IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 1, 2021, accepted March 14, 2021, date of publication March 30, 2021, date of current version April 26, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3069714

Arabic Sign Language Recognition System Using 2D Hands and Body Skeleton Data

MOHAMED A. BENCHERIF[1,5], MOHAMMED ALGABRI[2,5], MOHAMED A. MEKHTICHE[1,5], MOHAMMED FAISAL[4,5], MANSOUR ALSULAIMAN[1,5], HASSAN MATHKOUR[2,5], MUNEER AL-HAMMADI[1,5], (Member, IEEE), AND HAMID GHALEB[3,5]

[1]Department of Computer Engineering, College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia.
[2]Department of Computer Science, College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia.
[3]Department of Software Engineering, College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia.
[4]College of Applied Computer Science, King Saud University, Riyadh 11543, Saudi Arabia.
[5]Center of Smart Robotics Research, College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia.

Corresponding author: Mohamed A. Bencherif (mbencherif1@yahoo.com)

This work was supported by the Deanship of Scientific Research and RSSU at King Saud University through research group no: RG-1437-018.

ABSTRACT This paper presents a novel Arabic Sign Language (ArSL) recognition system, using selected 2D hands and body key points from successive video frames. The system recognizes the recorded video signs, for both signer dependent and signer independent modes, using the concatenation of a 3D CNN skeleton network and a 2D point convolution network. To accomplish this, we built a new ArSL video-based sign database. We will present the detailed methodology of recording the new dataset, which comprises 80 static and dynamic signs that were repeated five times by 40 signers. The signs include Arabic alphabet, numbers, and some daily use signs. To facilitate building an online sign recognition system, we introduce the inverse efficiency score to find a sufficient optimal number of successive frames for the recognition decision, in order to cope with a near real-time automatic ArSL system, where tradeoff between accuracy and speed is crucial to avoid delayed sign classification. For the dependent mode, best results were obtained for dynamic signs with an accuracy of 98.39%, and 88.89% for the static signs, and for the independent mode, we obtained for the dynamic signs an accuracy of 96.69%, and 86.34% for the static signs. When both the static and dynamic signs were mixed and the system trained with all the signs, accuracies of 89.62% and 88.09% were obtained in the signer dependent and signer independent modes respectively.

INDEX TERMS Arabic sign language, OpenPose, skeleton, key points, parallel convolutions.

I. INTRODUCTION

Sign Language can be considered as the most preliminary means of human communication. If someone travels to any foreign country, ignoring their language completely, he can easily, and by instinct, find a way to make signs to at least drink water, eat food, and get a place to sleep. Throughout this context, deaf and hard of hearing (HoH) people and their surrounding families have developed, over decades, a set of hand gestures, facial expressions, and lip movements to communicate with each other. These signs vary between different countries and languages, though they have some similarities.

Nowadays, researchers have conducted many advanced studies in sign languages to help the deaf in their daily life. In addition, through the emergence of the video calls, the deaf society and HoH are no more clustered and pouched in their bulbs, but contribute actively in many domains.

Although, the task of recognizing signs in videos could be similar to recognizing actions and gestures it is actually more complex. The complexity comes from detecting singular gestural boundaries in long sequences. This full recognition process is dependent on what is recognized instantly, as a part of an independent sign, or as a boundary between two successive signs. The first point is the absence of a clear stopping sign. In theory, the conventional ''hands down'' can be used as a stop sign like a ''silent pause'' in speech. However, this pause might not come at every second, and the length of successive signs may undergo an un-deterministic duration. Moreover, some signers have their own way of stopping or pausing that is more related to the signer fatigue. Many sign experts

The associate editor coordinating the review of this manuscript and approving it for publication was Sudhakar Radhakrishnan.

TABLE 1. Arabic sign language datasets.

