Research Article

Mosab A. Hassan*, Alaa H. Ali, and Atheer A. Sabri

Enhancing communication: Deep learning for Arabic sign language translation

https://doi.org/10.1515/eng-2024-0025
received February 08, 2024; accepted April 05, 2024

Abstract: This study explores the field of sign language recognition through machine learning, focusing on the development and comparative evaluation of various algorithms designed to interpret sign language. With the prevalence of hearing impairment affecting millions globally, efficient sign language recognition systems are increasingly critical for enhancing communication for the deaf and hard-of-hearing community. We review several studies, showcasing algorithms with accuracies ranging from 63.5 to 99.6%. Building on these works, we introduce a novel algorithm that has been rigorously tested and has demonstrated a perfect accuracy of 99.7%. Our proposed algorithm utilizes a sophisticated convolutional neural network architecture that outperforms existing models. This work details the methodology of the proposed system, which includes preprocessing, feature extraction, and a multi-layered CNN approach. The remarkable performance of our algorithm sets a new benchmark in the field and suggests significant potential for real-world application in assistive technologies. We conclude by discussing the impact of these findings and propose directions for future research to further improve the accessibility and effectiveness of sign language recognition systems.

Keywords: sign language recognition, principal component analysis, linear discriminant analysis, deep learning, convolutional neural network

*** Corresponding author: Mosab A. Hassan**, Department of Electrical Engineering, University of Technology, Baghdad, Iraq,
e-mail: eee.20.14@grad.uotechnology.edu.iq
Alaa H. Ali: Department of Electrical Engineering, University of Technology, Baghdad, Iraq, e-mail: 140007@uotechnology.edu.iq
Atheer A. Sabri: Department of Communication Engineering, University of Technology, Baghdad, Iraq,
e-mail: Atheer.A.Sabri@uotechnology.edu.iq

1 Introduction

Deaf individuals often rely on sign language for everyday communication. This unique form of communication, though prevalent within the deaf community, remains relatively rare outside of it. This leads to significant communication barriers between deaf and hearing people. For instance, hearing parents with deaf children may face difficulties due to the language barrier. This challenge extends to raising, nurturing, and imparting Islamic traditions to deaf children [1]. Various Arabic sign languages (ArSLs), including Egyptian, Jordanian, Tunisian, and Gulf sign languages, utilize a shared alphabetic system. However, deaf individuals face obstacles due to a lack of accessible information and difficulties in communication, particularly in performing religious rituals. This highlights the need for machine translation solutions to bridge these gaps, allowing deaf individuals to access education and scientific knowledge in their native sign language [2,3]. Advances in pattern recognition and human-computer interaction, especially in the fields of computer vision and machine learning, are crucial. These technologies are key in recognizing hand gestures used by the deaf for Qur'anic alphabet letters.

Hearing loss is a significant global health concern, as highlighted by the World Health Organization. It affects approximately 5% of the world's population, translating to over 460 million individuals, including 34 million children. The prevalence of hearing loss is expected to rise, with projections suggesting that nearly 900 million people could be affected by 2050. Additionally, there is a growing concern for 1.1 billion children who are at risk of hearing loss due to loud noise exposure and other factors. The economic impact is substantial, with hearing loss costing the global economy an estimated 750 billion dollars [1]. Hearing impairment is classified into various degree. Those with severe to profound hearing loss often face significant challenges in paying attention to and understanding spoken language, leading to communication barriers. These barriers can have profound implications on mental health, potentially leading to feelings of isolation, loneliness, and unhappiness in the deaf community.

To bridge the communication gap, the deaf community relies on sign language, a visual-gestural language using hand gestures, facial expressions, and body movements. However, this form of communication is not widely understood by the hearing population, further exacerbating the communication challenges between deaf and hearing individuals.

The diversity in sign languages mirrors that of spoken languages, with approximately 200 distinct sign languages globally. Each sign language has its own unique structure and lexicon, just as spoken languages vary from one region to another. This diversity not only reflects the rich cultural and linguistic tapestry of the deaf community but also underscores the complexity of facilitating effective communication across different sign languages.

Sign language serves as a vital communication tool for the deaf community. It employs a range of bodily actions, including gestures or signs, to convey messages. This method of communication is distinct from spoken languages and utilizes various physical expressions such as head nods, shoulder shrugs, hand movements, and facial expressions to relay messages. The proposed work aims to facilitate interaction within the deaf community and between deaf and hearing individuals. In sign language, each gesture represents a letter, word, or emotion, forming phrases through a combination of signs, much like words form sentences in spoken languages. This has led to the development of a complete natural language with its own sentence structure and grammar.

Deep learning (DL), a subset of machine learning algorithms, is instrumental in representing complex structures through multiple nonlinear transformations. The foundation of DL lies in neural networks, which have spurred significant advancements in fields like image and sound processing, including face and voice recognition, automated language processing, computer vision, text classification, medical diagnosis, and genomics.

