Intelligent Automation & Soft Computing

Tech Science Press

DOI: 10.32604/iasc.2023.038235
Article

A Robust Model for Translating Arabic Sign Language into Spoken Arabic Using Deep Learning

Khalid M. O. Nahar[1], Ammar Almomani[2,3*], Nahlah Shatnawi[1] and Mohammad Alauthman[4]

[1]Department of Computer Sciences, Faculty of Information Technology and Computer Sciences, Yarmouk University–Irbid, 21163, Jordan
[2]School of Computing, Skyline University College, Sharjah, P. O. Box 1797, United Arab Emirates
[3]IT-Department-Al-Huson University College, Al-Balqa Applied University, P. O. Box 50, Irbid, Jordan
[4]Department of Information Security, Faculty of Information Technology, University of Petra, Amman, Jordan
*Corresponding Author: Ammar Almomani. Emails: ammarnav6@bau.edu.jo, ammar.almomani@skylineuniversity.ac.ae
Received: 03 December 2022; Accepted: 12 April 2023; Published: 23 June 2023

Abstract: This study presents a novel and innovative approach to automatically translating Arabic Sign Language (ATSL) into spoken Arabic. The proposed solution utilizes a deep learning-based classification approach and the transfer learning technique to retrain 12 image recognition models. The image-based translation method maps sign language gestures to corresponding letters or words using distance measures and classification as a machine learning technique. The results show that the proposed model is more accurate and faster than traditional image-based models in classifying Arabic-language signs, with a translation accuracy of 93.7%. This research makes a significant contribution to the field of ATSL. It offers a practical solution for improving communication for individuals with special needs, such as the deaf and mute community. This work demonstrates the potential of deep learning techniques in translating sign language into natural language and highlights the importance of ATSL in facilitating communication for individuals with disabilities.

Keywords: Sign language; deep learning; transfer learning; machine learning; automatic translation of sign language; natural language processing; Arabic sign language

1 Introduction

Sequence labeling is one of the critical and challenging problems in Natural Language Processing (NLP). A suggested sequence labeling framework based on the concept of latent variables in a random field by Shao et al. [1]. It can capture the hidden variable structure in the row data, which can be used for clustering and labeling data. Meanwhile, by Lin et al. [2], a semi-Markov model for sequence labeling was constructed. The model incorporates both character and word levels. The model is essential in extracting meaningful information from fewer ones in segment representation. The model performance was evaluated, and its effectiveness was proved. Sequence labeling was used in Machine

Translation (M.T.), such as the one used by Tebbifakhr et al. [3], which was a multitask Neural Machine Translation (NMT) adaptation capable of processing multiple tasks within a single system. The Arabic-English Machine Translation (AEMT) used in translating spoken Arabic into equivalent English language sentences highly depends on speech tagging and sequence labeling. Zakraoui et al. [4] an evaluation using multiple evaluation criteria. The aim was to improve AEMT. Zakraoui et al. [4] found that Neural AEMT outperforms many approaches. Moreover, the approach proves that the performance of AEMT depends on the quality of the targeted dataset.

Individuals in the deaf community communicate using Sign Language (S.L.). However, most non-def persons didn't know the meaning of these signs since they were not concerned about learning them unless they needed them. Moreover, some deaf persons want to say something through the internet, via video conference, for example, or through media, which makes it very beneficial to translate the signs directly to the target language through a webcam. Many things motivate us to think and perform this research, primarily for the Arabic language.

Sign Language (S.L) is a manual representation of language that utilizes a signed vocabulary to convey concepts, as highlighted in the works of El-Bendary et al. [5] and Firas ibrahim et al. [6] S.L. is crucial for communication with individuals who have hearing difficulties and enables them to interact with their community. However, the issue of widespread unfamiliarity with S.L. highlights the need for automated systems that can translate S.L. into written and spoken language and vice versa, as emphasized by Mohandes et al. [7].

S.L. transformation systems differ in ability and accuracy in identifying a single letter corresponding to a hand gesture or recognizing the full sentence corresponding to continuous hand movements. Automatic transformation of Arabic sign language (ArSL) into natural Arabic language and vice versa is still an uncovered area and needs further research, Al-Ayyoub et al. [8].

ArSL has more than 9000 gestures and uses 26 static hand postures and 5 dynamic gestures to represent the Arabic alphabet, Tolba et al. [9]. Many techniques and approaches were employed to transform ArSL into a natural Arabic language. On the one hand, some approaches were based on electronic sensors as a part of a Computer Vision process. For instance, Mohandes et al. [10]. Used a Leap Motion Controller (LMC) to allow a computer system to track and detect hands and fingers, acquiring gesture and position information. On the other hand, other approaches to transforming S.L. were based only on image processing techniques. For instance, Tolba et al. [9] and Al-Smadi et al. [11] adopted a graph-matching technique for recognizing continuous sentences.

There are three types of Sign Language Recognition (SLR), by Dipietro et al. and Nadger et al. [12,13]; first, the Alphabet sign recognition system deals with each letter separately. Second, the Isolated Word sign recognition system deals with a sequence of input images of a specific word. The Continuous sign language recognition system deals with a continuous stream of words.

Methods of automatic SLR could be classified into two classes, vision-based SLR approaches, and sensor-based SLR approaches. Vision-based approaches work by processing input video streams from a camera or a pre-captured video sequence. Input streams are tokenized and pre-processed to get valid S.L. gestures, Lv et al. [14] and Sharma et al. [15] SLR uses vision-based cameras only to capture gestures (signs). Since the user is not obliged to wear any devices such as data gloves or motion tracking devices, which leads to easier recognition, this approach is completely sensitive to changes in the background or lighting conditions. The sensor-based approach uses wearable devices to capture signs accurately. One of its weaknesses is that it can be uncomfortable compared to the vision-based approach but gives a subtle recognition of the signs. Electronic gloves and motion tracking sensors are the most popular electronic devices for SLR, Dipietro et al. [12]. Recently, new electronic devices

have been introduced to facilitate human-machine interaction. Namely, Microsoft Kinect was formerly used to interact with Xbox games, and LMC received considerable attention in this field. The Kinect device utilizes infrared emitters, rear sensors, and a high-resolution video camera. LMC utilizes three LEDs and two infrared cameras to capture information. Moreover, LMC is more accurate than the Kinect, although it does not provide images of detected objects, Nadgeri et al. [13].

In vision-based approaches, the valid S.L. gestures of the input stream passed through a machine learner-based model to predict the corresponding natural language text. Sahoo et al. [16], Firas ibrahim et al., and Nahar et al. [6,17] define machine learning as a subfield of artificial intelligence that provides the machine the ability to learn and improve the experience from training sets without the need to be programmed explicitly.

Classification is one of the machine learning methods to predict the right class of a given input based on prior learning from a training set with known labels, Bhatti et al. [18] and Nahar et al. [19]. To achieve more accurate classification, deep learning emerged as an advanced machine learning branch that depends on many hidden layers in a neural network and learns from a vast amount of data, Yen et al. [20]. Transfer learning takes advantage of previously stored knowledge of trained models to employ it in developing a solution for another related problem. Keras is one of the deep-learning libraries. It can support convolutional and recurrent neural networks. It has two main model categories, the Sequential model and the Model class, Bhatti et al. [18]. We use transfer learning to get the benefits of an existing model and to have higher performance results based on an existing trained model in a related domain.

In this research paper, we follow a novel image-based approach based on deep learning techniques for automatically translating ARSL in a continuous mode. We handle the problem of translating ARSL into the natural Arabic language as a classification problem; this paves for more accurate results considering Arabic letters as the classes to be predicted. We adopt a deep learning-based classification approach. The transfer learning technique has been employed to retrain 12 image recognition models to be familiar with the used Arabic sign letters dataset. The recognition process of an input Arabic sign letter is achieved throughout the 12 retrained models. Most of their predictions are considered as corresponding Arabic letters for that input sign letter.