Ref	# Signers	# Signs	Collection	Devices	Modality	Gloss	Remark	Acc.	Year	Publicly Available	Country
[1]	3	23 x 50	Words	Camera	RGB	No	-	71%	2007	No	UAE
[2]	40	32	Alphabet	Camera Gray Images	Hands	No	54049 images One Hand	-	2018	Yes	Saudi Arabia
[3]	10	500 (Mixed signs)	Words Sentences	Canon Power Shot A490 Camera	RGB, + Face Emotion	No	Signs World Atlas	97% 95.28%	2014	No	Egypt
[4]	4	1216	Words Sentences	Leap Motion KinectV2, Digital Cam		No	4 Rec. angles		2015	-	Egypt
[5]	1	300		Camera		No	Gloves	93%	2007	-	-
[6]	2	20 x 10 repetitions	Words	Leap Motion +Camera	Face, Body	No		95%		-	-

may stop signing for a short instant, "time to recall the next sign", to recapitulate the idea in their heads and continue, but the pause is not unified between signers, and does not appear to be a pause if one is not aware of it in advance. It is a similar approach to pauses in speech, but the hands are still used, unlike the speech, where speech sound stops for a short instant (time to take a breath or even drink some water in long discourses). From our discussions with Arabic Sign language (ArSL) experts, signers do this approach to make pauses, and care more about the next idea in the sign discussion.

The second point is the inclusion of the facial expressions, including the lips and the eyebrows movement, which lead to additional vectors or modalities in the sign recognition process. Mouth closure is also used as a pause, but more research still needs to be done in this case. Nevertheless, many SL research papers simply ignore the face entirely, either it was not recorded, or recordings did not focus on the face.

In this paper, we propose a new approach for recognizing dynamic and static signs. Our approach is a two steps fold, the first step, we estimate the body joints including the finger positions from input sequence of frames, in the second step a point CNN model is used to differentiate between the sign classes.

In order to select the best number of successive frame images useful for a fast decision, we introduce the index inference score (IES), that relates the processing speed of the system, to the accuracy of the model. This parameter will determine the optimal number of frames to be fed to the network to decide and give a result.

The rest of the paper is organized as follows, in section 2, we develop a literature review of ArSL datasets, methods and results. In section 3, we present our methodology in recording the dataset and the proposed methods to recognize automatically the signs. In section 4, we present the experimental results, while in section 5, we mention the diverse problems encountered during the ArSL dataset recording stage. Finally, in the last section, we conclude our research and propose some extension ideas.

II. LITTERATURE REVIEW

In this part of the research paper, we will introduce the latest datasets used in ArSL as well as the methods and algorithms used for the sign recognition, including a detailed section for the hardware used to record the signs.

A. EXISTING ARSL DATASETS

Most of the previous studies in ArSL recognition have undertaken to record their own Arabic datasets. Unfortunately, many of these datasets are still not publicly available, or contain either few signs or enough signs, which are related to a certain country. In all the previous cases considered so far, these datasets require more investigation from authors, when they are not publicly available, or have modalities deficiencies and cannot be used in comparative studies.

Due to the lack of a public and rich Arabic Sign Language dataset (ArSL), many research groups opted for local recordings. Although these datasets contain tens or hundreds of signs, they are in many cases domain-specific or contain few signs and variations; and research findings need a good adaptation effort to be successfully reused. Additionally, some datasets either contain few words with few repetitions or two signers with many repetitions.

In general, these signs do not exceed hundreds. Thus, a generalization process cannot be invoked in both cases. In addition to the data acquisition and processing methods, the main bottleneck problem in SL recognition is known as the long sequences of successive signs over time.

In this paper, we focus specifically on some major ArSL findings, and the corresponding datasets used for the experimentation, emphasizing the availability of each dataset for other research groups, as illustrated in TABLE 1, which presents a short survey of some existing ArSL datasets, from which, we can notice three points: Firstly, the number of the signers does not exceed 40 signers, and the number of recorded signs varies inversely, teams were obliged either to increase signers or signs. The second point is the variety of recording devices, which starts from simple cameras such as

TABLE 2. Devices used in acquiring sign language data.

Device	Two Hands' Fingers Data	Body Joints	View	Type of Data	Remarks
Kinect 1 [7], [8]	No	Yes	F+S	RGB, Depth (Cloud Points)	0.4m to 8m depth range
Kinect 2 [9]	yes (Only 2 fingers / hand)	Yes	F+S	RGB, Depth (Cloud Points)	21 Body Key points + 2 Key points per hand 0.5m to 4.5m depth range
Leap Motion [10]–[12]	yes	Arms + hands	F+S	Skeleton raw data	Range (60cm)
RGB Camera [13], [14]	no	yes	-	RGB images	-
Intel Real Sense [15]	yes	no	F+S	Raw Data	Distance
Thermal Camera [16]	yes	no	-	20 x 3D space points	Hand Wrist Positioned sensors
Sensor Gloves [17]	yes	no	-	Raw Data (per sensor)	

R: Right, L: Left, D: Depth, F: Front, S: Side

webcams to RGB + depth cameras to cameras using infrared. The third point is that most datasets include words and short sentences, as sequence or single images cropped around the hands.