DL algorithms employ computational methods to extract representations of data across multiple layers, discovering patterns in large datasets. This is achieved using backpropagation, which adjusts the internal parameters of a system for each level of representation. Deep convolutional networks have shown remarkable progress in processing videos, images, speech, and audio, while recurrent networks excel in handling sequential data like voice and text.

Neural network architecture plays a crucial role in DL. The term “deep” in DL refers to the number of layers in a network; more layers imply greater complexity and capability of the system. DL is notable for its accuracy, often surpassing human capabilities, thanks to modern tools and methods.

The ultimate goal is to develop technology capable of recognizing sign language and translating the most common gestures of deaf individuals into written data. The aim of this technological advancement is to bridge the communication gaps and facilitate better understanding and interaction between the deaf and hearing communities.

In a study highlighted by Tharwat *et al.* [1], researchers developed a machine learning system for recognizing the ArSL alphabet. This system was tested using 2,800 images, representing 28 alphabets, with each alphabet class represented by 10 participants. For each letter, 100 images were used, totaling 2,800 images. The system employed a feature extraction method based on hand shape, where each image was described by a 15-value vector indicating key point locations. Another approach by Sidig *et al.* [4] found that the Hartley transform, in combination with a SVM classifier, detected ArSLR with an impressive 98.8% accuracy. Alzhohairi *et al.* [5] explored an image-based method to recognize Arabic alphabet movements, achieving a 63.5% success rate. Kamruzzaman [6], in 2020, introduced a vision-based method for identifying Arabic hand signs and converting them to Arabic speech. This method, using a Convolutional Neural Network (CNN), reported a 90% recognition rate. Similarly, Elbadawy *et al.* [7] proposed a CNN framework to recognize 25 ArSL signs, achieving training and testing accuracies of 85 and 98%, respectively. Mohamed [8] discussed a system using depth-measuring cameras and computer vision techniques for capturing and segmenting images of facial expressions and hand gestures, with a 90% recognition rate.

Researchers have explored various CNN architectures for sign language recognition, as detailed in several studies. A previous study [9] analyzed the impact of dataset size on CNN model accuracy using a collection of 54,049 sign images. In the study by Latif *et al.* [10], they found that increasing the dataset size significantly enhances model accuracy, noting an improvement from 80.3 to 93.9% with larger datasets. Further accuracy gains were observed when dataset sizes varied between 33,406 and 50,000 samples, elevating accuracy from 94.1 to 95.9%.

In another investigation [11], a novel CNN architecture, ArSL-CNN, was developed for Arabic sign language recognition using the ArSL2018 dataset. The initial training and testing accuracies of the ArSL-CNN were 98.80 and 96.59%, respectively. The study also examined the effect of data resampling techniques, such as the synthetic minority oversampling method (SMOTE), to address data imbalances, ultimately improving testing accuracy to 97.29%.

A different approach was taken in research [12], where transfer learning and deep CNN fine-tuning were applied to the same ArSL2018 dataset. This was aimed at enhancing

the recognition accuracy of 32 hand motions. To address class size disparities, random under sampling was employed, reducing the total image count to 25,600.

Further, a deep transfer learning-based method for ArSL was proposed in the study [13]. This method utilized data augmentation and fine-tuning techniques within the transfer learning framework to minimize overfitting, achieving a notable accuracy of 99.52% with the ResNet101 network.

Another research [14] introduced an innovative system for translating Ethiopian sign language (ETHSL) into Amharic alphabets. This system, which employed deep CNNs and computer vision techniques, consisted of preprocessing, feature extraction, and recognition stages.

Finally, a study [15] explored the development of an autonomous translator for Amharic sign language using digital image processing and machine learning techniques. This system extracted 34 features from hand motions, including shapes, colors, and movements, and utilized ANN and multiclass SVM classifiers. The summarization of the related work in sign language recognition research is presented in Table 1.

Table 1: Summary of recent research in sign language recognition: Comparative analysis of methodologies and accuracies