The scientific contributions of this study can be summarized as follow:

- Using transfer learning in ARSL translation based on the majority voting of predictions.
- We are developing an image processing-based technique for hand edge detection in which we recognize hand shapes based on detecting human skin colors and mathematical morphology techniques.
- To improve the model's accuracy, 12 image recognition models were employed to identify the Arabic sign language, presenting a superior methodology.

The rest of this research paper is organized as the following: Section 2 encompasses a review of important related literature on the area of research. The architecture of the adopted ARSL translation model and its working flow is provided in Section 3. Results, discussions, and conclusions are demonstrated in Sections 4 and 5, respectively.

2 Literature Review

Transforming SL into natural language is recognizing and mapping hand gestures, facial expressions, and body movements into correspondence letters, words, or terms. Many scientific researchers

have investigated this problem due to its social importance. We classified the reviewed literature works into three classes, Image-based, Sensor-based, and Deep learning-based approaches.

2.1 Image-Based Approaches

A model for translating the English language to the Pakistani deaf community sign language was proposed by Khan et al. [21]. Manual and automatic evaluations were carried out to measure the accuracy of the translation model, which was 78%.

Assaleh et al. [22] presented a system for continuous ArSL. The dataset comprises 40 sentences carried out by only one user. The Authors used Motion Estimation (M.E.) and Accumulated Differences (A.D.s) approaches to extract features and also used hidden Markov models (HMM) for the classification process. The results appeared that the proposed system achieved an accuracy reaching 94%.

Tharwat et al. [23] developed the ArSL system based on an extract for the gesture from the Arabic sign images. The authors used the Scale Invariant Features Transform (STIF) technique to extract features and the Linear Discriminant Analysis (LDA) technique to solve the dimensionality problem, which increased the accuracy of the suggested system. They used three classifiers. These classifiers are k-Nearest Neighbor (KNN), minimum distance, and support vector machine (SVM). The data set was composed of 30 Arabic signs, with 7 images of each sign collected by Suez Canal University. The results demonstrated that the suggested system obtained an accuracy of 98.9%.

Luqman et al. [24] studied three transformation techniques to extract the description of the features from the whole or part of the theaccumulatedsign's image. These techniques are Hartley, Fourier, and Log-Gabor transforms. Three signers collected the data set composed of 23 signs, and each sign is repeated 50 times by each signer. The authors used three classifiers: KNN, SVM, and MLP. The results demonstrated that the Hartley transform using the SVM classifier achieved high accuracy, about 98.8%, compared to Fourier and Log-Gabor transforms, which achieved 94.9% and 75.5%, respectively.

Elpeltagy et al. [25] suggested a method composed of three phases: hand segmentation, hand shape sequence, body motion description, and sign classifications. They Authors used the Histogram of Oriented Gradients (HOG). They applied Principal Component Analysis (PCA) (HOG-PCA) for hand shape description, canonical correlation analysis (CCA) for hand shape matching, and finally, Random Forest (R.F.) for motion classification. The data collection was collected throughtheAsdaa' the Association for Sophisticating the Deaf in Alexandria, Egypt, composed of 150 isolated ArSL signs, 92 of which used one hand and the remaining used two hands. The results showed that the proposed method achieved an accuracy of 55.57%.

Ibrahim et al. [26] proposed an automatic visual sign recognition language system to translate isolated Arabic word signs into text. The suggested system consists of four phases. The first phase is hand segmentation which uses a skin detector based on the face color. The second phase is tracking using the skin-blob tracking technique, the third phase is geometric features extraction, and the last is classification by Euclidean distance. The data set consists of 450 colored ArSL videos representing 30 isolated words. The results showed that the introduced system achieved an accuracy of 97%.

To translate Arabic text into ArSL, Luqman et al. [27] introduced a rule-based automatic machine translation system. The introduced systems carry out a morphological, syntactic, and semantic analysis of an Arabic sentence to translate it into a sentence with the grammar and structure of ArSL. The Authors used a gloss system to represent ArSL. They used 8 features for each Arabic word. The

data set is composed of 600 sentences. The results showed that the proposed system achieved precise translation for more than 80% of the translated sentences.

Ong et al. [28,29] reviewed the evolution of sign language detection using many aspects. It began with simply detecting simple signs like letters by observing hand movement. The authors attended to reveal the importance of no manual signals (NMS) representing head movement, facial expressions, etc., because some words are expressed through hands and NMS combined. Also, they mentioned the importance of dynamic recognition of the signs.

Munib et al. [30] introduced a system for recognizing finger-spelling for American Sign Language (ASL). The dataset was composed of 5254 samples of 10 ASL finger-spelling alphabets. The Authors usedk-NearestNeighbor's (KNN) Classifier based on dimensional features and applied Principal Component Analysis (PCA) to reduce data dimensions. The results demonstrated that the proposed system achieved an accuracy of 99.8%, which was decreased when applied PCA with (KNN) due to a high number of features correlated for alphabet ASL, which led to PCA being unable to separate data.

Rao et al. [31,32] developed a novel approach to translating Indian sign language extracted from selfie videos. They developed a real-time application to extract images from videos and recognize the sign using minimum Distance (M.D.) and Artificial Neural Network (ANN) classifiers. The system scored 85.58% for MD and 90% for ANN. The dataset contained 18 signs from 10 different signers.

2.2 Sensor-Based Approaches

Elon et al. [33], offered a system for ArSL recognition using Leap Motion Controller (LMC). The dataset contains 50 signs. The Authors used Multilayer Perceptron Neural Network (MLP) for classification based on two features, which are finger position distances and finger positions. The result demonstrated that the proposed system obtained an accuracy reaches 88%.

Another study used the Leap Motion Controller (LMC) by Elon et al. [33] for developing an ArSL recognition system. The suggested system comprises three phases: the pre-processing phase, the feature extraction phase, and a classification phase. In the classification phase, The Authors used Multilayer Perceptron (MLP) neural networks and the Naive Bayes classifier based on 12 features. They used a dataset containing 28 Arabic alphabets. The proposed system achieved an accuracy of 98% using the Nave Bayes classifier compared to Multilayer Perceptron (MLP), which achieved 99%.

Tubaiz et al. [34] presented a system for continuous ArSL recognition based on two DG5-VHand data gloves to capture hand movement. The dataset includes 40 sentences. The Authors used the window-based statistical approach to extract features. In addition, the Modified K-Nearest Neighbor (MKNN) approach was used for the classification process. The results appeared that the suggested framework obtained an accuracy of 98.9%.

Using two Leap Motion Controllers (LMCs), Mohandes et al. [35] developed a new method for Arabic Sign Language Recognition (ArSLR). The introduced system consists of three phases. These phases are pre-processing, feature extraction, and classification. In the classification phase, the Authors used Linear Discriminant Analysis (LDA) classifier and Dumpster-Shafer (D.S.) theory. They used 12 features selected from 23 features. The data sets include 28 Arabic characters, 10 samples for each character. The proposed system achieved an accuracy of 97.7% using the LDA classifier compared to the D.S. theory, which achieved 97.1%.

Aliyu et al. [36] suggested an ArSL recognition system using the Microsoft Kinect (M.K.) system. The data set consists of 20 Arabic signs, 10 samples of each sign collected by a native deaf signer. The

authors used Linear Discriminant Analysis (LDA) to extract features and classification. The proposed system obtained an accuracy of 99.8%.