B. SIGN LANGUAGE RECORDING HARDWARE

Throughout our investigation, we noticed that many research papers on ArSL use very specific hardware sensors and diverse sets of cameras that are provided with depth sensors generating cloud points. In Table 2, we present some of the most known used devices in SL recordings. From which, we can notice that complexity in the data recording has varying modalities, ranging from cameras that generate simple RGB data to cameras or sensors that generate, at the frame level, multi-type data like RGB images, depth information and sometimes body skeleton, or devices that generate exclusively raw data of the hands, and the fingers.

C. METHODS USED IN ArSL

The automatic machine learning (ML) algorithms have allowed diverse SL recognition approaches to be developed over the last decades. Some recent research papers have used traditional ML algorithms, such as decision trees, Bayesian Network and Dynamic Warping [18], SVM [14], HMM [19], Grabbcut [20], Simple Neural Networks [6], [21], [22], for SL recognition. Although these algorithms show good performances on specific recorded datasets, results cannot be generalized, thus preventing easy deployment and necessitate full models retraining while adding new unseen signs. In many SL researches, papers have improved accuracies over time and with varying methods [12], [23], [24], but with some ultimate constraints, such as very selected datasets and controlled conditions, leading to difficulties in generalization.

SL is a varying sequential language, where diverse movements of the right and left hands, facial expressions, and sometimes body direction are used to express a single definition. Sometimes, the recognition could be restricted to a minimal movement of just one hand to express a number or a letter, or a more complex sequence of both hands and face. These details make the problem complicated, as it is not essentially a full repetitive set of actions that mean the same thing, but a constructed discourse [25] with verbs and actions, as well as facial expressions, including happiness, deception, disappointment, and other impressions that can be driven in synchronization with the hand movements. Authors of [26], have recapitulated, in a very organized way, all the methods categorizing sign language over a full pipeline methodology recognition process, from the sensor acquisition to processing, and then to recognition methods.

The comparison of the results in ML and specifically in ArSL is a very tedious task, and one cannot relate to the true story. Probable reasons for this issue is the lack of a unified test bench and test data, as per the datasets used in community challenges, a clear example is the speech NIST challenge [27], or similar challenges where rules are very strict and recording conditions are initially known and need to be taken into account.

Nowadays, sign language can be efficiently handled by machine learning algorithms, because it is a kind of gesture recognition, additionally signs can also be easily collected, because of two reasons: the first is the simplicity to find these gestural signs. The second reason is the ability to learn these signs by non-deaf people, allowing students and researchers to use a simple webcam and record a dataset, which can include tens of signs made by many signers. Table 3 illustrates some of the latest ArSL datasets and some of their properties, mentioned within their respective research papers.

Our interest is focused on Arabic SL, readers can refer to [28], [29], where a very thorough and informative study has been established. Let us recall that few papers recapitulate the state of the art in terms of SL recognition, for instance authors of [30], mentioned previous works from 1995 till 2004. Extended works of [31], and [29] mention deeper research in the SL field.

In this research paper, we tried to emphasis on some latest papers in automatic ArSL recognition, and their respective algorithms. Table 3 presents some of the latest studies in the ArSL recognition. We can notice that the CNN models are more related to processing high video frames, and prone to give results for long time segments. Results from [32] show a very high accuracy, but the number of signs is still limited to 40 basic signs. Reference [33] use a CNN model and obtained a 90% accuracy on SL letters. Work of [34] shows a very high accuracy on two signers, with 22 signs. Both [35] and [36] presented high accuracies but on a restricted set of images. Reference [37] used a BiLSTM model with 98.59% accuracy on isolated ArSL words.

III. METHODOLOGY

We propose in this research paper a framework for automatic recognition of ArSL. Firstly, we recorded an Arabic sign language dataset. Secondly, we use a deep learning method to recognize static and dynamic signs from the set of recorded videos, mentioning at each step the details of the experiments.

TABLE 3. Some algorithms used in the recognition of ArSL.