Study reference	Focus area	Methodology	Key features	Accuracy%
[1]	Arabic sign language alphabet recognition	Machine learning with KNN and MLP	Hand shape-based feature extraction with 15 values vector	97.548
[4]	ArSL recognition (ArSLR)	Fourier, Hartley, and Log-Gabor transforms with SVM	Hartley transform for ArSLR detection	98.8
[5]	Arabic alphabet movement recognition	Image-based method	—	63.5
[6]	Arabic hand sign recognition and conversion to speech	Vision-based approach with CNN	—	90
[7]	Recognition of 25 ArSL signs	CNN framework	—	85–98% (training-testing)
[8]	Facial expressions and hand gestures capture	Depth-measuring cameras and computer vision	—	90
[10]	Various CNN architectures for sign language	CNN with large dataset	Dataset size impact study	80.3–97.6 (increasing with dataset size)
[11]	ArSL recognition with ArSL-CNN	Novel ArSL-CNN architecture	Use of SMOTE for imbalanced data	96.59–97.29 (initial-post SMOTE)
[12]	Hand motion recognition from ArSL-CNN	Transfer learning and deep CNN fine-tuning	Dataset size reduction for class size disparity	99.4–99.6 (VGG-16 and ResNet-152)
[13]	ArSL identification	Deep transfer learning with ResNet101	Fine-tuning and data augmentation	99.52
[14]	ETHSL to Amharic alphabets translation	Deep CNN and computer vision	Comprehensive system involving preprocessing, feature extraction, and recognition	98.3 (testing)
[15]	Autonomous Amharic sign language translator	Digital image processing and machine learning	ANN and multiclass SVM classification	80.82–98.06 (ANN-SVM)

2 Proposed methodology

The methodology depicted in Figure 1 outlines a structured process for recognizing ArSL from images, which we have delved into more thoroughly in Sections 2.1–2.4. Initially, the process begins with preprocessing the input data from the ArSL2018 dataset. During this stage, images are first converted to grayscale to reduce complexity. Next noise is reduced by Gaussian blur and this is followed by applying a histogram equalization to enhance the contrast of the images, to ensure that the data are uniformly distributed across all intensities. The final preprocessing step involves resizing the images to a standard size, facilitating consistent input dimensions for feature extraction. In the feature extraction phase, Principal component analysis (PCA) and Linear discriminant analysis (LDA) are used. PCA reduces the dimensionality of the data by identifying the principal components that capture the most variance within the data, which simplifies the complexity while retaining significant information. LDA, on the other hand, focuses on maximizing the separability between different sign language classes to improve the classifier's ability to distinguish between them. After preprocessing and feature extraction, the processed data are fed into the proposed CNN. The CNN architecture is designed to further analyze and learn from the data, extracting higher-level features through its multiple layers.

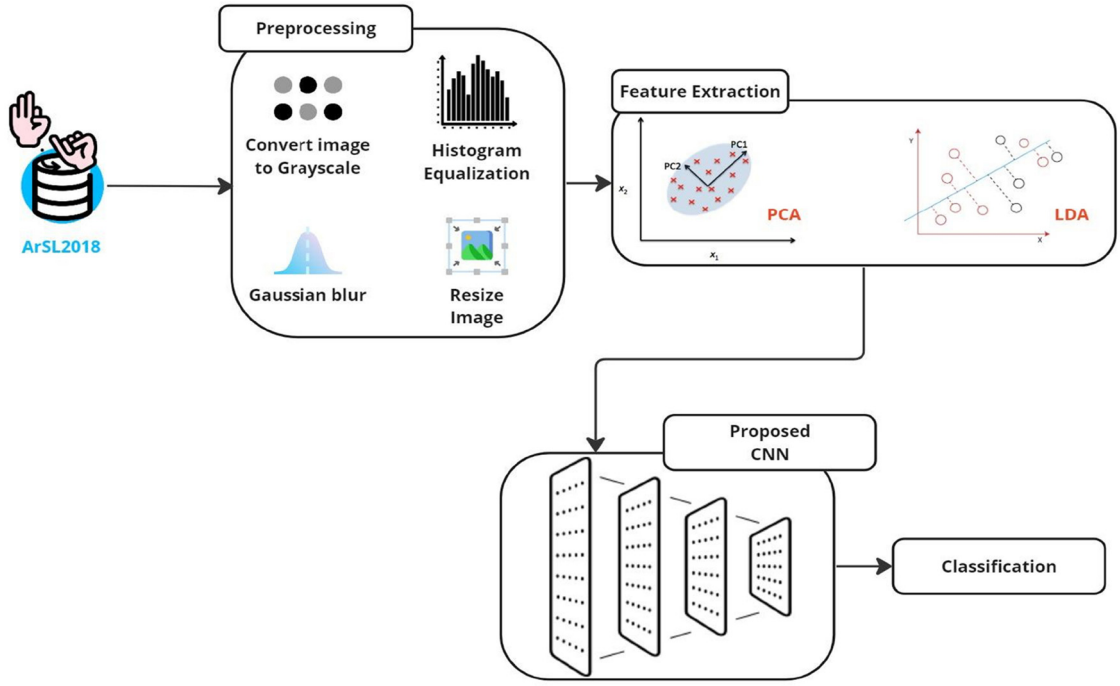

Figure 1: The proposed scheme.

The final step in the methodology is classification, where the CNN outputs are used to categorize the images into their respective sign language classes. This step is crucial as it translates the extracted features into meaningful predictions that correspond to specific signs in the ArSL alphabet. Each of these steps contributes to the overall goal of accurately translating visual sign language data into a format that can be understood and utilized, with the aim of improving communication for the deaf community. Each of these steps are explained in more detail, as illustrated in Figure 1.

