 Iraqi Journal of Science, 2023, Vol. 64, No. 10, pp: 5265-5279
DOI: 10.24996/ijs.2023.64.10.33

ISSN: 0067-2904

Alphabets Arabic Sign Language Recognition Based on A Hybrid Model Combining Linear Discrimination Analysis and A One-Dimensional Convolutional Neural Network

Maha S. Altememe [1]*, Nidhal K. El Abbadi [2]
[1]*College of Information Technology, University of Babylon, Iraq*
[2]*Computer Science Department, Faculty of Education, University of Kufa, Najaf, Iraq*

Received: 5/3/2022 Accepted: 12/11/2022 Published: 30/10/2023

Abstract
Due to the increasing number of people suffering from hearing impairment in Arab countries, automatic sign language translation has become a pressing necessity to reduce the gap between hearing impairment people and the community, hence minimizing their isolation. In this paper, we provide a new proposal for an Arabic Sign Language (ArSL) detection and recognition system capable of localizing and recognizing ArSL alphabets via a merge of features extracted using a Linear Discriminant Analysis (LDA) algorithm and a one-dimension Convolutional Neural Network (CNN), which is a new method to our knowledge. The important parameters used in this proposal are measured to select the best. The accuracy of this proposal was about 99.98%. Also, this model worked well with some of the challenges related to the detection of sign languages, such as variation of image illumination and background. Finally, comparing the results with other works prove the robustness of this proposal

Keywords: deaf people, image processing, hearing impairment, ArSL2018, feature extraction.

التعرف على لغة الإشارة العربية الأبجدية بناءً على نموذج هجين يجمع بين تحليل التمييز الخطي والشبكة العصبية التلافيفية أحادية البعد

مها صبري التميمي [1]*, نضال خضير العبادي [2]
[1] قسم البرمجيات، كلية تكنولوجيا المعلومات، جامعة بابل، بابل، العراق
[2] قسم علوم الحاسوب، كلية التربية ، جامعة الكوفة، النجف, العراق

الخلاصة
نظرًا لزيادة عدد الأشخاص الذين يعانون من ضعف السمع في الدول العربية، أصبحت الترجمة الآلية للغة الإشارة ضرورة ملحة لتقليص الفجوة بين الأشخاص ضعاف السمع والمجتمع ومن ثم تقليل عزلتهم. في هذا البحث، نقدم اقتراحًا جديدًا لنظام الكشف والتعرف على ArSL القادر على توطين الحروف الهجائية للغة الإشارة العربية والتعرف عليها باستخدام مجموعة من الميزات المستخرجة باستخدام خوارزمية تحليل التمييز الخطي والشبكة العصبية التلافيفية أحادية البعد وهي طريقة جديدة. على حد علمنا. يتم قياس المعلمات الهامة المستخدمة

*Email: maha.sabri@uokerbala.edu.iq

في هذا الاقتراح لاختيار الأفضل. كانت دقة هذا الاقتراح حوالي 99.98٪. أيضًا، يعمل هذا النموذج جيدًا مع بعض التحديات المتعلقة باكتشاف لغات الإشارة مثل تباين إضاءة الصورة والخلفية. تثبت مقارنة النتائج مع الأعمال الأخرى متانة هذا الاقتراح.

1. Introduction

Sign language is a way for hearing impaired people to communicate with others. Instead of communicating by sound, as most people do, they use hand gestures and facial emotions to convey a message to others. There is no one universal sign language, and regional variations within the same nation as well as between nations are significant. Each sign language possesses its own rules. Arab Sign Language (ArSL) is the sign employed by the Arabic community [1]. Most normal people cannot comprehend sign language, which enforces limitations on communication between normal and impaired people. Around 466 million (one in every twenty) people worldwide suffer from hearing impairment. This number may double in 30 years due to different causes [2].

Unfortunately, there are no standard signs for ArSL. This is a big challenge for the Arabic community to learn or interpret the ArSL. All the previous problems lead impaired people to lack learning opportunities, be isolated from society and harm their feelings. To reduce the gap between normal and impaired people in the Arabic community, there is an urgent need to use machines and computer vision as a translator between them [3]. Currently, hand gestures are crucial for the communication of Hearing and Speech Impaired (HSI) people. There are many features related to sign language.

Sign language has five main components: shape, position, gesture, orientation, and non-hand gestures. One type of sign language is finger language based on orientation and shape. Researchers suggested various methods to recognize ArSL. One is deep learning, which has become increasingly viable for diverse applications due to the large volumes of data availability (unlabelled and labelled), high-performance computing resources, as well as state-of-the-art open-source frameworks [4].

The Convolutional Neural Network (CNN) refers to a type of deep learning technique with multiple layers [5]. CNN is based on the idea of extracting local features from input (typically an image) at the upper layers and incorporating them into more complicated features at the lower layers [6]. Nevertheless, it is computationally expensive because of its multi-layered architecture. In addition, its training, which includes networks on a big dataset, may take a long time.

This paper aims to develop an effective method to work on image data of ArSL gestures and automatically recognize thirty-two different Arabic sign images, eliminating the requirement for human intervention throughout the interpretation. Also, the Linear Discriminant Analysis (LDA) technique is offered as a promising candidate for feature extraction. The LDA method turns the pre-processed hand image matrix into a low-dimensional space; it is employed in the extraction of features. The LDA approach is an excellent algorithm since its features are scale and rotation invariant [7]. The biggest challenges facing the researchers in sign language recognition are the variation of illumination of images captured and the variation of background (mostly non-uniform background) [8].