	Research work	Algorithm	Data Type	Signs /Signer	Recognition rate	Type of Data
Deep Learning methods	[32]	3D CNN	Arabic dataset	40 / 40	98.12% / 84.38%	Videos /Images
	[33]	CNN	Arabic letters	31 / -	90%	Raw images
	[34]	3D-CNN	Arabic	25/2	98%, 85%	Video
	[36]	CNN	Arabic	32 / -	99%	Images
	[37]	Bi-LSTM	Isolated words	23/3	89.59%	Videos
	[35]	CNN	Arabic alphabet	32/40	97.6%	Image
Traditional ML methods	[23]	SVM-KNN-ANN & DTW	Arabic medical gestures	42/2	89%KNN 91% Majority Voting	Kinect body joints
	[39]	Euclidean distance	Isolated words from daily school life	30 / -	97%	Videos
	[40]	KNN, SVM, MLP	Arabic words	23/3	99%	Image
	[41]	KNN/ HMM	Arabic sentences	40/	Vision-based: Word level = 95.60% Sent. Level = 89.17%	Sensor-based & vision-based

A. KSU ARABIC SIGN LANGUAGE DATASET

With the lack of an existing ArSL public dataset, covering sign diversity and numerous repetitions of the video sequences. We decided to build our own Arabic SL dataset, by following a defined methodology, where recording procedures, selection of tasks, and verification methods are inherited from our previous work on the public KSU Arabic Speech Dataset [42], [43], hosted in the LDC website [44], and the KSU speech voice pathology dataset [45], [46].

1) SIGNER AND SIGNS' SELECTION

We started by investigating and studying deeply different sign language datasets, specifically some well-known public SL datasets, namely the RWTH-Boston Dataset [47] and the ASL Dataset [48]. Then, we defined the required global recording parameters according to our resources (static and dynamic signs, number of signers, speed for making the signs, the recording cameras, the video storage extension, etc.,…), and established a procedure for the entire recording

Once all the video recording steps were revised, the team in charge of the sign's selection, a specialized sign language group from the Department of Deaf and HoH at King Saud University, proposed an initial list of 80 signs, (Arabic Alphabet letters, numbers, and some usual signs used in daily life). Some of the selected signs were static, like the numbers and most alphabetical letters. Others were dynamic signs, i.e., composed of a sequence of successive signs known as moving signs. The complete list of signs is presented in TABLE 4, selected signs comply to the reference ArSL dictionary [49] and were performed accordingly to this reference.

2) SELECTION OF THE SIGNERS

The recording procedure started with some sample signs. This initial step allowed us to estimate the time of recording and tune the cameras. We worked on the signers' recording procedure over three-time phases. At phase one, our goal aimed to record the signs by deaf students from HoH department. We did this under the supervision of a sign language translator, in order to ensure that the recorded signs are not local to the student and fully conform to the ArSL dictionary. We arranged this with our colleagues in the department. Unfortunately, this phase did not proceed as we planned. We arranged with almost the number of students we desired, but only five students worked with us in a serious manner. Regrettably, even with these five students, it was difficult to complete all the five required sessions, since these deaf students had a tendency of not accepting sign repetition, and sometimes were in a very bad mood for what they saw as long recording times. On average, each session lasted for more than 20 minutes. Even though, the deaf students were paid for this recording, which was also done in their department inside a managed office, they still found it annoying and boring. Moreover, some of them did not like to repeat and correct a sign that was not in accordance with the translator's dictation. After spending around 50 days on trying to record deaf students, we decided to search for another solution and went into the second phase.

At the second phase, an application for sign recording was launched at the university, as an alternative to the deaf recordings; About 40 signers were chosen, some of them were undergraduate, postgraduate, and PhD students. Training sessions allowed the students to be trained on the selected signs; this training phase took a couple of weeks, and allowed the students to replicate confidently the signs. Each signer revised the recorded signs templates many times. Details about the non-deaf signers and the recording team are presented in TABLE 5.

TABLE 4. Recorded static and dynamic signs.

Static Signs	['0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '10']	Numbers
	['Alf', 'Ba', 'Taah', 'Daah', 'Gim', 'Haa', 'Kha', 'Dal', 'Thal', 'Ra', 'Zai', 'Sin', 'Shin', 'Sad', 'Thad', 'Ta', 'Tha', 'Ain', 'Gin', 'Fa', 'Qaf', 'Kaf', 'Lam', 'Mem', 'Non', 'Waw', 'Ya', 'Ha',]	28 letters—Arabic alphabets
	['Father', 'Feel', 'Hospital', 'King', 'Manager', 'Meeting', 'Mosque', 'Prayer', 'Saud', 'Sorry', 'Thank', 'University', 'Where']	Common words
Dynamic Signs	['Alslam Alikom' (Salutation), 'Amir Riyadh' (Riyadh Governor), 'Arabic Language', 'Brother', 'Cold', 'Come In', 'Deaf', 'Death', 'Doctor', 'English Language', 'Evening', 'Family', 'File', 'Hot', 'How Are You?', 'Job', 'Medication', 'Morning', 'Mother', 'Name', 'Pain', 'Pharmacy', 'Reason', 'Sign Language', 'Sister', 'Surgery', 'Tired', 'Vacation']	Common words