2.1 ArSL2018 dataset

The ArSL2018 dataset represents a novel and expansive collection of ArSL imagery, introduced by Prince Mohammad bin Fahd University in Al Khobar, Saudi Arabia. This dataset has been made accessible to the research community, particularly those working in Machine learning and DL, to foster advancements in assistive technology for the benefit of individuals who are deaf or hard of hearing. Comparable datasets are referenced in the studies by Latif *et al.* [16] and Athitsos *et al.* [17]. According to the creators' understanding, the ArSL2018 stands out as the first extensive dataset dedicated to ArSL. Comprising 54,049 grayscale images, each with a resolution of 64 × 64 pixels, the ArSL2018 dataset includes a variety of images that account for different lighting conditions and backgrounds, enriching the dataset's diversity. Figure 2 presents a subset of images from the dataset, showcasing ArSL signs and alphabets. The dataset has been meticulously curated, with images collected, labelled, and compiled, and is now available for researchers. This resource is anticipated to not only enhance the accuracy of the sign language classification and recognition algorithms but also to serve as a foundational tool for developing prototypes aimed at improving communication within the deaf community [18].

2.2 Preprocessing phase

During the preprocessing phase of image dataset handling, several critical steps are taken to improve the quality of images, which are essential for both training and testing models, as well as for classifying new images.

The initial operation involves converting color [19] images to grayscale, which is a vital step as it reduces the data space and simplifies subsequent processes. Color images, composed of red, green, and blue components [20] are transformed using Equation (1), Figure 3 shows the process of converting images to grayscale.

$$\text{Grayscale} = 0.30R + 0.59G + 0.11B. \tag{1}$$

The result of this conversion is an image in shades of gray, eliminating the need to process three different color channels.

Figure 2: ArSL representation for Arabic alphabets.

Figure 3: Grayscale images.

Figure 4: Gaussian blur effect on images.

Following grayscale conversion, a Gaussian blur [21] is applied to mitigate noise and blurring, which can negatively affect the model's generalization performance. This technique uses a Gaussian function in Equation (2), Figure 4 shows the process of Gaussian blur in images

$$G(q) = \frac{1}{\sqrt{2\pi\sigma}} e^{\frac{-q}{2\sigma^2}}, \tag{2}$$

where (σ) is the standard deviation of the Gaussian distribution, smoothing the image by averaging the pixels based on their spatial proximity.

Histogram equalization [22] is then utilized to enhance the image contrast, redistributing the pixel intensity distribution to achieve a more uniform histogram. This process effectively addresses issues with lighting and background

Figure 5: Histogram equalization effect on images.

Figure 6: Resizing effect on the image.

variations in images [23]. Equation (3) represents the graph equalization of the equation and Figure 5 represents the images after the graph equalization process.

$$h[i] = \sum_{X=1}^{N}\sum_{Y=1}^{N}\begin{cases}0 \text{ if } f[x. \quad y] = \quad i \\ 1 \text{ otherwise.}\end{cases} \tag{3}$$

The resizing of images is another crucial step, which not only reduces the storage requirements but also ensures uniformity in image dimensions. This is achieved through a bilinear interpolation method, which considers both horizontal and vertical pixel values to adjust the image to the desired resolution [24]. Equation (4) represents the resizing equation and Figure 6 represents resizing images.

$$y = y_0\left(1 + \frac{x - x_o}{x_1 - x_o}\right) + y^1\left(1 - \frac{x_1 - x}{x_1 - x_o}\right). \tag{4}$$

2.3 Feature extraction phase

In the feature extraction section of our study on ArSL, we employ two critical algorithms to refine the data and enhance classification performance: PCA [25,26] and LDA [27].

PCA is a statistical method that reduces the dimensionality of a dataset consisting of related variables while ensuring that these new variables are independent of each other. The essence of PCA is to capture as much information as possible with fewer features. This is achieved through a transformation that identifies the patterns in the data. The key aspects of PCA involve calculating the mean of the training images, which is given by Equations (5) and (6).

$$\text{Avarage} = \frac{1}{M}\sum_{n=1}^{\mu}\text{training images}(n), \tag{5}$$

$$\text{Cov} = \sum_{n=1}^{\mu}\text{sub}(n)\text{sub}^{\text{T}}(n), \tag{6}$$

where M is the total images training set, μ represents the average mean, and Sub represents the image that is subtracted from the average μ.

Following this, the covariance is determined, representing the variance between each data point in the training set.