There are two main contributions present in this paper. First: Designing one-dimensional CNN architecture to properly handle images for gestures extracted from videos. The LDA

extracts low-level features of hand gestures and uses them as input to the CNN. The CNN is made up of numerous completely connected conventional layers as a typical multilayer neural network where a machine learning classifier is added as an additional layer on the CNN.
Second: This paper focuses on the Gesture of ArSL Recognition for impaired people. To our knowledge, this work is the first one that focuses on interpreting the sign language words instead of the alphabet and focusing on the ArSL words with various illumination and background.

The remaining sections of the paper are arranged as follows: Section 2 focuses on the earlier works, while the theoretical background and design are presented in Section 3. Next, Section 4 introduced the architecture of the suggested method. Then, the results are introduced in Section 4. Finally, Section 6 presents on the conclusion.

2. Related Works

Arabic Sign language still finds wide interest from researchers and many challenges need to be addressed. In this section, we list some previous works that concern this field.

In [9], the researchers created a model to analyse 1400 hand gestures made by 20 users for 28 ArSL letters, and they utilized the Principle Component Analysis (PCA) algorithm to simplify the enormous dataset by removing redundant, irrelevant, or erroneous data due to noise. The 103 collected data points for each gestured letter were reduced to 36, resulting in a data variance of more than 99%. The accuracy for recognizing the relevant Arabic sign language letters from the test data was 86% using the SVM technique [9].

In [10], introduced a model for Arabic Sign Language Recognition Based on Convolutional Neural Networks for recognizing 28 Arabic letters and digits from 0 to 10 from a dataset of 7869 images. The proposed model had seven layers and was trained multiple times on different training-testing variations, and the highest accuracy achieved was 90.02 %. The authors presented a comparison with traditional approaches based on K-Nearest Neighbours (KNN) and support vector machines (SVM).

In [11],the authors presented a paper explaining the method for fine-tuning deep convolutional neural networks using a database containing 54,049 images by first constructing models that match the VGG16 and ResNet152 structures and then using the pre-trained models. This study produced results that were up to 99% accurate, although it relied on the VGG16 and ResNet152 models rather than a system created by the researcher.

In [12], an auto-recognition of ArSL letters is suggested, with each letter converting to the appropriate voice. A dataset of 100 photographs was used in the training phase, and 25 images were used in the testing phase for each of ArSL's 31 alphabets. The suggested CNN's architecture consists of two layers and two maximum pools, and the system is trained over 100 epochs. This system has a 90%accuracy rate.

In [13] presented a database for automatic ArSL recognition made up of 15,360, 720*960*3 photos that use a faster, region-based convolutional neural network to recognize and localize the Arabic sign language alphabet. To extract and map picture features as well as learn the position of a hand in a given image, the faster R-CNN was designed. To encode the sign descriptors, the suggested method tends to choose relevant features and implement segmentation to determine the hand region. This idea achieved 93% accuracy rate. The VGG-16 and ResNet-18 models were used in the proposed method.

Design and Implementation Procedure

The proposed system is divided into three stages: pre-processing, Arabic alphabet sign language features extraction, and classification based on a new suggested method, which is a combination of LDA and deep neural network called (LDA-CNN) model. Each stage contains multiple processes that perform different activities. All the processes of this project, as well as the procedures for collecting and evaluating the data, are depicted in Figure (1). Also, algorithm 1 shows the steps for the recognition of ArSL.

Algorithm (1): Proposed Model Stages
Input: Captured Image
Output: classification of thirty-two letters.
BEGIN **Step 1:** Pre-Processing Stage. **Step 2:** Implementation of LDA algorithm and return Features Vector **Step 3:** Split and Prepare Dataset into Training and Testing **Step 4:** Train the CNN model (layers, activation function, hidden neurons). Return model and save weights **Step 5:** Test the Prediction model and return the predicted class. Partnership the return predicted classes to produce the final class. **Step 6:** Evaluation of the results. **END**