3) RECORDING CONFIGURATION

In video recording and processing, the type of the camera, the calibration parameters, the room lighting, the distance from the camera, etc.., affect the quality of the images. To avoid the specific hardware that cannot be available during the deployment of the trained models, we selected the Microsoft

TABLE 5. Recording phases and signers.

Category	Number	Average Age	Used previously sign language	Instruction level	Phases
Non-Deaf Students	40 Males	25	No	Graduate (Master/ Ph.D.)	2, 3
SL Translator	2	25	Yes	Master	1
Programmers	3	32	No	Ph.D. Student	1, 2, 3
SL Verifier (Non-Deaf)	1	30	No	Ph.D. Student	1, 2, 3
SL Expert	1	55	Yes	Ph.D.	1, 2
Team manager	1	-	No	Ph.D.	1, 2, 3

FIGURE 1. ArSL dataset video recording setup.

TABLE 6. Devices used in the KSU Arabic sign language.

Device	Modality	Image Size	Field of View	Depth Distance	Bits	Additional Info
Kinect V1	RGB	640 x480x3	58.5° x 46.6°	-	24bits	Up to 30 fps
	Depth image	320 x 240 pixels		0.4 to 8m	11bits	
Kinect V2	RGB	1280x920x3	70.6° x 60°		-	Up to 30fps
	Depth image	512 x 424 pixels		0.5 to 4.5m	13bits	
	Skeleton Data	2D points				
Sony HandyCam	Continuous recording	1920 x 1080HD	-	-	-	24p/60i /60p

Kinect cameras, both version 1 and 2, for their low cost and acceptable resolutions.

The recording setup is presented in Figure 1, where the signers were asked to position themselves facing the cameras at a distance of 1 to 2 meters from both cameras and background wall.

We additionally used a Sony handheld camera as a standby solution for continuous recording in case the program controlling the two cameras got bugs during live session. The physical disposition of the cameras was initially inspired from the RWTH-Boston Dataset for American SL [47], since they were recording the signers by diverse cameras at diverse angles. We had to compensate for the side angles cameras, by including the depth information of the Kinect cameras. All our cameras were facing the signer. The details of the cameras and the recording modalities are listed in Table 6 and additional details of the Microsoft Kinects are found in [50].

4) ArSL RECORDING STATISTICS

The recording stage started by two to three signers per day, where each signer took approximately three hours to complete the five required sessions. A quality controller PhD student oversaw the sign shape and speed. The two Kinects recorded each sign alone, giving time to the signer and the team to tune the various parameters, such as the signer position, the camera up and down adjustments, and the checking of the recording.

The team in charge of the recording checked the videos daily, and made backups at the end of each session. The full dataset containing the KinectV1 (RGB + Depth), KinectV2 (RGB + Depth + Skeleton) and handy-cam Sony recordings, was approximately 450GB. Each signer had 80 videos per camera, and 40 signers recorded each sign five times, totaling in 16000 videos per camera. Each video was recorded with a frame rate of 30 images/sec, inside a classroom with a light gray background color, and a neon lighting. No clothing constraint was imposed on the signers (long sleeves, short sleeves, ...), aiming to have a SL dataset as near as possible to the wearing conditions in real life cases, primarily when our SL solution will be deployed outside of the lab.

5) POST PROCESSING ON THE KINECT VIDEOS

The videos of each of the 80 signs were checked manually. The unnecessary start and end waiting frames of each video were trimmed from the videos, these waiting frames ranged from two to four seconds (60 to 120 frames).

The KinectV1 and KinectV2 were recorded by the same program, allowing to have the same starting and ending times. This hopefully allowed us to process the frames content using the same index for both videos.

The second phase of the post processing is the checking of the sign videos content frame by frame. It allows the enumeration of the index of the frames containing blurred hands. Frames were scrolled in a manual way, with the help of superposed key points, as described in section VI.A. An example of some frames from one video recording is shown in FIGURE 2.

