *Medical image analysis*, vol. 70, p. 101996, 2021. https://doi.org/10.1016/j.media.2021.101996
21. J. Fontenrose, *Python: A Study of Delphic Myth and Its Origins*. Univ of California Press, 2023.
22. W. Vallejo, C. Díaz-Uribe, and C. Fajardo, "Google collab 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
23. Suharjito, Gunawan, H., & Thiracitta, N. (2020). Optimization of Modified Hidden Markov Model for Vision-Based Indonesian Sign Languages Recognition. In International Journal of Recent Technology and Engineering (IJRTE) (Vol. 8, Issue 5, pp. 615–621). https://doi.org/10.35940/ijrte.e5666.018520
24. Das, S., S, S., M, A., & Jayaram, S. (2021). Deep Learning Convolutional Neural Network for Defect Identification and Classification in Woven Fabric. In Indian Journal of Artificial Intelligence and Neural Networking (Vol. 1, Issue 2, pp. 9–13). https://doi.org/10.54105/ijainn.b1011.041221
25. Kumar, P., & Rawat, S. (2019). Implementing Convolutional Neural Networks for Simple Image Classification. In International Journal of Engineering and Advanced Technology (Vol. 9, Issue 2, pp. 3616–3619). https://doi.org/10.35940/ijeat.b3279.129219
26. Raj, H., Duggal., A., M., A. K. S., Uppara, S., & S., S. M. (2020). Hand Motion Analysis using CNN. In International Journal of Soft Computing and Engineering (Vol. 9, Issue 6, pp. 26–30). https://doi.org/10.35940/ijsce.f3409.059620
27. Magapu, H., Krishna Sai, M. R., & Goteti, B. (2024). Human Deep Neural Networks with Artificial Intelligence and Mathematical Formulas. In International Journal of Emerging Science and Engineering (Vol. 12, Issue 4, pp. 1–2). https://doi.org/10.35940/ijese.c9803.12040324

AUTHOR'S PROFILE

Manar Hamza 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.

Faezah Hamad completed a B. in computer science at Kerbala University and a Master's in Networks at Babylon University. She has had 14 years of Experience 6 years in teaching. Her research interests are Computer Networks, Artificial Intelligence, Machine Learning, and Deep Learning.

Noor Sabah completed a B. in computer science at Kerbala University and a Master's in Artificial Intelligence at Al-Nahrain University. She has had 14 years of Experience eleven years in teaching. Her research interests: are Artificial Intelligence, Machine Learning, Deep Learning, and Big Data.

Wasan Mueti Hadi is a Computer Science culture at Kerbala University. She has a Master's degree in AI from Turkey. She has a strong enthusiasm for Artificial Intelligence and has delved into various domains, including computer vision and image analysis.

Retrieval Number: 100.1/ijese.I258412100924
DOI: 10.35940/ijese.I2584.12100924
Journal Website: www.ijese.org

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