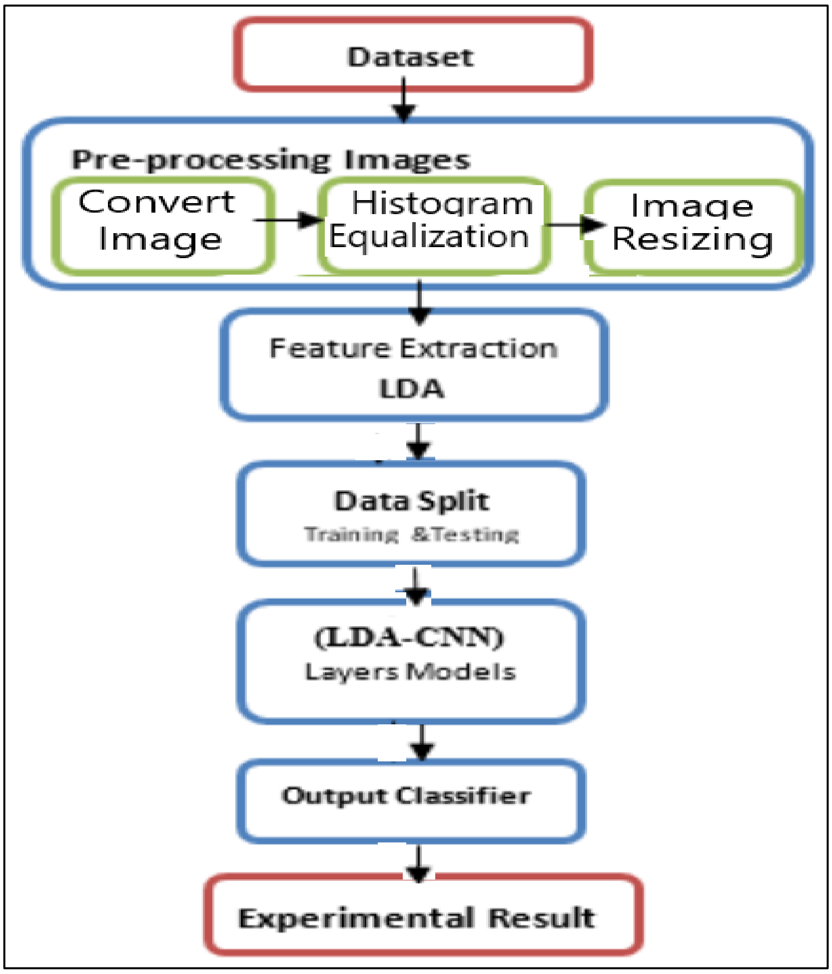

Figure 1: The block diagram of the proposed system

3.1 Images Dataset Used

In this study, the ArSL2018 dataset [14] was used. This dataset is a fully categorized and validated dataset of ArSL alphabet images launched at Prince Muhammad Bin Fahd University, Al Khobar, Saudi Arabia. It is available for researchers in machine learning as well as deep learning. The ArSL dataset contains 54049 grayscale images, each having a 64 * 64 dimension. The images are 8-bit JPG images of thirty-two character ArSL letters (thirty-two classes). In a specific setting, the utilization of various sorts of images and lighting. Various styles of photographs were generated using distinct lighting and backgrounds. Figure 2 shows the hand signs that represent a form of Arabic letters. Also, Table 1 demonstrates the pronunciation of the alphabets and the number of images for each letter in the datasets.

In this proposal, different data collected from different sources was used to test the model in addition to the ArSL2018 dataset, most of which differs from the previous data on which the system was trained. This data consists of 158 colour images of 224 * 224 size, a sample of it is illustrated in Figure 3.

Figure 2: Sample of Arsl2018 Image Dataset.

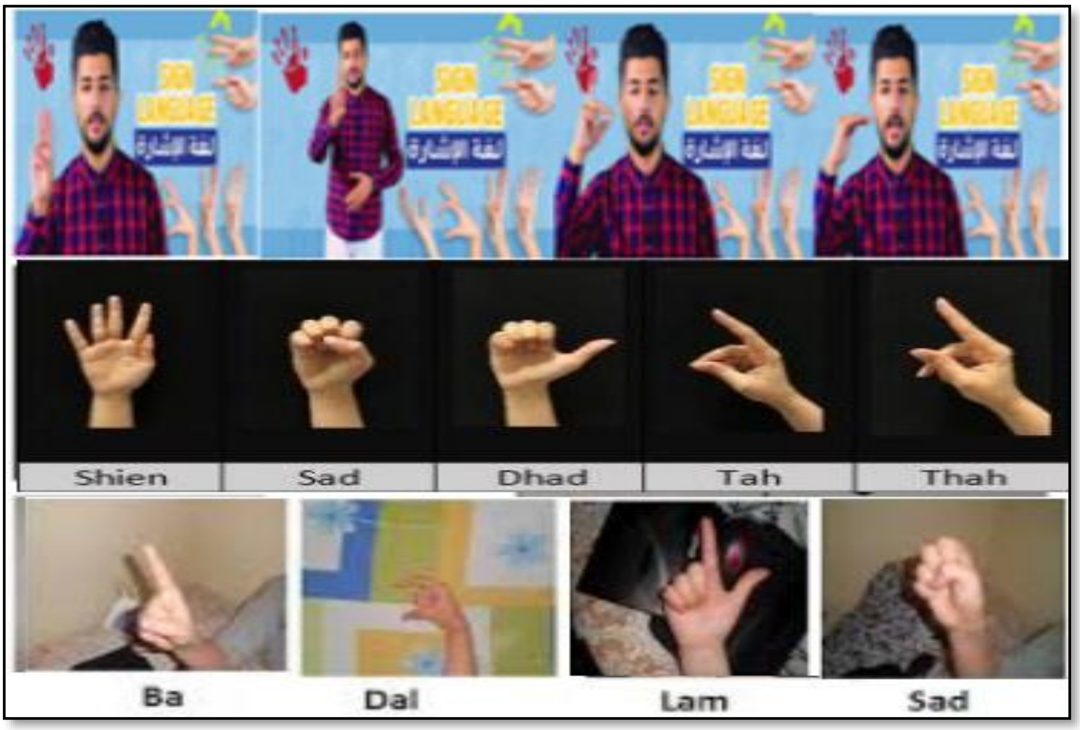

Figure 3 : Sample for Collection Dataset

Table 1: Pronunciation of Each Alphabet with A Corresponding Number of Images.

#	Letter name in English Script	Letter name in Arabic script	# No. of Images	#	Letter name in English Script	Letter name in Arabic script	# No. of images
1	Alif	(أَلِف) أ	1672	17	Zā	(ظَاء) ظ	1723
2	Bā	(بَاء) ب	1791	18	Ayn	(عَين) ع	2114
3	Tā	(أتَاء) ت	1838	19	Ghayn	(غَين) غ	1977
4	Thā	(ثَاء) ث	1766	20	Fā	(فَاء) ف	1955
5	Jīm	(جِيمْ) ج	1552	21	Qāf	(قَاف) ق	1705
6	Hā	(حَاء) ح	1526	22	Kāf	(كَاف) ك	1774
7	Khā	(خَاء) خ	1607	23	Lām	(لَامْ) ل	1832
8	Dāl	(دَالْ) د	1634	24	Mīm	(مِيمْ) م	1765
9	Dhāl	(ذَال) ذ	1582	25	Nūn	(نُون) ن	1819
10	Rā	(رَاء) ر	1659	26	Hā	(هَاء) ه	1592
11	Zāy	(زَاي) ز	1374	27	Wāw	(وَاو) و	1371
12	Sīn	(سِينْ) س	1638	28	Yā	(يَا) ئ	1722
13	Shīn	(شِينْ) ش	1507	29	Tāa	(ة) ة	1791
14	Sād	(صَادْ) ص	1895	30	Al	(ال) ال	1343
15	Dād	(ضَاد) ض	1670	31	Laa	(لا) لا	1746
16	Tā	(طَاء) ط	1816	32	Yāa	(يَاء) يَاء	1293