Another investigation used the Microsoft Kinect (M.K.) by Hamed et al. [37] to develop an ArSL alphabet recognition system in complex backgrounds. The suggested system comprises three phases: the signer segmentation process, hand segmentation, and feature extraction. In the features extraction phase, the Authors used the Histogram of Oriented Gradients (HOG) and applied Principal Component Analysis (PCA) on HOG. They used the support vector machine (SVM) based on HOG-PCA for the classification phase. The data set contains 30 Arabic alphabets. The result showed that the introduced system achieved an accuracy of 99.2%.

Khelil et al. [38] introduced a novel approach for hand gesture recognition using a Leap Motion Controller (LMC) to obtain the number of fingers, fingertips, hand sphere radius, hand position, and normal. The data set consists of 10 different gestures, and each gesture is repeated 10 times. The authors used 16 features and a support vector machine (SVM) classifier algorithm for classification. The suggested system obtained an accuracy of 91%.

To help interested learners who wish to learn sign language, Fasihuddin et al. [39] introduced an intelligent tutoring system for ArSL using leap motion technology. The suggested system consists of five steps: pre-processing, tracking, feature extracting, classification, and sign recognition. The datasets used in this system depend on a learner level and generally include Arabic alphabetic, numbers, and some words. The authors used 12 features and a K-Nearest Neighbor (KNN) algorithm to classify the signs. The results demonstrated satisfactory user acceptance and willingness to use the system.

Hassan et al. [40] and Sidig et al. [41], comprehensively compared two different recognition approaches for the continuous ArSLR: the modified K-nearest neighbor and Hidden Markov Models (HMMs). The authors used two datasets consisting of 40 Arabic sentences, each repeated with 10 iterations. The datasets were collected using DG5-VHand data gloves, two Polhemus G4 motion trackers, and a camera. Additionally, the author has used window-based statistical features and Two-Dimensional Discrete Cosine Transform of an Image (2D DCT). They used three classifications: MKNN, Robust Alignment and Illumination by Sparse Representation (RASR), and Georgia Tech Gesture Toolkit (GT2K). The modified KNN has achieved the best wholesale recognition rates for all data sets that exceed the two HMM toolkits.

Sun et al. [42] suggested a novel approach for American Sign Language recognition in videos containing signs using Microsoft Kinect (M.K.) sensor. The dataset consists of 2000 phrases of 73 American Sign Language signs. The Authors used a Latent Support Vector Machine (SVM) classifier based on Kinect features and HOG features in the classification process. The results appeared that the proposed system realized an accuracy of 86%.

2.3 Deep Learning-Based Approaches

Altaf [43] and Tolba et al. [44] introduced a new method for Arabic sign language recognition using Pulse Coupled Neural Network (PCNN). The proposed system is divided into four steps. The purpose of the first step is to decrease the random noise (image smoothing). After that, the smoothed image is exposed to the number of iterations, where the output is an image signature that distinguishes the image contents. The extraction and selection of features were accomplished using Discrete Fourier Transform (DFT), which helped reduce dimensionality. The authors used Multilayer Perceptron (MLP) neural network for the classification process. In a signature generation, this model adds the continuity factor as a weight of the current pulse, representing how the surrounding pixels are fired in the same iteration. The data set was composed of 28 Arabic Sign Language alphabets posture, each represented by 8

samples. The proposed system achieved an accuracy of 93%. The drawback of PCNN is that many iterations lead to an increased background effect. Also, the proposed system encountered difficulties identifying certain postures because the single-hand show did not distinguish between two distinct positions.

Tolba et al. [44] presented a system for Arabic continuous gestures reorganization using the graph-matching approach. In the proposed system, the alphabet words are represented as graph models. The authors used a dataset containing 30 sentences of 100 words. They applied a pulse-coupled neural network (PCNN) approach to the generation of features and a Multilayer Perceptron (MLP) neural network to the classification process. The suggested system achieved an accuracy of 80%. The drawback of the suggested system is insensitive to the signer position, view angle, and background effects.

ElBadawy et al. [45] presented a system for Arabic sign language recognition using depth and intensity channels based on 3D Conventional Neural networks (CNN). The authors used a data set composed of 25 Arabic sign words collected from a unified Arabic sign dictionary. Each word has eight samples, thus containing 200 samples divided into 125 samples for training and 75 samples for testing. The proposed system used the word's video stream and obtained a normalized depth input, extracting spatial-temporal features from an input. The SoftMax layer in Three-Dimensional Deep Convolutional Neural Network (3D CNN) is used to classify features. The result showed that the proposed system achieved an average accuracy of 90%. The drawbacks of the proposed system are that less depth causes lower accuracy, the large depth leads to the overfitting problem, and misclassification occurs because the produced features may include unnecessary features. Also, the proposed system does not recognize any change in the testing words.

Munib et al. [30] developed a way to recognize static gestures to detect American Sign Language without needing worn devices or marking signs. They used Hough transform and neural networks to implement the system. The neural network consisted of 200 inputs and 214 outputs. Hough transform does not affect by image noise. The dataset size was 300 static hand signs; they used 200 for training and 100 for testing. The system achieved an accuracy of 92.3%. Yang et al. [46] proposed a new way to recognize Chinese Sign Language (CSL) by extracting images from videos of the upper body part, focusing on the hands. They used a convolutional neural network (CNN) which can score higher accuracy than artificial neural networks. The accuracy reached 99% without needing motion capture devices, gloves, or sensors. The dataset contains 40 videos of daily vocabulary for CSL. Summarization of the literature of this section is shown in Table 1.

The research by Amin et al. [47] explores the use of recurrent neural networks (RNNs) in classifying public discourse related to COVID-19 symptoms on Twitter. The authors adapt RNNs to capture the Twitter data's sequential nature and the tweets' context. This approach contributes to social media analysis and the study of public discourse related to current global issues. The study results indicate that RNNs can effectively classify the public discourse on COVID-19 symptoms present in Twitter content. This research highlights the importance of natural language processing techniques in understanding and analyzing public discourse and supports the growing literature on applying deep learning techniques in this field. This study's findings can inform public health decision-making by providing insights into the public perception and understanding of COVID-19 symptoms.

Table 1: Deep learning-based approaches-arabic sign language recognition models

Author	Approach	Dataset size	Accuracy	Drawbacks
Tolba et al. [48]	PCNN	28 alphabets, 8 samples for each one	93%	- Many iterations lead to increased background effect - The single show of the hand did not distinguish between two distinct positions.
Tolba et al. [44]	PCNN & graph matching	30 sentences of a total of 100 words	80%	Insensitive to signer position, view angle, and background effects.
Elbadawy et al. [45]	3D CNN	25 words, 8 samples for each one	90%	- The less depth causes lower accuracy - The large depth leads to the overfitting problem - not recognize any change in the testing words
Sharma et al. [15]	ANN	300 Static hand signs	92.3%	- The image noise did not affect
Hisham et al. [31,32]	CNN	40 videos of daily vocabulary	99%	- The target Chinese Sign Language (CSL)

In summary, we examine three approaches focusing on deep learning. For the sake of fact, we have to mention that each approach's main advantages and disadvantages are illustrated in Table 2.

Table 2: Approaches pros and cons

Approach	Main advantages (pros)	Main disadvantage (cons)
Image-based approaches	There is no need for a huge dataset; only limited Arabic sign images are needed.	-The accuracy is low compared to other approaches -The need for pre-processing and background settings of the image. -Slow processing.

(Continued)

Table 2: Continued

Approach	Main advantages (pros)	Main disadvantage (cons)
Sensor-based approaches	-Real-time image feature capturing and tracking. -Complex background processing	-Need special hardware for motion tracking. In some circumstances, the leap motion technology controller has a high error rate.
Deep-learning approaches	-No pre-processing is needed for the image. -Accurate and firm learning. -More than one learner could be combined to get more accuracy.	- A huge dataset is needed. -Learning is slow, especially when more than one learner is combined, or many hidden layers are used.