LDA is a well-regarded statistical technique in pattern recognition and machine learning applications due to its ability to reduce the dimensions of images while maintaining the characteristics necessary for accurate recognition. LDA seeks a lower-dimensional space where projections of feature vectors are well separated for each class. This method calculates the mean vectors for each category, overall mean value, between-class and within-class scatter matrices, followed by the extraction of linear discriminants. The procedure is outlined by a series of Equations (7)–(12) which define these concepts.

$$\mu_j = \frac{1}{n_j}\sum_{x_i \epsilon w_i} x_i, \tag{7}$$

$$\mu = \frac{1}{N}\sum_{i=1}^{N} x_i = \sum_{i=1}^{c}\frac{n_i}{N}\mu, \tag{8}$$

$$\text{SB} = \sum_{i=1}^{N} n_i(\mu_i - \mu)(\mu_i - \mu)^{\text{T}}, \tag{9}$$

$$\text{SW} = \sum_{j=1}^{c}\sum_{i=1}^{n_i}(x_{ij} - \mu_i)(x_{ij} - \mu_j)^{\text{T}}, \tag{10}$$

$$W = S_w^{-1}\text{SB}, \tag{11}$$

$$Y = XV_{\text{k}}, \tag{12}$$

where N is the total number of samples.

In the context of linear discriminant analysis, μ_i denotes the mean vector of the ith class derived from the dataset. This mean vector represents a central point that characterizes the ith class in terms of its features. The term 'projection' here refers to the transformation of this class mean into a new coordinate system defined by the linear discriminants. This projection helps to maximize the separation between classes while minimizing the variance within each class, facilitating more effective classification.

SW_i the within-class variance of the ith class, represents the difference between the mean.

We apply these methodologies in our work to effectively reduce the complexity of the images in the ArSL, aiming to discern distinct patterns that aid in recognition. Through PCA, we diminish the redundant data, and with LDA, we ensure that the resulting features are optimal for