3.2 Image Pre-processing

- Convert the input RGB images to a grayscale image. Grayscale images have only one dimension (channel) and are composed of only gray shades of colours (256 grey colours) with an 8-bit representation. This is done with almost any collected dataset, but the ArSL2018 dataset is already grayscale images.
- Image processing used to highlight a particular element of an image is known as image enhancement. The following scenarios involve the use of image enhancement: removing the image's noise, the image becomes more sombre when the dark areas are enhanced, and the edges of the images are highlighted. For some specific uses, the outcome is better than the original image. To improve the image contrast and increase the global contrast on images, the histogram equalization technique is used to adjust image intensities. Due to its simplicity and relative superiority over other conventional methods, this strategy is typically used for image enhancement paradigms.
- The image size clearly affects the detection accuracy; thus, we test to determine which size will improve the accuracy. The most efficient size to utilize is 20 * 20, which reduces computing time and cost. Therefore, this is the optimal size that produces the best results in this proposal. The two types of datasets used to run the test are depicted in Table 3.

3.3 Features Extraction

The process of extracting features from a set of images is an initial stage that will be used as input to the one-dimension Convolutional Neural Network (CNN) or machine learning algorithms, where it extracts the important distinction from the most important. Linear methods are used for this process by performing a linear mapping of data to an area of lesser dimensions. The most common linear method for feature extraction is LDA, which maximizes the component axes for class separation between multiple classes. The feature extraction is performed with the aid of the LDA method, which converts the pre-processed hand image

matrix into a low-dimensional space. Three steps must be followed to achieve the objective of feature extraction. Estimating disengagement between multiple classes (the gap between their average values), also known as the inter-class matrix or distinction, is the initial step.
The subsequent stage is determining the within-class matrix or variance, which is the difference between the samples and the mean of the hand images for each class. Finally, the lower-dimensional space that has two effects will be generated: one is to reduce the dimension in the within-class variance, and the second is to increase the between-class variance. All the mentioned steps are explained in detail in algorithm 2.
In summary, an LDA's primary objective is to keep the class-discriminatory information while projecting a feature space (a collection of n-dimensional samples) onto a smaller subspace kk (in which k≤n−1, k≤n−1). In general, dimensionality reduction can help minimize overfitting by decreasing the error in parameter estimation (the "curse of dimensionality"), in addition to lowering computational costs for a specific classification task.

Algorithm (2): Linear Discriminant Analysis
Input: hand image
Output: Feature Vectors
BEGIN **Step1:** Read hand images. **Step2:** Calculate the average value of every one of the classes using Equation 1 $\mu_j = \frac{1}{n_j}\sum_{x_i \in \omega_j} x_i$ …….1 Where: N: The total number of samples. *ni* represents the number of samples of the *i*th class. **Step3:** Compute total mean of all database as in Equation 2 $\mu = \frac{1}{N}\sum_{i=1}^{N} x_i = \sum_{i=1}^{c} \frac{n_i}{N}\mu_i$…………..2 μ_i represents the projection of the mean of the *ith.* μ Projection of the total mean of all classes **Step4:** Obtain between-class matrix S_B $(M\times M)$ as Equation 3 $S_B = \sum_{i=1}^{c} n_i(\mu_i - \mu)(\mu_i - \mu)^T$ ….3 S_B between-class variance **Step5:** Obtain within-class matrix S_W $(M \times M)$, as Equation 4 $S_W = \sum_{j=1}^{c}\sum_{i=1}^{n_j}(x_{ij} - \mu_j)(x_{ij} - \mu_{j)})^T$ ......4 S_{Wi} the within-class variance of the *i*th class, represents the difference between the mean **Step6:** Transformation matrix (*W*) of the approach of LDA is computed according to Equation is referred to as Fisher's criterion. $W = S_W^{-1}S_B$ **Step7:** The Eigen values (λ) and Eigen vectors (*V*) of *W* are computed. **Step8:** Sorting eigenvalues in decrement order according to their equivalent Eigen values. The first *k* Eigen vectors are utilized as a lower dimensional space V_k. **Step9:** Employ every original sample (*X*) to the LDA's lower dimensional space as in Eq. $Y = XV_K$ **Step10:** Return feature vectors. **END**

3.4 Split the Data

The process of splitting the data is based on dividing the complete data into two parts: 70% for training data and 30% for testing data. The purpose of this process is to allow us to validate the evaluation data.

3. LDA-CNN Model

This study proposed a novel algorithm based on feature extraction using Linear Discriminant Analysis (LDA) and input of these features to the 1D Convolutional Neural Networks (1D CNNs) classifier method. Figure 3 depicts the proposed model's architecture. This architecture consists of eight one-dimension convolution layers, seven activation function layers, six max-pooling layers, five dense layers, and one flattening layer. Table 2 summarizes the proposed model's representation. More than one programming language was used, namely Visual Studio, language C++ and Python, in this work. The goal was to create a smart application that is extremely accurate and with less computing time that can be used in real-time applications.