3 Arabic Sign Language Translation Model Architecture

3.1 Arabic Sign Language Dataset

The dataset for this research paper is composed form two datasets. The first was taken from the ArSL2018, Latif et al. [49] dataset, and the other from the Sign Language Digits Dataset built by Mavi [50]. To achieve better results, the dataset size was chosen to be large. ArSL2018 was collected by Latif et al. [49], and the images were taken from 40 participants of different ages in Prince Mohammad Bin Fahd University and the Khobar Area, Kingdom of Saudi Arabia. The dataset contains 54,049 images for each of the 32 Arabic sign language letters and poses. Images were collected from 40 participants differing in age.

Different images differed according to light, angles, timings, and backgrounds. Images were taken in Red, Green and Blue (RGB) (colored images) mode and were different in size, so a pre-processing step was made to make them easy to read for classification purposes. They were resized to have 64×64 and converted to grayscale mode with a pixel range of 0–255. The dataset was fully labeled according to each letter. The table below shows the dataset for each letter. Full dataset details are shown in Table 3.

Table 3: Arabic sign language recognition dataset [49]

#	Letter name in English script	Letter name in Arabic script	# of images	#	Letter name in English script	Letter name in Arabic script	# of images
1	Alif	أ(أَلِف)	1672	17	Zā	ظ(ظَاء)	1723
2	Bā	ب (بَاء)	1791	18	Ayn	ع(عَين)	2114
3	Tā	ت (أَتَاء)	1838	19	Ghayn	غ(غَين)	1977
4	Thā	ث (ثَاء)	1766	20	Fā	ف(فَاء)	1955
5	Jīm	ج (جِيمْ)	1552	21	Qāf	ق (قَاف)	1705
6	Hā	ح (حَاء)	1526	22	Kāf	ك(كَاف)	1774
7	Khā	خ (خَاء)	1607	23	Lām	ل(لأمْ)	1832
8	Dāl	د (دَالْ)	1634	24	Mīm	م(مِيمْ)	1765

(Continued)

Table 3: Continued

#	Letter name in English script	Letter name in Arabic script	# of images	#	Letter name in English script	Letter name in Arabic script	# of images
9	Dhāl	ذ (ذَال)	1582	25	Nūn	ن(نُون)	1819
10	Rā	ر (رَاء)	1659	26	Hā	ه(هَاء)	1592
11	Zāy	ز (زَاي)	1374	27	Wāw	و(وَاو)	1371
12	Sīn	س (سِينْ)	1638	28	Yā	ى (يَا)	1722
13	Shīn	ش (شِينْ)	1507	29	Tāa	ة(ة)	1791
14	Sād	ص(صَادْ)	1895	30	Al	ال(ال)	1343
15	Dād	ض(ضَاد)	1670	31	Laa	لا(لا)	1746
16	Tā	ط(طَاء)	1816	32	Yāa	ياء (يَاء)	1293

Students collected the second dataset from Turkey Ankara Ayrancı Anadolu High School [50]. The images were taken from 118 students, with 10 samples for each student. The dataset contains 2062 images for the 10 Arabic sign language numbers (from 0 to 9) with different poses for each number. Images were taken in RGB mode (colored images), and the size of each image is 100×100. We labeled the dataset according to the ArSL2018 labeling procedure. We divided the dataset into two parts, 90% for training and 10% for validation. The default k-fold cross-validation used was $k = 10$. We used outlet data taken from live video streams for the testing phase.

3.2 General Architecture

We propose a multitask model for translating Arabic sign language into Arabic text. A camera stream input is pre-processed to obtain S.L. gestures. The classification module predicts the corresponding letter for each input gesture. Most of the generated predictions are used to elect the most corresponding letter. Finally, the produced Arabic text is converted to speech, as shown in Fig. 1.

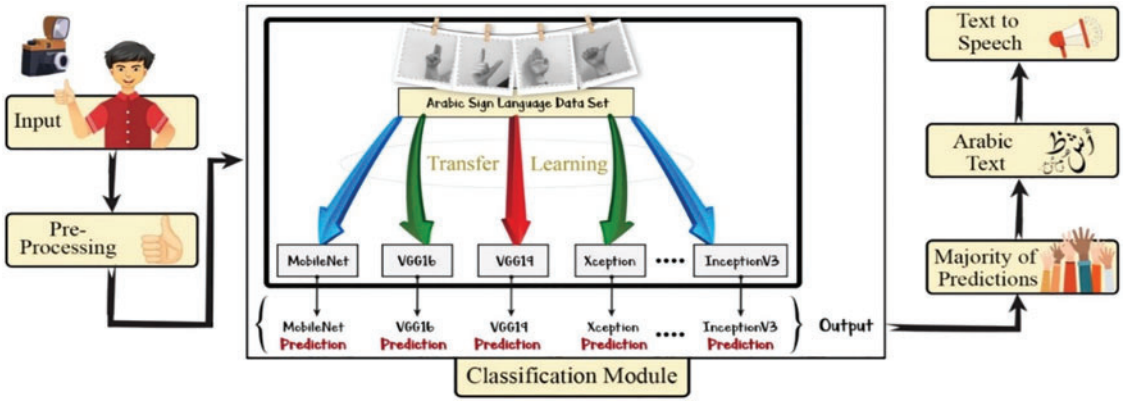

Figure 1: ArSL translation model architecture

3.3 Hand Edge Detection

Recognizing an input sign language gesture requires the detection of hand edges in a camera video stream. We develop a hand edge detection technique based on mathematical morphology theory and

human skin color detection. Mathematical morphology techniques were used to detect the geometrical structure of the hand based on procreated hand shape topologies. To determine image regions of skin color, RGB mode (colored images), which are taken from the camera video stream, are converted into Hue, Saturation and value (HSV) color space because it is more related to human color perception. Fig. 2 represents the hand detection scheme.

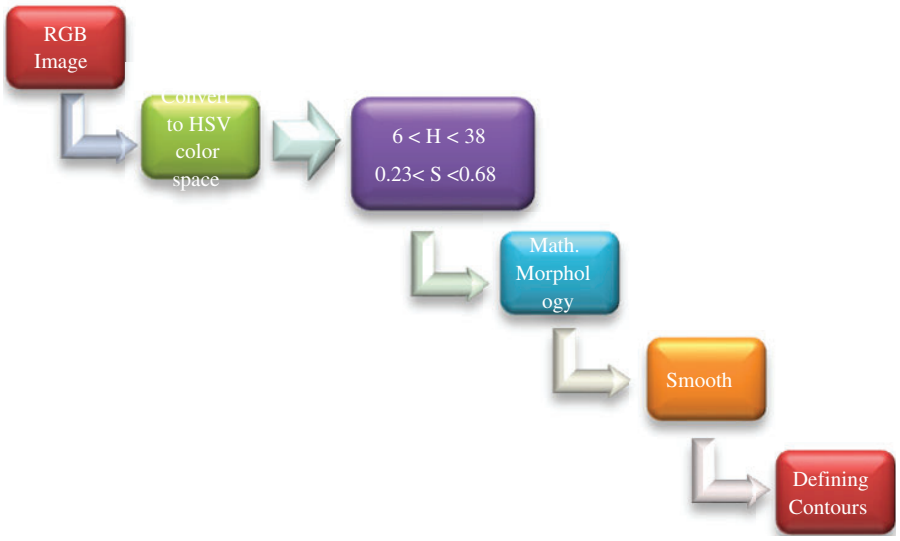

Figure 2: Hand edge detection scheme

3.4 The Classification Module

The adopted classification module relies on utilizing 12 deep-learning classification models. Each model was designed depending on convolutional neural networks (CNNs) that can achieve high accuracy. The classification process depends on many neural layers that can assign weights and biases for images to distinguish between them. as shown in Fig. 3.