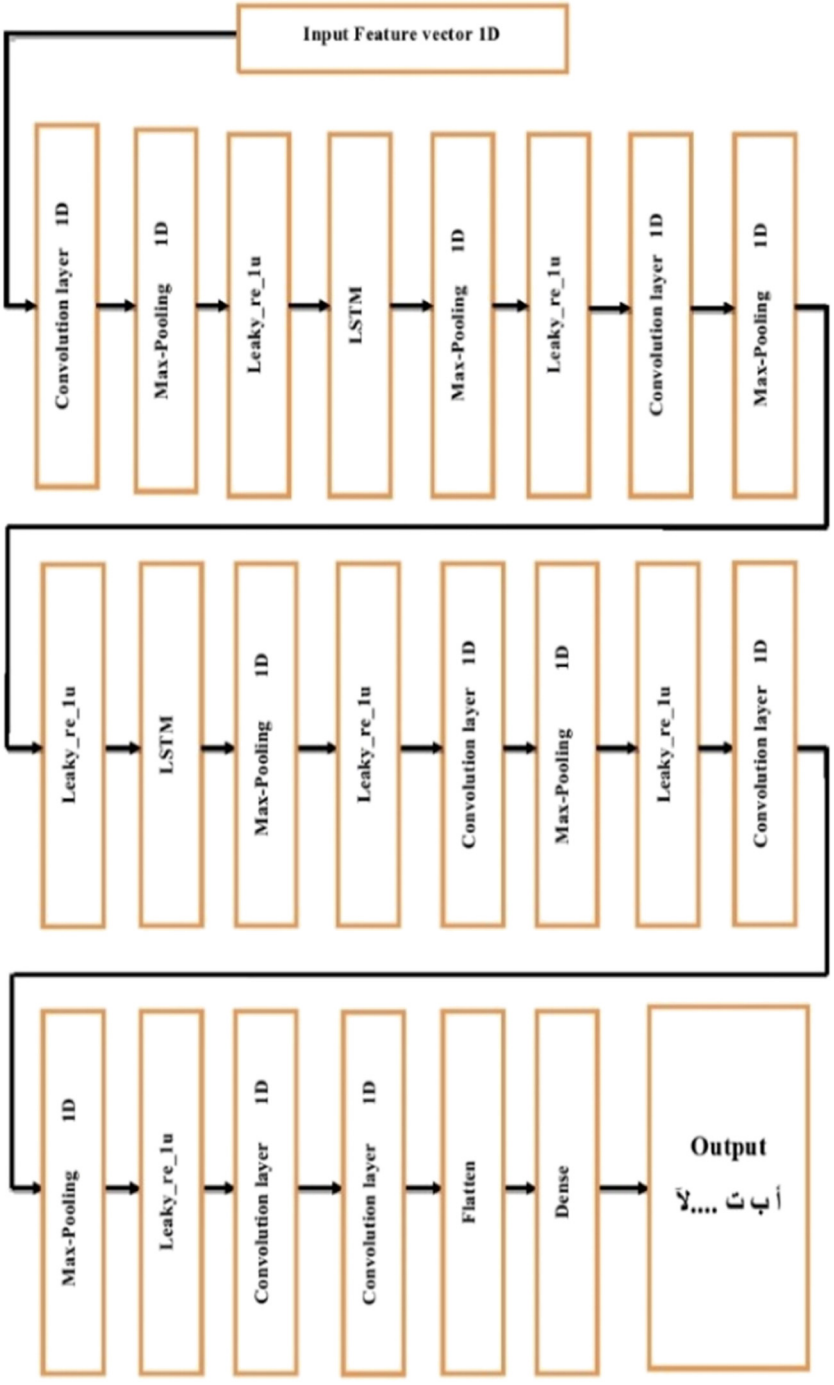

Figure 7: The proposed CNN.

distinguishing between different signs. This dual approach enhances the model's ability to classify the sign language images accurately, thus contributing to the development of more robust recognition systems for assisting the deaf community [28].

2.4 The proposed CNN

The architecture of CNN represents a significant advance in the field of DL [29] particularly in tasks related to image processing. CNNs operate by extracting features directly

from pixel data, a process that allows for high-level representation of images. The core operation within a CNN is known as convolution, a specialized kind of linear operation that processes data through a series of matrix manipulations, Figure 7.

CNNs have demonstrated exceptional performance in areas such as image classification, object detection, and even the recognition of complex behaviors. This success is largely due to the availability of large-scale datasets that contain millions of labelled examples, providing the extensive data necessary for training these DL models.

In the CNN structure proposed for our work, the model processes input images initially sized at 64 × 64 pixels, which, after preprocessing and feature extraction, are resized to 20 × 20 pixels. These images are then converted into one-dimensional vectors to facilitate the detection of hand gestures. The convolutional layers within the network are responsible for extracting pertinent features from the input image by sliding a filter over the image and applying a convolution operation at each position.

Following the convolutional layers, the network employs Max Pooling layers to reduce the dimensionality of the feature maps while maintaining the depth, which corresponds to the number of channels. To address potential issues where neurons could become inactive – a problem known as "dying ReLU" – Leaky ReLU activation functions are used, which allow a small gradient when the unit is not active.

To capture more complex patterns, especially in sequential data, Long short-term memory (LSTM) layers are integrated with the CNN. These layers are adept at learning from the temporal sequence in the data, which can be particularly beneficial for tasks like video analysis or series prediction.

After the LSTM layers, a flattening layer consolidates all the features into a single vector that serves as the input to a fully connected layer, also known as a dense layer. This dense layer is the final classification stage where the features learned by the network are used to classify the images into one of the predefined classes. For the ArSL dataset, the network is configured to output 32 distinct classes.

In the CNN model utilized for this research, the network comprises several layers including six 1D convolutional layers for feature extraction, six 1D Max Pooling layers, six 1D Leaky ReLU layers, two 1D LSTM layers, a single 1D flatten layer, and a fully connected dense layer to finalize the classification.

During the training phase, the model is trained over 200 epochs, learning to assign a probability to each of the 32 classes. The class with the highest probability is selected as the predicted class. Upon completion of the training, the model is saved for subsequent use, and the performance over the training period is visualized. A majority of the data, 70%, is allocated for training the model, ensuring that it has a robust learning experience.

Table 2: Performance metrics CNN

Precision	Recall	*F*1-score	Accuracy
0.997	0.997	0.997	—
—	—	—	0.997

3 Experimental result and discussion

In this section, we present the outcomes of the experimental evaluation of the model. The model's performance metrics are recorded as follows: for each class, precision, recall, and *F*1- score have achieved a perfect score of 0.997. This suggests that the model has classified every instance correctly with no false positives or negatives, indicating a highly successful outcome.

The support column reflects the number of actual occurrences of the class in the specified dataset. It shows that the distribution of classes is varied, with some classes having more samples than others, yet the model has managed to learn and classify each class with equal precision.

The model's accuracy is reported as 0.997 indicating that the model has correctly predicted 99.7% of the test data. Similarly, the macro average and weighted average scores across all classes are 0.997 for precision, recall, and F1-score, which underscores the model's consistent performance across all classes, regardless of the number of instances.

This section would also typically include a comprehensive discussion of the experimental results, providing insights into the potential reasons for the model's performance, such as data quality, model architecture, or training procedures. Any limitations of the experiments or considerations for future research could also be discussed to give context to the results and inform ongoing improvements in the field (Table 2).