1. **1D convolution layer:** The convolutional layer's purpose is to recognize an image's features. Typically, it progresses from the broad (for instance, shapes) to the precise. This solution makes use of 1D CNNs, a modified form of 2D CNNs. Since 1D CNN have a minimal processing need, they are well suited for real-time applications. A single spatial (or temporal) dimension is passed through a convolution kernel created by a 1D convolution layer to produce a tensor of outputs.
2. **Pooling:** In between the convolution layers, pooling layers are frequently added. By doing this, the number of parameters is decreased, and over-fitting is avoided.
The max-pooling technique was utilized since the convergence rate is faster than other subsampling techniques. It summarizes the features present in a region of the feature map generated by a convolution layer.
3. **The Rectified Linear Unit:** this is not an isolated component but is a supplementary stage to the CNN. In general, the image consists of a lot of non-linear features such as edge, borders, colour, and any transition pixels. In this proposal, the leaky rectangular linear unit (Leaky ReLU) is employed to increase the non-linearity in images. An activation function known as a Leaky ReLU is based on a ReLU but has a tiny slope for negative values as opposed to a flat slope. Prior to training, the slope coefficient is calculated. This led to the addition of a Leaky ReLU activation function before the pooling layer and after each convolution layer.
4. **Activation Function**: The Softmax activation function is employed to predict the output of the fully connected layer. Here, a vector of K real values between zero and one is transformed by the Softmax function (the summation is always equal to 1). Some other experiments were carried out using the Sigmoid function. It decides which value to classify as output and what not to pass.

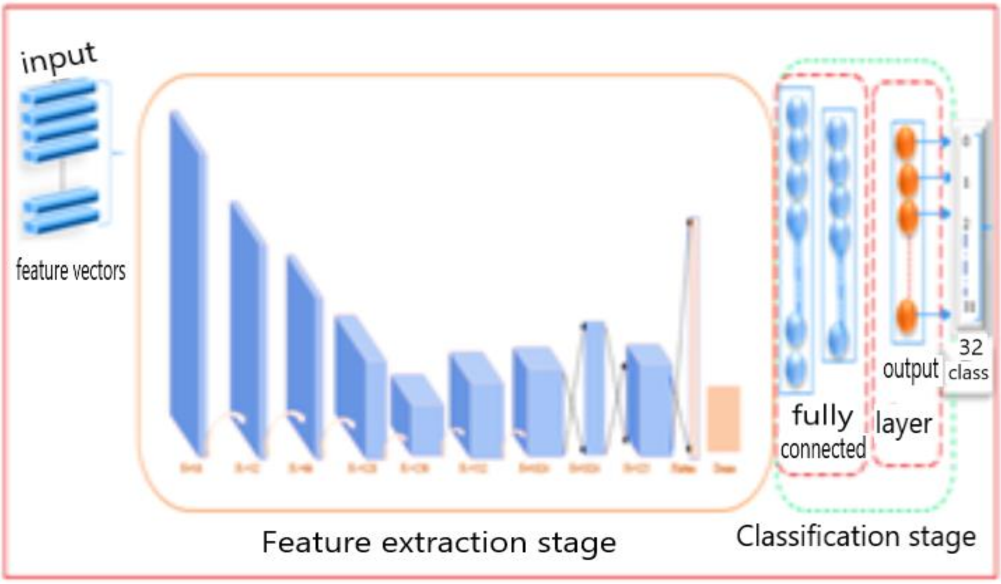

Figure 4: The proposed architecture of CNN used to classify alphabet ArSL.

Table 2: Summary of components for the proposed model's architecture

Block no.	Layer type	Output shape	Params no.
1	conv1d_1(Conv1D-size(3,1))	(None,29,16)	64
	leaky_re_lu_1(LeakyReLU)	(None,29,16)	0
	pooling1d_1(MaxPooling1)	(None,14,16)	0
2	conv1d_2(Conv1D-size(3,1))	(None,12,32)	1568
	leaky_re_lu_2 (LeakyReLU)	(None,14,16)	0
	pooling1d_2(MaxPooling1)	(None,6,32)	0
3	conv1d_3 (Conv1D)	(None, 4, 64)	6208
	leaky_re_lu_3 (LeakyReLU)	(None, 4, 64)	0
	pooling1d_3 (MaxPooling1)	(None, 4, 64)	0
4	conv1d_4 (Conv1D)	(None, 4, 128)	8320
	leaky_re_lu_4(LeakyReLU)	(None, 4, 128)	0
	pooling1d_4 (MaxPooling1)	(None, 4, 128)	0
5	conv1d_5 (Conv1D)	(None, 4, 256)	33024
	leaky_re_lu_5 (LeakyReLU)	(None, 4, 256)	0
	pooling1d_5 (MaxPooling1	(None, 4, 256)	0
6	conv1d_6 (Conv1D)	(None, 4, 512)	131584
	leaky_re_lu_6 (LeakyReLU)	(None, 4, 512)	0
	pooling1d_6 (MaxPooling1	(None, 2, 512)	0
7	conv1d_7 (Conv1D)	(None, 2, 1024)	525312
	leaky_re_lu_7 (LeakyReLU)	(None, 2, 1024)	0
8	fc_1 (Dense)	(None, 2, 512)	524800
	fc_2 (Dense)	(None, 2, 256)	131328
	fc_3 (Dense)	(None, 2, 256)	65792
	fc_4 (Dense)	(None, 2, 128)	32896
	conv1d_8 (Conv1D)	(None, 2, 185)	23865
9	flatten_1 (Flatten)	(None, 370)	0
10	Fc-out(Dense)	(None, 32)	11872
Total params:		**1,496,633**	
Trainable params:		**1,496,633**	
Non-trainable params:		**0**	

4. Experimental Results for LDA-CNN Proposal Model

The proposed model was implemented in two consequent phases: the feature extraction phase using LDA, and the CNN Model phase. The objective of this model is to recognize thirty-two Arabic letters. The suggested model's LDA-CNN model phase was tested on two types of datasets (ArSL2018 and collected dataset).

The image size has a clear effect on the detection accuracy, so, it was tested to select the best size that can be used to increase the accuracy. The test was implemented with the two types of the datasets as shown in Table 4. Size 20x20 is the best size that can be used to reduce the computation cost and time.