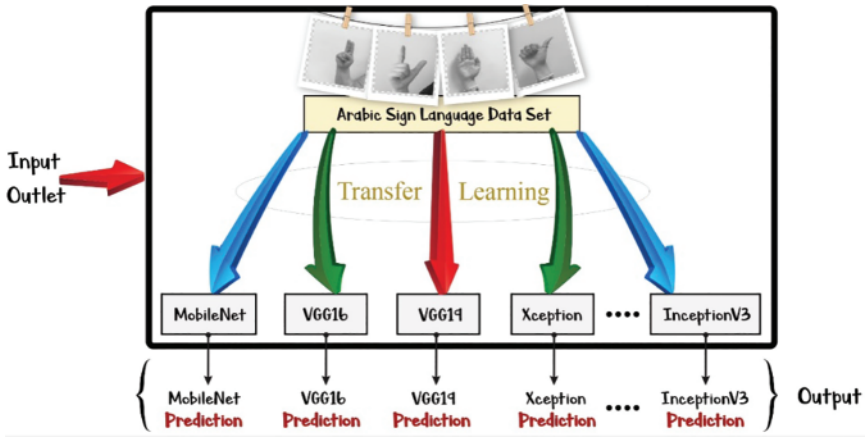

Figure 3: Classification module

The main idea in Fig. 3 is to emphasize that the recognition is with the minimum or no error. The idea is to use multiple famous trained nets and reuse them in our problem based on Transfer Learning. Then, making majority voting guarantees the accuracy of the recognition, which avoids the high error rate for a single learner. The features one learner does not cover could be covered by another, enhancing recognition accuracy. The pseudo-code in Algorithm 1 represents Arabic Sign Language (ArSL) recognition based on deep learning modules.

Algorithm 1: ArSL recognition pseudo code

Algorithm Arsl *(Video Stream)*
{ *This algorithm is to convert Arabic sign sentence to coressponiding Arabic Speech*}
Repeat
1: Set Arabic_sentence=""
{ *This is an empty string for concatination of letters and words* }
2: Repeat
Capture the arabic sign image from the video stream
Preprocces Image
Send the captured image to the trained modules
Store predictions of modules
Final prediction=Majority(Modules Predictions)
Assert predicted letter or word at end of Arabic_Sentence string
Untile sentence end_sign predicted
3: call Text_to_Speech(Arabic_Sentence)
Untile end of conversation

The models were retrained using Arabic Sign Language Dataset; every model had four epochs of training with 1635 iterations. After working, each model will provide the module with a prediction for every input. To predict the final output, transfer learning is employed to extract knowledge from 12 models to get a prediction from each model and to select the most accurate prediction.

Despite its simplicity, image-based approaches suffer from errors due to many factors such as lighting conditions, image background, face, hands segmentation, and different types of noise [10]. Our adopted model will determine the best prediction by electing the majority's voice based on the highest number of predictions extracted from the 12 retrained models to overcome these obstacles, increase the accuracy, and overcome those weaknesses. As the number of predictions for the same sign increases, the probability of having the right prediction increases, and the error will decrease. Every model has its strengths and drawbacks, which vary according to the architecture and accuracy. The details are below.

MobileNet is one of the models developed by Google Inc., which focuses on optimizing latency time with high accuracy and small networks. MobileNet model depends on depthwise separable convolutions with less computational complexity due to factorized convolutions by Howard et al. [51]. MobileNet uses 3×3 depthwise detachable, which makes the computational more timeless than the standard convolutions about 8 to 9 times. The accuracy of MobileNet is 0.91.

VGG16 and VGG19 are convolutional neural network models for evaluating depth-increasing networks using architecture. This model was considered the preferred choice for image features extraction; however, VGG16 consists of 138 million parameters, whereas VGG19 consists of 143 million parameters which can be challenging to handle. The accuracy for VGG16 is 0.901, and for

VGG19 is 0.9. VGGNet has two drawbacks: they are slow to train, and the weights of the network architecture are quite large (disk/bandwidth), by Simonyan et al. [52].

ResNet50 is a short name for a residual network and is another model to ease the training of networks. The model is initialized by the ImageNet classification models and then fine-tuned on the object detection data. This model consists of 25 million parameters, although much deeper than VGG16 and VGG19. Due to the use of global average pooling rather than fully connected layers, the model size is small and reaches approximately 98 M.B. Due to the increased depth, higher accuracy could easily be achieved, 0.921, by He et al. [53].

The InceptionV3 model comprises a fundamental unit called an "Inception cell." This cell performs a series of convolutions at multiple scales and combines the outputs for a complete result. To save computation, 1x1 convolutions are used to reduce the input channel depth. Each cell has a set of 1×1, 3 3, and 5 5 filters to learn to extract features at different scales from the input. Max pooling is also used, albeit with "sam" padding, to preserve the dimensions so that the output can be appropriately concatenated, Szegedy et al. [54]. Its accuracy is 0.937.

The architecture of Xception is a linear stack of wise depth separable layers of convolution with residual connections. This model has 36 convolution layers that form the network's base for extracting features. Except for the first and last modules, the 36 convolution layers are structured into 14 modules, all of which have linear residual connections around them. Xception shows small performance gains in the ImageNet dataset classification and large gains in the JFT dataset compared to Inception V3, Chollet [55]. It can provide an accuracy of 0.945.

Inception-ResNet-v2 is a convolutional neural network trained on more than a million images from the ImageNet database. This model combined Inception architectures with residual connections, was consisted of 143 million parameters, unlike inceptionV3 and ResNet50, which were composed of 23 and 25 million, respectively. Additionally, the size of this model is greater than inceptionV3 and ResNet50, which reaches 215 MB compared to 92MB and 98 M.B. for inceptionV3 and ResNet50, respectively, which led to higher accuracy of 0.953, Szegedy et al. [56].

DenseNet connects each layer to every other layer in a feed-forward fashion. It may be useful to reference feature maps from earlier in the network. Thus, each layer's feature map is concatenated to the input of every successive layer within a dense block. This allows later layers within the network to directly leverage the features from previous layers, encouraging feature reuse. For each layer, the feature maps of all preceding layers are used as inputs, and their feature maps are used as inputs into all subsequent layers. This model has four advantages: to alleviate the vanishing gradient problem, reinforce the propagation of features, encourage the reuse of features, and significantly reduce the number of parameters. DenseNets improved performance with less complexity than ResNet, Huang et al. [57]. The accuracy of DenseNet121, DenseNet169, and DenseNet201 are 0.923, 0.932, and 0.936, respectively.

The neural architecture search (NAS) framework uses a reinforcement learning search method to optimize the architecture configurations. This model is made up of convolutional cells that are repeated several times, where each convolutional cell has the same architecture but different weights. According to Zoph et al. [58], these convolutional cells are a normal cell that returns the feature map of the same dimension and a reduction cell that returns the feature map with the feature map height and width reduced by a factor of two. The 2 used models are NASNetLarge and NASNetMobile, with an accuracy of 0.960 and 0.919, respectively. The accuracy of retrained prediction rates is illustrated in Table 4.