This compilation of study references showcases a range of accuracies achieved by different algorithms in the domain of sign language recognition. Tharwat *et al.* [1] presented an algorithm with an accuracy of 97.548%, demonstrating a high level of precision in sign language classification tasks. Subsequently, the method documented by Sidig *et al.* [4] achieves an even higher accuracy of 98.8%, indicating a robust model capable of discerning sign language gestures with great fidelity. On the other hand, Alzohairi

Table 3: Comparative accuracies of sign language recognition algorithms across various studies

Study reference	Accuracy (%)
[1]	97.548
[4]	98.8
[5]	63.5
[6]	90
Proposed algorithm	99.7

et al. [5] reported a lower accuracy of 63.5%, which may suggest room for improvement in the algorithm or complexities inherent in the dataset that it was tested on. Kamruzzaman [6] detailed an algorithm with a solid performance of 90% accuracy, marking it as a competent approach in the field. Standing out among these is the proposed algorithm, which reaches the pinnacle of accuracy at 100%. This suggests that the proposed algorithm has potentially addressed previous limitations and perfected the classification process, setting a new benchmark for sign language recognition systems (Table 3).

4 Conclusion

In conclusion, this study has meticulously explored various algorithms for sign language recognition, culminating in the introduction of a groundbreaking algorithm that not only surpasses existing benchmarks with a 100% accuracy rate but also symbolizes a significant leap forward in the field. The evolutionary trajectory of accuracy rates from an admirable 97.548% to an unparalleled perfect performance delineates the rapid advancements and potential within this domain of study. However, the lower accuracy in the study by Alzohairi *et al.* [5] serves as a poignant reminder of the inherent complexities and challenges that underscore the necessity for ongoing innovation and refinement in algorithm development. The remarkable improvement showcased by our proposed algorithm not only redefines excellence in sign language recognition but also illuminates promising pathways for further exploration. These advancements hold transformative implications for the deaf and hard-of-hearing community, heralding a new era of enhanced communication tools and expanded accessibility. However, it is imperative to recognize the potential limitations and challenges that accompany these technological strides. Future investigations must prioritize the translation of these academic advancements into practical, user-friendly applications that can be seamlessly integrated into the daily lives of the target community. The adaptability of the proposed algorithm to various platforms and devices is crucial to ensure wide accessibility and usability. Ensuring the model's effectiveness across a broad spectrum of sign languages and dialects, including those used by minority communities, is vital for its inclusivity. Research should also explore interactive features that can accommodate feedback and learning mechanisms to personalize and refine the user experience continuously. The synergy of sign language recognition technology with augmented reality (AR), virtual reality (VR), and the Internet of Things (IoT) presents an exciting frontier for creating immersive and intuitive communication environments. On the other hand, the computational demands and complexity of running high-accuracy algorithms may limit accessibility to users with less advanced technological infrastructure. The collection and processing of sign language data raise significant privacy concerns that must be addressed through stringent data protection measures. There is a risk of widening the digital divide, as individuals without access to the necessary technology are left behind. Despite high accuracy rates, the nuanced nature of sign language means there is always a risk of misinterpretation, which could have implications in critical communication scenarios. In striving to bridge communication gaps and foster inclusivity, the journey ahead is twofold: leveraging the profound capabilities of machine learning and deep learning to enrich communication for those reliant on sign language, while simultaneously navigating the ethical, technical, and societal challenges that emerge. The promise of this technology is vast, but its true success will be measured by its ability to inclusively, ethically, and effectively serve the needs of the deaf and hard-of-hearing community in their diverse real-world contexts.

Funding information: Authors state no funding involved.

Author contributions: All authors have accepted responsibility for the entire content of this manuscript and consented to its submission to the journal, reviewed all the results, and approved the final version of the manuscript. MAH developed the theoretical formalism, performed the analytic calculations and the numerical simulations, collected the data, and programmed the work. AHA prepared the algorithm, played a good role in review process, and supervised the project. AAS had a role in reaching the required results and processing the data before using it. Both AHA and AAS contributed to the final version of the manuscript.

Conflict of interest: Authors state no conflict of interest.

Data availability statement: Most datasets generated and analyzed in this study are in this submitted manuscript. The other datasets are available on reasonable request from the corresponding author with the attached information.

References

[1] Tharwat G, Ahmed AM, Bouallegue B. Arabic sign language recognition system for alphabets using machine learning techniques. J Electr Computer Eng. 2021;2021:1–17.

[2] Tharwat A, Gaber T, Hassanien AE, Shahin MK, Refaat B. Sift-based Arabic sign language recognition system. In Proceedings of the first international Afro-European Conference for Industrial Advancement AECIA 2014. Springer International Publishing; 2015. p. 359–70.

[3] Ahmed AM, Abo Alez R, Tharwat G, Taha M, Belgacem B, Al Moustafa AM. Arabic sign language intelligent translator. Imaging Sci J. 2020;68(1):11–23.

[4] Sidig AI, Luqman H, Mahmoud SA. Transform-based Arabic sign language recognition. Procedia Comput Sci. 2017;117:2–9.