Table 4: shows the results of the image size selection test.

Type of Dataset	No. Images	Image size	Accuracy %	Loss Fun.
ArSL 2018	54,049	64*64 (original size)	66.85	1.35
		60*60	68.88	1.115
		40*40	69.01	1.05
		20*20	**72.89**	1.012
Dataset collected	158	224*224 (original size)	34.00	2.76
		60*60	92.41	0.28
		40*40	93.04	0.24
		20*20	**94.94**	0.18

The second test was implemented to find the best number of epochs that can be used to maximize detection accuracy. Results are shown in Table 3. Although the accuracy is very close when the number of the epoch was 100 and 200, we recommend using the number of epochs equal to 200, which gives very good results (with both ArSL2018 and collected dataset) without significant effect on the training time as shown from Figure 6 that shows the training run time.

The third test was implemented to a distinction between two types of activation functions: Softmax and Sigmoid. We know as the default that the Softmax is more suitable when the desired results are more than two classes, but due to the use of the sigmoid function by some authors when detecting the ArSL so the possibility of using the Sigmoid instead of Softmax was considered. From Figure 5 it is clear that Softmax is better to use in this proposal (that has a very low loss function).

Table 3: Effect of epoch numbers on the detection accuracy

No. epochs	Accuracy %		Loss Fun.	
	ArSL2018	Collected Dataset	ArSL2018	Collected Dataset
10	72.49	37.34	1.065	2.5433
50	95.59	95.57	0.172	0.0997
100	97.00	96.00	0.098	0.0740
200	99.98	98.57	0.0023	0.0682

The fourth test was implemented to find the run time according to the number of the epoch. This time is the training time for each batch size of 1024. The results are shown in Figure 6. Where the test time is a very trivial time for each image.

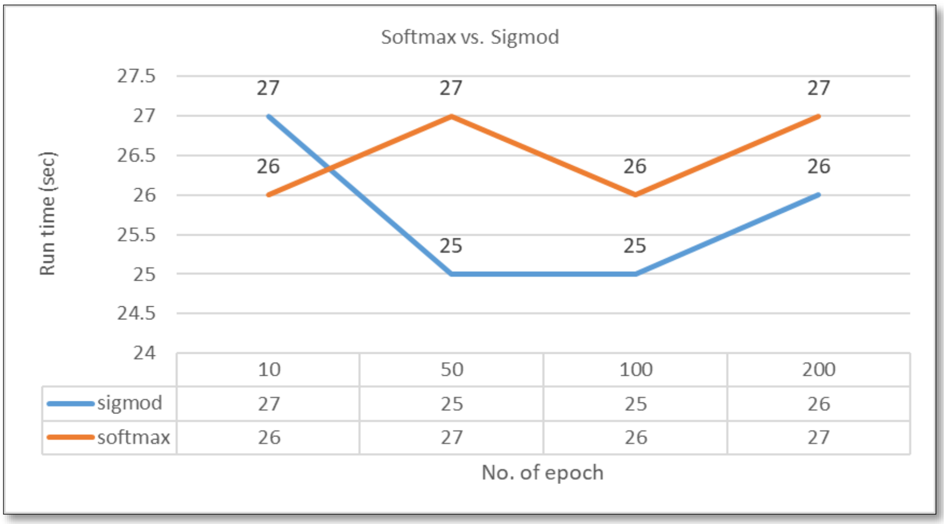

Figure 5: The results of testing the activation functions.

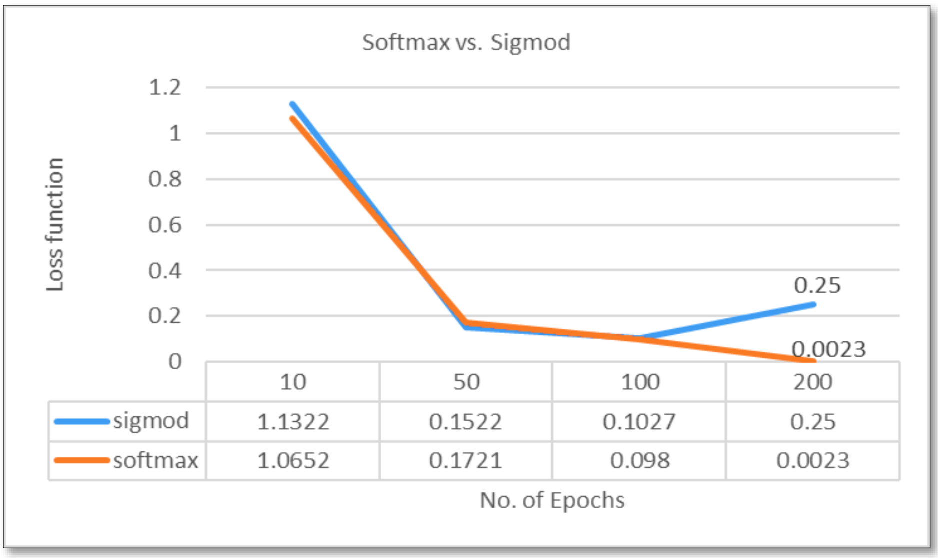

Figure 6: CNN training Time Analysis

The fifth test was to measure the best learning rate that gives the best accuracy. Table 5 summarizes the findings. A learning rate with a value of 0.001 gives the best accuracy for both datasets. Increasing the learning rate will increase the run time.

Table 5: Testing Accuracy vs. Learning rate.

initial learning rate		0.01	0.001	0.0001
Accuracy %	**(ArSL2018) Dataset**	33	96	64
	(Collection Dataset)	31	95	96

Finally, we compared the proposed results with other similar works, the comparison was with papers that used the standard dataset (ArSL2018) (which is created a few years ago) and the collected dataset that was collected by authors, as shown in Table 6.