Table 4: The highest number of predictions extracted from the 12 retrained models- using the Arabic sign language dataset

Model #	Model name	Training accuracy
1	VGG16	0.9997
2	VGG19	0.9985
3	ResNet50	0.9621
4	InceptionV3	0.9569
5	Xception	0.9679
6	InceptionResNetV2	0.9532
7	MobileNet	0.9858
8	DenseNet121	0.9920
9	DenseNet169	0.9897
10	DenseNet201	0.9930
11	NASNetLarge	0.9901
12	NASNetMobile	0.9933

3.5 Translation Stages

3.5.1 Stage (1): Input and Pre-Processing

The input will be acquired from a camera as a stream, divided into frames, each consisting of 14 pictures. Those pictures will enter the pre-processing procedure, and some will be discarded after the analysis to filter the input based on valid sign language gestures. Each valid picture contains the entire view with the background. It needs to be analyzed to detect the hand contour.

3.5.2 Stage (2): Classification

When the pre-processed input enters the classification module, each retrained model will generate a prediction that may differ from one to another. There will be 12 predictions.

3.5.3 Stage (3): Majority of Predictions

After obtaining the 12 predictions, the classification module, to achieve higher accuracy, will use the majority of predictions. Simple majority voting is a collective technique that finds the prediction depending on the majority of the classifiers to improve the accuracy of the final output [59], as shown in Fig. 4.

3.5.4 Stage (4): Output

The output of the majority voting, which is the corresponding letter written in English characters, will be used to produce the equivalent letter in an Arabic Text format. This text will be converted to speech so the user can hear the output as vocal signals.

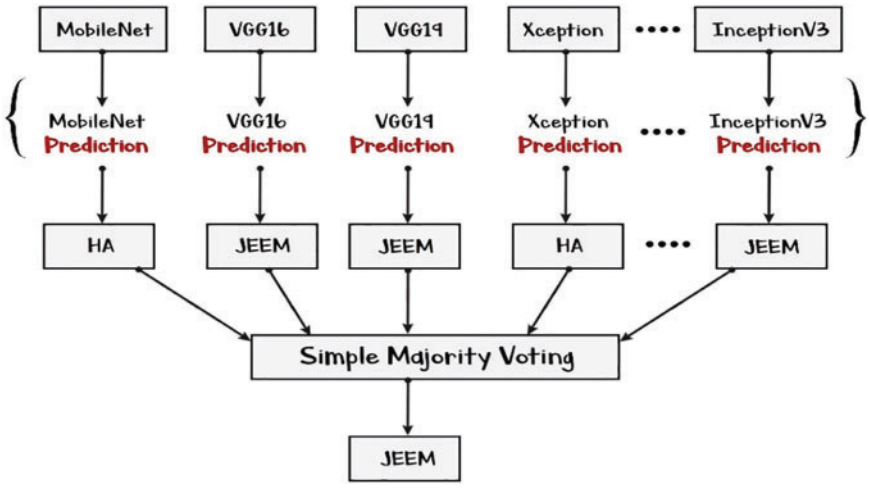

Figure 4: Majority of predictions

4 Result and Evaluation

4.1 Testing

Live examples are used in the testing process using new data. We used 100 outlet pictures taken from a camera video stream. After pre-processing, they are entered into the classification module, and 12 predictions are generated. Those outlet pictures are used to test each model individually. On the other hand, most predictions were used to measure the overall accuracy. Fig. 5 represents a sample snapshot.

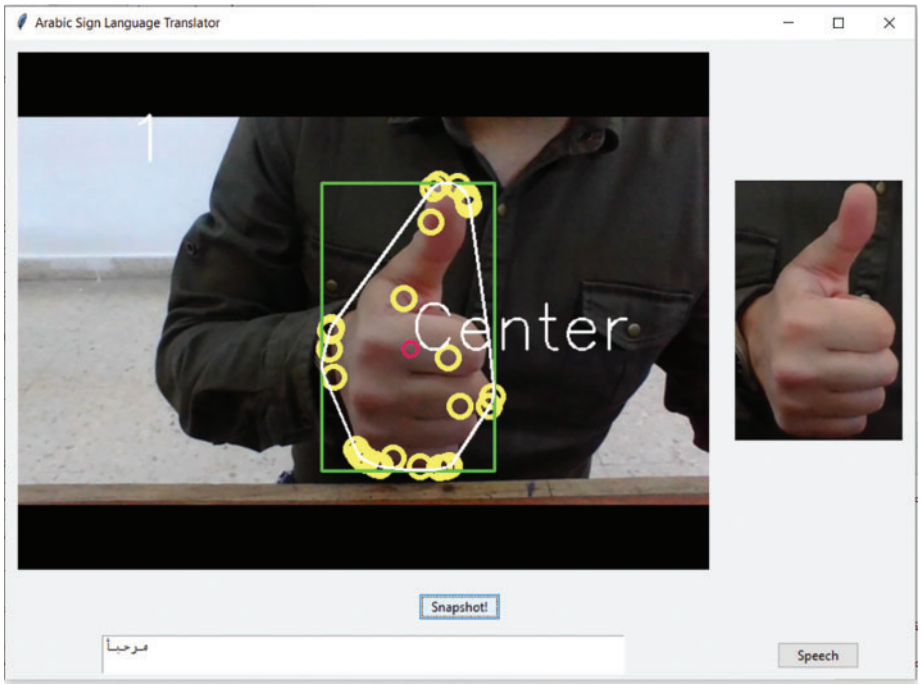

Figure 5: Sample result snapshot

4.2 Accuracy Measurements

The measurements for the tested data will focus on calculating each model's and the majority's accuracy. The used measurements are the Recall, the Precision, and the F-measure. The measure we will focus on is the F-measure. Table 5 illustrates the best deep learning accuracy from the literature compared to our approach.

Table 5: A simple accuracy comparison of our adopted model and the state of the art of approaches for translating Arabic sign language into Arabic text

Author	Approach	Accuracy
Hamed et al. [37]	PCNN	93%
Rao et al. [31,32]	PCNN & graph matching	80%
Tubaiz et al. [34]	3D CNN	90%
Elbadawy et al. [45]	C.N. with multi-dimensional layers	89.65%
Yang et al. [46]	A survey of different models	The nearest model to ours was 89.33%
Tolba et al. [48]	Semantic boundary detection (SBD) with reinforcement learning (R.L.), SBD-RL agent	The accuracy by word error rate is 26.6%, real Accuracy percentage is 74%.
Our adopted model	Transfer learning CNN with majority selection	93.7%

The confusion matrix in Table 6 shows the meaning of the combinations between actual and predicted classes. The used measurements (Precision, Recall, and F1-Measure) depend on that combination.

Table 6: A combination of actual and predicted classes upon recognition

		Predicted class	
		Negative	Positive
Actual class	Negative	True negative (T.N.)	False positive (F.P.)
	Positive	False negative (F.N.)	True positive (T.P.)

The Precision value is used to measure the preciseness of the model by finding how many from the predicted positive are positive, as shown in Eq. (1).

$$\text{Precsion} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{1}$$

The recall is used in Eq. (2) to show how many actual positive classifications were labeled as positive by the model.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{2}$$

F-measure combines the two measurements to get more powerful and precise results about the model's accuracy. The formula is shown in Eq. (3).

$$F - Measure = 2 * \frac{Precsion * Recall}{Prescision + Recall} \tag{3}$$

Using multiple Deep learners and selecting most of their predictions contributes to achieving more accurate results in Arabic sign language translation. Table 7 shows each learner's Recall, Precision, and F-measure values. Comparing Tables 5 and 7, the discrepancies between training and prediction accuracy could be due to overfitting (not severe) some of the deep learning models, which could be investigated in a future job.