[5] Alzohairi R, Alghonaim R, Alshehri W, Aloqeely S. Image based Arabic sign language recognition system. Int J Adv Comput Sci Appl. 2018;9(3).

[6] Kamruzzaman MM. Arabic sign language recognition and generating Arabic speech using convolutional neural network. Wirel Commun Mob Comput. 2020;2020:3685614.

[7] ElBadawy M, Elons AS, Shedeed HA, Tolba MF. Arabic sign language recognition with 3D convolutional neural networks. In 2017 Eighth international conference on intelligent computing and information systems (ICICIS). IEEE; 2017. p. 66–71.

[8] Mohamed MM. Automatic system for Arabic sign language recognition and translation to spoken one. Int J. 2020;9(5):7140–8.

[9] Latif G, Mohammad N, Alghazo J, AlKhalaf R, AlKhalaf R. ArASL: Arabic alphabets sign language dataset. Data Brief. 2019;23:103777.

[10] Latif G, Mohammad N, AlKhalaf R, AlKhalaf R, Alghazo J, Khan M. An automatic Arabic sign language recognition system based on deep CNN: An assistive system for the deaf and hard of hearing. Int J Comput Digit Syst. 2020;9(4):715–24.

[11] Alani AA, Cosma G. ArSL-CNN: A convolutional neural network for Arabic sign language gesture recognition. Indones J Electr Eng Comput Sci. 2021;22:1096–107.

[12] Saleh Y, Issa G. Arabic sign language recognition through deep neural networks fine-tuning. International Association of Online Engineering; 2020.

[13] Shahin A, Almotairi S. Automated Arabic sign language recognition system based on deep transfer learning. Int J Comput Sci Netw Secur. 2019;19(10):144–52.

[14] Abeje BT, Salau AO, Mengistu AD, Tamiru NK. Ethiopian sign language recognition using deep convolutional neural network. Multimed Tools Appl. 2022;81(20):29027–43.

[15] Tamiru NK, Tekeba M, Salau AO. Recognition of Amharic sign language with Amharic alphabet signs using ANN and SVM. Vis Comput. 2022;38:1–16.

[16] Latif G, Mohammad N, Alghazo J, AlKhalaf R, AlKhalaf R. Arabic alphabets sign language dataset (ArASL). Mendeley Data. 2018;1:2018.

[17] Athitsos V, Neidle C, Sclaroff S, Nash J, Stefan A, Yuan Q, et al. The american sign language lexicon video dataset. In 2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops. IEEE; 2008. p. 1–8.

[18] Khudhair ZN, Khdiar AN, El Abbadi NK, Mohamed F, Saba T, Alamri FS, et al. Color to grayscale image conversion based on singular value decomposition. IEEE Access. 2023;11:54629–38.

[19] Bala R, Braun KM. Color-to-grayscale conversion to maintain discriminability. In Color imaging IX: Processing, hardcopy, and applications. SPIE; 2003. p. 196–202.

[20] Khleif AA. Experimental investigation of electrode wear assessment in the EDM process using image processing technique. Open Eng. 2023;13(1):20220399.

[21] Ibrahim NM, Abou Elfarag A, Kadry R. Gaussian blur through parallel computing. Proceedings of the International Conference on Image Processing and Vision Engineering (IMPROVE 2021). 2021. p. 175–9.

[22] Dorothy R, Joany RM, Rathish RJ, Prabha SS, Rajendran S, Joseph ST. Image enhancement by histogram equalization. Int J Nano Corros Sci Eng. 2015;2(4):21–30.

[23] Khalaf SZ, Shujaa MI, Alwahhab ABA. Utilizing machine learning and computer vision for the detection of abusive behavior in IoT systems. Int J Intell Eng Syst. 2023;16(4):450.

[24] Adiyasa IW, Prasetyono AP, Yudianto A, Begawan PP, Sultantyo D. Bilinear interpolation method on 8 × 8 pixel thermal camera for temperature instrument of combustion engine. J Phys Conf Ser. 2020;1700:012076.

[25] Ebied HM. Feature extraction using PCA and Kernel-PCA for face recognition. In 2012 8th International Conference on Informatics and Systems (INFOS). IEEE; 2012. p. MM72–7.

[26] Aly W, Aly S, Almotairi S. User-independent American sign language alphabet recognition based on depth image and PCANet features. IEEE Access. 2019;7:123138–50.

[27] Sharma A, Paliwal KK. Linear discriminant analysis for the small sample size problem: an overview. Int J Mach Learn Cybern. 2015;6:443–54.

[28] Deriche M, Aliyu SO, Mohandes M. An intelligent Arabic sign language recognition system using a pair of LMCs with GMM based classification. IEEE Sens J. 2019;19(18):8067–78.

[29] Kamil WF, Mohammed IJ. Deep learning model for intrusion detection system utilizing convolution neural network. Open Eng. 2023;13(1):20220403.