Table 6: Comparison results with previous works

index	Model	Accuracy	Dataset
1	VGG-16 [11]	99%	ArSL2018
2	ResNet-18 [13]	93%	ArSL2018
3	CNN [15]	96.4	ArSL2018
4	CNN [7]	92.9	ArSL2018
5	CNN [16]	88.87	ArSL2018
6	**Proposal Model**	**99.98%**	**ArSL2018**
1	CNN Model [12]	90%	Collected Dataset
2	PCNN [17]	90.4	Collected Dataset
3	ANFIS networks [18]	93.55	Collected Dataset
4	R-CNN [13]	93	Collected Dataset
5	CNN [12]	90	Collected Dataset
6	**Proposal Model**	**98.57**	**Collected Dataset**

5. Conclusion

Due to its detrimental impact on a person's capacity to interact with others, deafness has an adverse impact on all facets of a person's everyday life. The model that is being provided is an illustration of how a Convolutional Neural Network (CNN) can be used for image recognition as well as classification in a specific technique. The proposed system gets better performance by enriching the training dataset through adding new sample versions from the already available samples to provide a generalization to the model. We developed and applied an Arabic Sign Language (ArSL) recognition system, which used Linear Discriminant Analysis (LDA) and 1D Convolutional Neural Networks (1D CNNs) to detect and recognize the ArSL alphabet, considering the strengths and drawbacks of previous contributions. The suggested model (LDA-CNN) gives very good results in detecting the ArSL, which reached 99.9%.

This paper's primary contribution is introducing a new method that uses one-dimension CNN after extracting the features, which increases the accuracy significantly. Also, the suggested method processes the images with variant illumination and variant image backgrounds. The proposed method achieved very good results compared with other methods in the same field. Constructing a database of ArSL words and classifying them from future works, as well as distinguishing the sign through moving images, is a progressive work.

Reference

[1] Maha S. Altememe, Nidhal K. El Abbadi, "A Review for Sign Language Recognition Techniques," in *1st. Babylon International Conference on Information Technology and Science 2021 (BICITS 2021)*, Babil- IRAQ, 2021.

[2] W. H. Organization, "Deafness and hearing loss," 28 Oct 2019. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss. [Accessed 06 OCT 2021].

[3] B. S. Parton, "Sign Language Recognition and Translation: A Multidisciplined Approach From the Field of Artificial Intelligence," *The Journal of Deaf Studies and Deaf Education*, vol. 11, no. 1, pp. 94-101, 2006.

[4] I. H. a. R. P. Wildes, What Do We Understand About Convolutional Networks?, Canada: York University, 2018.

[5] S. Jansen, "Convolutional Neural Networks," in *Hands-On Machine Learning for Algorithmic Trading: Design and implement*, UK, packt publishing, 2018, p. 496.

[6] Ruiyang Zhang,Yang Liub, Hao Suna, "Physics-guided Convolutional Neural Network (PhyCNN) for Data-driven Seismic Response Modeling," *Preprint submitted to Elsevier,* vol. arXiv:1909.08118v1 , no. [eess.SP], p. 25, September 19, 2019.

[7] Arslan Arif and Syed Tahir Hussain Rizvi and Iqra Jawaid and Muhammad Adam Waleed and Muhammad Raheel Shakeel, "An American Sign Language (ASL) Translator," in *International Conference on Control, Decision and Information Technologies (CoDIT)*, DOI:10.1109/CoDIT.2016.7593642, 2016.

[8] Wang, Zilong ; Cha, Young-jin, "Automated damage-sensitive feature extraction using unsupervised convolutional neural networks," *Proceedings of the SPIE*, vol. 10598, no. id. 105981J 7 pp. DOI: 10.1117/12.2295966 , p. 8, 2018.

[9] M. A. A. a. H. Al-Nuaim, "Recognizing Arabic Sign Language gestures using depth sensors and a KSVM classifier," in *2016 8th Computer Science and Electronic Engineering (CEEC)*, Colchester, UK, doi: 10.1109/CEEC.2016.7835904., 2016.

[10] S. Hayani, M. Benaddy, O. El Meslouhi and M. Kardouchi, "Arab Sign language Recognition with Convolutional Neural Networks," in *International Conference of Computer Science and Renewable Energies (ICCSRE)*, pp. 1-4, doi: 10.1109/ICCSRE.2019.8807586, 2019.

[11] Yaser Saleh (∗), Ghassan F. Issa, "Arabic Sign Language Recognition Through Deep Neural Networks Fine-Tuning," *International Journal of Online and Biomedical Engineering (iJOE)*, vol. 16, no. 2626-8493, https://doi.org/10.3991/ijoe.v16i05.13087, p. No. 5, 2020.

[12] M. M. Kamruzzaman, "Arabic Sign Language Recognition and Generating Arabic Speech Using Convolutional Neural Network," *Hindawi Wireless Communications and Mobile Computing*, vol. Volume 2020, no. Article ID 3685614, doi.org/10.1155/2020/3685614, p. 9, 2020.

[13] Rahaf Abdulaziz Alawwad, Ouiem Bchir, Mohamed Maher Ben Ismail, "Arabic Sign Language Recognition using Faster R-CNN," *International Journal of Advanced Computer Science and Applications* , Vols. Vol. 12,No. 3, no. (DOI) : 10.14569/IJACSA.2021.0120380, p. 692, 2021.

[14] Ghazanfar Latif, Nazeeruddin Mohammad, Jaafar Alghazo, Roaa AlKhalaf, Rawan AlKhalaf, "ArASL: Arabic Alphabets Sign Language Dataset," *ELSEVIER,Data in Brief*, vol. Volume 23, no. 2352-3409,https://doi.org/10.1016/j.dib.2019.103777, p. 103777, April 2019.