Table 7: Summarizes the prediction accuracies

Model name	Recall	Precision	F-measure
VGG16	93.6%	91.7%	92.6%
VGG19	93.2%	91.4%	92.1%
ResNet50	91.3%	89.9%	90.6%
InceptionV3	90.2%	89.3%	89.6%
Xception	92.1%	91.2%	91.5%
InceptionResNetV2	90%	89.1%	89.3%
MobileNet	92.7%	90.9%	91.9%
DenseNet121	92.9%	91.9%	92%
DenseNet169	92.9%	91.1%	91.9%
DenseNet201	93.1%	91.2%	92.2%
NASNetLarge	92.8%	91.8%	92.1%
NASNetMobile	93.1%	91.3%	92%
Majority	94.4%	93.4%	93.7%

5 Conclusion

In conclusion, this research paper presents a novel and significant contribution to sign language translation. Our approach to translating Arabic sign language into written and spoken Arabic in real-time utilizes deep learning techniques, specifically transfer learning, to achieve high accuracy in translation. Our results demonstrate the effectiveness of our approach, as we achieved an average training accuracy of 98.2% and a testing accuracy of 93.7%. This is a noteworthy improvement compared to other related works in the same area, making our proposed solution a promising and valuable contribution to the field. Our approach offers a practical solution for improving communication between individuals with hearing disabilities and the general population, and it has the potential to significantly enhance the quality of life for people with special needs. The future direction for this research involves exploring other advanced deep-learning techniques to improve the model's accuracy and real-world applications for people with special needs. Additionally, evaluating the scalability and robustness of the model for other sign languages can also be considered for future research.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare they have no conflicts of interest to report regarding the present study.

References

[1] Y. Shao, J. C. -W. Lin, G. Srivastava, A. Jolfaei, D. Guo *et al.,* "Self-attention-based conditional random fields latent variables model for sequence labeling," *Pattern Recognition Letters*, vol. 145, pp. 157–164, 2021.

[2] J. C. -W. Lin, Y. Shao, Y. Djenouri and U. Yun, "Asrnn: A recurrent neural network with an attention model for sequence labeling," *Knowledge-Based Systems*, vol. 212, pp. 106548, 2021.

[3] A. Tebbifakhr, M. Negri and M. Turchi, "Automatic translation for multiple nlp tasks: A multi-task approach to machine-oriented nmt adaptation," in *Proc. of the 22nd Annual Conf. of the European Association for Machine Translation*, Lisboa, Portugal, pp. 235–244, 2020.

[4] J. Zakraoui, M. Saleh, S. Al-Maadeed and J. M. AlJa'am, "Evaluation of Arabic to English machine translation systems," in *11th Int. Conf. on Information and Communication Systems (ICICS)*, Irbid, Jordan, pp. 185–190, 2020.

[5] N. El-Bendary, H. M. Zawbaa, M. S. Daoud, A. E. Hassanien and K. Nakamatsu, "Arslat: Arabic sign language alphabets translator," in *Int. Conf. on Computer Information Systems and Industrial Management Applications (CISIM)*, Krakow, Poland, pp. 590–595, 2010.

[6] k. M. O. N. Firas ibrahim and M. A. A. AL-shannaq, "Gender identification and age estimation of arabic speaker using machine learning," *Journal of Theoretical and Applied Information Technology*, vol. 98, no. 19, pp. 3242–3251, 2020.

[7] M. Mohandes, S. Aliyu and M. Deriche, "Arabic sign language recognition using the leap motion controller," in *IEEE 23rd Int. Symp. on Industrial Electronics (ISIE)*, Istanbul, Turkey, pp. 960–965, 2014.

[8] M. Al-Ayyoub, A. Nuseir, K. Alsmearat, Y. Jararweh and B. Gupta, "Deep learning for arabic nlp: A survey," *Journal of Computational Science*, vol. 26, pp. 522–531, 2018.

[9] M. Tolba, A. Samir and M. Abul-Ela, "A proposed graph matching technique for arabic sign language continuous sentences recognition," in *8th Int. Conf. on Informatics and Systems (INFOS)*, Giza, Egypt, pp. MM-14-MM-20, 2012.

[10] M. Mohandes, M. Deriche and J. Liu, "Image-based and sensor-based approaches to arabic sign language recognition," *IEEE Transactions on Human-Machine Systems*, vol. 44, no. 4, pp. 551–557, 2014.

[11] M. Al-Smadi, O. Qawasmeh, M. Al-Ayyoub, Y. Jararweh and B. Gupta, "Deep recurrent neural network vs. support vector machine for aspect-based sentiment analysis of arabic hotels' reviews," *Journal of Computational Science*, vol. 27, pp. 386–393, 2018.

[12] L. Dipietro, A. M. Sabatini and P. Dario, "A survey of glove-based systems and their applications," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews*, vol. 38, no. 4, pp. 461–482, 2008.

[13] S. Nadgeri and D. Kumar, "Survey of sign language recognition system," Available at SSRN 3262581, 2018.

[14] X. Lv, H. Hou, X. You, X. Zhang and J. Han, "Distant supervised relation extraction via disan-2cnn on a feature level," *International Journal on Semantic Web and Information Systems (IJSWIS)*, vol. 16, no. 2, pp. 1–17, 2020.

[15] Y. Sharma, R. Bhargava and B. V. Tadikonda, "Named entity recognition for code mixed social media sentences," *International Journal of Software Science and Computational Intelligence (IJSSCI)*, vol. 13, no. 2, pp. 23–36, 2021.

[16] S. R. Sahoo and B. B. Gupta, "Classification of various attacks and their defence mechanism in online social networks: A survey," *Enterprise Information Systems*, vol. 13, no. 6, pp. 832–864, 2019.

[17] K. Nahar, O. Al-Hazaimeh, A. Abu-Ein and N. Gharaibeh, "Phonocardiogram classification based on machine learning with multiple sound features," *Journal of Computer Science*, vol. 16, no. 11, pp. 1648–1656, 2020.