[15] Mohamed Mohandes,Mohamed Abdelouaheb Deriche,Salihu Oladimeji Aliyu, "ARABC SIGN LANGUAGE RECOGNITION USING MULTI-SENSOR DATA FUSION," *U.S. PATENT DOCUMENTS*, vol. No. 62, no. 113.276, p. 17, 2016.

[16] Salma A. Essam El-Din and Mohamed A. Abd El-Ghany, "Sign Language Interpreter System: An alternative system for machine learning," in *2nd Novel Intelligent and Leading Emerging Sciences Conference (NILES)*, DOI:10.1109/NILES50944.2020.9257958, 2020,pp(332-337).

[17] Tolba, Mohamed & Abdel-wahab, Mohamed & Aboul-Ela, Magdy & Samir Roshdy, Ahmed, "Image Signature Improving by PCNN for Arabic Sign Language Recognition," *Canadian Journal on Artificial Intelligence, Machine Learning and Pattern Recognition*, Vols. Volume 1. 1-6., no. AM publisher Corporation canada, p. 8, 2014.

[18] Omar Al-Jarrah, Alaa Halawani, "gestures in Arabic sign language using neuro-fuzzy systems," *Artificial Intelligence*, p. 117–138, 2001.

[19] A. M. Kadim and W. R. Saleh, "Morphological and Optical Properties of CdS Quantum Dots Synthesized with Different pH values," *Iraqi Journal of Science* , vol. 58, no. 3A, pp. 1207-1213, 2017.

[20] Q. Zhang and H. Li, "MOEA/D: A Multiobjective Evolutionary Algorithm based on Decopposition," *IEEE Transactions on Evolutionary Computation* , vol. 11, no. 6, pp. 712-731, 2007.

[21] S. Romano, J. Baily, V. Nguyen and K. Verspoor, "Standarized Mutual Information for Clustering Comparisions: One Step Further in Adjustment for Chance," in *Proceedings of the 31st Intenational Conference on Machine Learning*, Beijing, 2014.

[22] J. Brownlee, Clever Algorithms: Nature-Inspired Programming Recipes, Autralia: LuLu Enterprice , 2011, p. 436.

[23] H. O. A. (. Procedures), "Code of Practice for the Housing and Care of Animals Used in Scientific Procedures.," Act 1986.

[24] C. o. P. E. (COPE), "Code of Conduct and Best-Practice Guidelines for Journal Editors," 2011.

[25] N. Tajbakhsh, J. Y. Shin, S. R. Gurudu, R. T. Hurst, C. B. Kendall, M. B. Gotway and J. Lian, "Convolutional Neural Networks for Medical Image Analysis: Full Training or Fine Tuning," *IEEE,* vol. 35, no. 5, pp. 1299 - 1312, 2016.

[26] M. Arif WaniFarooq Ahmad BhatSaduf AfzalAsif Iqbal Khan, " Advances in Deep Learning," in *Basics of Supervised Deep Learning*, Springer,DOI: 10.1007/978-981-13-6794-6_2, 2020, pp. pp 13-29.

[27] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, Andrew Rabinovich; , "Going Deeper With Convolutions," in *the IEEE Conference on Computer Vision and Pattern Recognition (CVPR),*, Boston, 2015, pp. 1-9.

[28] Ala addin I. Sidig, Hamzah Luqman, and Sabri A. Mahmoud, "Arabic Sign Language Recognition Using Optical Flow-Based Features and HMM," in *Recent Trends in Information and Communication Technology, Lecture Notes on Data Engineering and Communications Technologies*, DOI 10.1007/978-3-319-59427-9_32, Springer International Publishing AG 2018.

[29] Yann LeCun, Koray Kavukcuoglu and Cl´ement Farabet, "Convolutional Networks and Applications in Vision," in *Proceedings of 2010 IEEE International Symposium on Circuits and Systems*, Paris, France,DOI: 10.1109/ISCAS.2010.5537907, 2010.

[30] A.I. Shahin1 , Sultan Almotairi, "Automated Arabic Sign Language Recognition System Based on Deep Transfer Learning," *IJCSNS International Journal of Computer Science and Network Security,* vol. VOL.19, p. No.10, 2019.

[31] Ja Hyung Koo,Se Woon Cho,Na Rae Baek,Min Cheol Kim andKang Ryoung Park, "CNN-Based Multimodal Human Recognition in Surveillance Environments," *Sensors is the leading international,* vol. 18, no. 3040, doi.org/10.3390/s18093040, p. 9, 2018.

[32] HOANG, Toan Minh, et al, "Deep retinanet-based detection and classification of road markings by visible light camera sensors," *Sensors,* vol. 19, no. 281, p. 2, 2019.

[33] Nada B. Ibrahim, Mazen M. Selim, Hala H. Zayed, "An Automatic Arabic Sign Language Recognition System (ArSLRS)," *Journal of King Saud University - Computer and Information,* vol. 30, no. 4,doi.org/10.1016/j.jksuci.2017.09.007, pp. 470-477, 2018.

[34] Subhash Chand Agrawal, Anand Singh Jalal and Rajesh Kumar Tripathi, "A survey on manual and non-manual sign language recognition for isolated and continuous sign," *International Journal of Applied Pattern Recognition,* Vols. Vol. 3, No. 2, no. 3, pp. 99-134, 2016.

[35] Mohamed Aktham Ahmed, Bilal Bahaa Zaidan,Aws Alaa Zaidan, Mahmood Maher Salih, Muhammad Modi bin Lakulu, "A Review on Systems-Based Sensory Gloves for Sign Language Recognition State of the Art between 2007 and 2017," *Sensors,* vol. 18, no. doi: 10.3390/s18072208, p. 44, 2018.