[18] M. H. Bhatti, J. Khan, M. U. G. Khan, R. Iqbal, M. Aloqaily *et al.,* "Soft computing-based eeg classification by optimal feature selection and neural networks," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 10, pp. 5747–5754, 2019.
[19] K. M. Nahar, M. A. Ottom, F. Alshibli and M. M. A. Shquier, "Air quality index using machine learning–a Jordan case study," *Compusoft*, vol. 9, no. 9, pp. 3831–3840, 2020.
[20] S. Yen, M. Moh and T. -S. Moh, "Detecting compromised social network accounts using deep learning for behavior and text analyses," *International Journal of Cloud Applications and Computing (IJCAC)*, vol. 11, no. 2, pp. 97–109, 2021.
[21] N. S. Khan, A. Abid and K. Abid, "A novel natural language processing (nlp)–based machine translation model for English to Pakistan sign language translation," *Cognitive Computation*, vol. 12, pp. 748–765, 2020.
[22] K. Assaleh, T. Shanableh, M. Fanaswala, F. Amin and H. Bajaj, "Continuous arabic sign language recognition in user dependent mode," *Journal of Intelligent Learning Systems and Applications*, vol. 2, no. 1, pp. 19–27, 2010.
[23] A. Tharwat, T. Gaber, A. E. Hassanien, M. K. Shahin and B. Refaat, "Sift-based arabic sign language recognition system," in *First Int. Afro-European Conf. for Industrial Advancement AECIA 2014*, Addis Ababa, Ethiopia, pp. 359–370, 2015.
[24] H. Luqman and S. A. Mahmoud, "Transform-based arabic sign language recognition," *Procedia Computer Science*, vol. 117, pp. 2–9, 2017.
[25] M. Elpeltagy, M. Abdelwahab, M. E. Hussein, A. Shoukry, A. Shoala *et al.,* "Multi-modality-based arabic sign language recognition," *IET Computer Vision*, vol. 12, no. 7, pp. 1031–1039, 2018.
[26] N. Ibrahim, M. Selim and H. Zayed, "An automatic arabic sign language recognition system (arslrs)," *Journal of King Saud University-Computer and Information Sciences*, vol. 30, no. 4, pp. 470–477, 2017.
[27] H. Luqman and S. A. Mahmoud, "Automatic translation of arabic text-to-arabic sign language," *Universal Access in the Information Society*, vol. 18, no. 4, pp. 939–951, 2019.
[28] S. C. Ong and S. Ranganath, "Automatic sign language analysis: A survey and the future beyond lexical meaning," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 27, no. 6, pp. 873–891, 2005.
[29] R. Mohammed and S. Kadhem, "A review on arabic sign language translator systems," *Journal of Physics: Conference Series*, vol. 1818, no. 1, pp. 012033, 2021.
[30] Q. Munib, M. Habeeb, B. Takruri and H. A. Al-Malik, "American sign language (asl) recognition based on hough transform and neural networks," *Expert Systems with Applications*, vol. 32, no. 1, pp. 24–37, 2007.
[31] G. A. Rao and P. Kishore, "Selfie video based continuous Indian sign language recognition system," *Ain Shams Engineering Journal*, vol. 9, no. 4, pp. 1929–1939, 2018.
[32] B. Hisham and A. Hamouda, "Arabic sign language recognition using ada-boosting based on a leap motion controller," *International Journal of Information Technology*, vol. 13, no. 3, pp. 1221–1234, 2021.
[33] A. Elons, M. Ahmed, H. Shedid and M. Tolba, "Arabic sign language recognition using leap motion sensor," in *9th Int. Conf. on Computer Engineering & Systems (ICCES)*, Cairo, Egypt, pp. 368–373, 2014.
[34] N. Tubaiz, T. Shanableh and K. Assaleh, "Glove-based continuous arabic sign language recognition in user-dependent mode," *IEEE Transactions on Human-Machine Systems*, vol. 45, no. 4, pp. 526–533, 2015.
[35] M. Mohandes, S. Aliyu and M. Deriche, "Prototype arabic sign language recognition using multi-sensor data fusion of two leap motion controllers," in *IEEE 12th Int. Multi-Conf. on Systems, Signals & Devices (SSD15)*, Mahdia, Tunisia, pp. 1–6, 2015.
[36] S. Aliyu, M. Mohandes, M. Deriche and S. Badran, "Arabie sign language recognition using the microsoft kinect," in *13th Int. Multi-Conf. on Systems, Signals & Devices (SSD)*, Leipzig, Germany, pp. 301–306, 2016.
[37] A. Hamed, N. A. Belal and K. M. Mahar, "Arabic sign language alphabet recognition based on hog-pca using microsoft kinect in complex backgrounds," in *IEEE 6th Int. Conf. on Advanced Computing (IACC)*, Bhimavaram, India, pp. 451–458, 2016.

[38] B. Khelil, H. Amiri, T. Chen, F. Kammüller, I. Nemli *et al.,* "Hand gesture recognition using leap motion controller for recognition of arabic sign language," in *3rd Int. Conf. ACECS,* Hammamet, Tunisia, pp. 233–238, 2016.

[39] H. Fasihuddin, S. Alsolami, S. Alzahrani, R. Alasiri and A. Sahloli, "Smart tutoring system for arabic sign language using leap motion controller," in *Int. Conf. on Smart Computing and Electronic Enterprise (ICSCEE)*, Shah Alam, Malaysia, pp. 1–5, 2018.

[40] M. Hassan, K. Assaleh and T. Shanableh, "Multiple proposals for continuous arabic sign language recognition," *Sensing and Imaging*, vol. 20, no. 1, pp. 4, 2019.

[41] A. A. I. Sidig, H. Luqman, S. Mahmoud and M. Mohandes, "Karsl: Arabic sign language database," *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, vol. 20, no. 1, pp. 1–19, 2021.

[42] C. Sun, T. Zhang, B. -K. Bao and C. Xu, "Latent support vector machine for sign language recognition with kinect," in *IEEE Int. Conf. on Image Processing*, Melbourne, VIC, Australia, pp. 4190–4194, 2013.

[43] M. M. Altaf, "A hybrid deep learning model for breast cancer diagnosis basedon transfer learning and pulse-coupled neural networks," *Mathematical Biosciences and Engineering*, vol. 18, no. 5, pp. 5029–5046, 2021.

[44] M. F. Tolba, A. Samir and M. Aboul-Ela, "Arabic sign language continuous sentences recognition using pcnn and graph matching," *Neural Computing and Applications*, vol. 23, no. 3, pp. 999–1010, 2013.

[45] M. ElBadawy, A. Elons, H. A. Shedeed and M. Tolba, "Arabic sign language recognition with 3D convolutional neural networks," in *Eighth Int. Conf. on Intelligent Computing and Information Systems (ICICIS)*, Cairo, Egypt, pp. 66–71, 2017.

[46] S. Yang and Q. Zhu, "Video-based Chinese sign language recognition using convolutional neural network," in *IEEE 9th Int. Conf. on Communication Software and Networks (ICCSN)*, Guangzhou, China, 2017, pp. 929–934, IEEE.

[47] S. Amin, A. Alharbi, M. I. Uddin and H. Alyami, "Adapting recurrent neural networks for classifying public discourse on covid-19 symptoms in Twitter content," *Soft Computing*, vol. 26, no. 20, pp. 11077–11089, 2022.

[48] M. F. Tolba, M. Abdellwahab, M. Aboul-Ela and A. Samir, "Image signature improving by pcnn for arabic sign language recognition," *Canadian Journal on Artificial Intelligence, Machine Learning & Pattern Recognition*, vol. 1, no. 1, pp. 1–6, 2010.

[49] G. Latif, N. Mohammad, J. Alghazo, R. AlKhalaf and R. AlKhalaf, "Arasl: Arabic alphabets sign language dataset," *Data in Brief*, vol. 23, pp. 103777, 2019.

[50] A. Mavi, "A new dataset and proposed convolutional neural network architecture for classification of American sign language digits," arXiv Preprint arXiv:2011.08927, 2020.

[51] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang *et al.,* "Mobilenets: Efficient convolutional neural networks for mobile vision applications," arXiv Preprint arXiv:1704.04861, 2017.

[52] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv Preprint arXiv:1409.1556, 2014.

[53] K. He, X. Zhang, S. Ren and J. Sun, "Deep residual learning for image recognition," in *IEEE Conf. on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, pp. 770–778, 2016.

[54] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens and Z. Wojna, "Rethinking the inception architecture fr computer vision," in *IEEE Conf. on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, pp. 2818–2826, 2016.

[55] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *IEEE Conf. on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, pp. 1251–1258, 2017.

[56] C. Szegedy, S. Ioffe, V. Vanhoucke and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Thirty-First AAAI Conf. on Artificial Intelligence*, San Francisco, California USA, pp. 2278–4284, 2017.

[57] G. Huang, G. LHuang, Z. iu, L. Van Der Maaten and K. Q. Weinberger, "Densely connected convolutional networks," in *The IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Computer Vision Foundation*, Honolulu, HI, USA, pp. 4700–4708, 2017.

[58] B. Zoph, V. Vasudevan, J. Shlens and Q. V. Le, "Learning transferable architectures for scalable image recognition," in *IEEE Conf. on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, pp. 8697–8710, 2018.

[59] D. Anil Kumar and V. Ravi, "Predicting credit card customer churn in banks using data mining," *International Journal of Data Analysis Techniques and Strategies*, vol. 1, no. 1, pp. 4–28, 2008.