B. PROPOSED SYSTEM

The main bottleneck problem in SL is the detection of the hands, the arms, the face, and the exact location of the fingers.

Once the output from a camera or a video file is fed to the OPL network, the hands and body key points are generated, as shown in the global view architecture of Figure 3.

Each RGB frame is thus converted to a set of 2D key points. The system waits to collect a small sequence of frames and sends this sequence to the SKN network to generate the recognized Arabic sign.

1) OPENPOSE NETWORK ARCHITECTURE

One of the state of the art framework in pose estimation is the OpenPose library (OPL) [51]. The OPL is a very powerful framework, as it estimates, from a single image, a set of key

FIGURE 2. Example of the OPL output on the ArSL dataset.

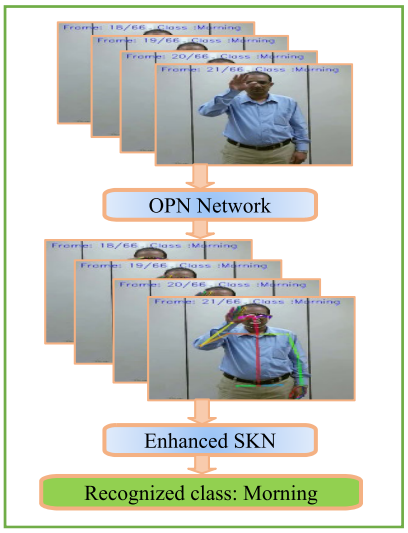

FIGURE 3. General overview of the proposed system.

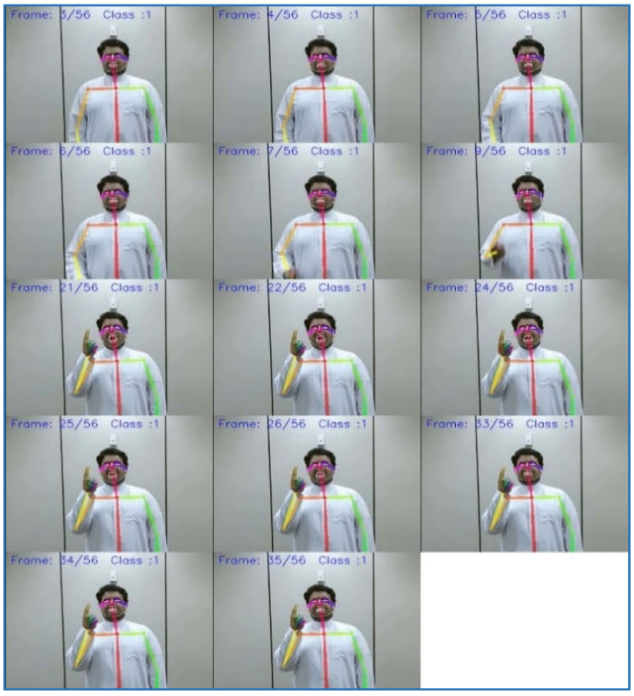

FIGURE 4. Example of the OPL output on the ArSL dataset, static sign—joints correctly identified.

points of one or more persons, and provides with a very high accuracy, the body, hands and finger joints. This human pose estimation library was developed in C++ and uses Caffe as a backbone deep neural network. The second network that will be used is a point convolution network that will be adapted and tuned to accept a two-dimensional series of data and will output the probability of an ArSL sign. OPL has surpassed the state-of-the-art research in recognizing or estimating the pose of individuals. The latest version of OPL can generate 21 key points per hand, 70 key points for the face, and 25 key points for the body/foot. The body key points might differ depending on the trained model (COCO, MPI), and the key points can be generated in real time. The OPL network, when fed by one or more successive frames, generates for each frame a set of X, Y coordinates, a probability or confidence score for each joint of the body, and relevant points within the face. The initial structure of the OPL is identical to the VGG19 network, where input images are fed to the first ten layers of the VGG19. The output feature maps are then transferred to a second network consisting of two parallel six stages sub-networks. OPL generates all the 2D key points from the input image.

FIGURE 4 and FIGURE 5 show some sample output images from our KSU-ArSL dataset, where joints are correctly identified, while FIGURE 6 focuses on some of the intermediate frames where the OPL could not generate a valid hand skeleton (No or bad finger candidate key point generated).

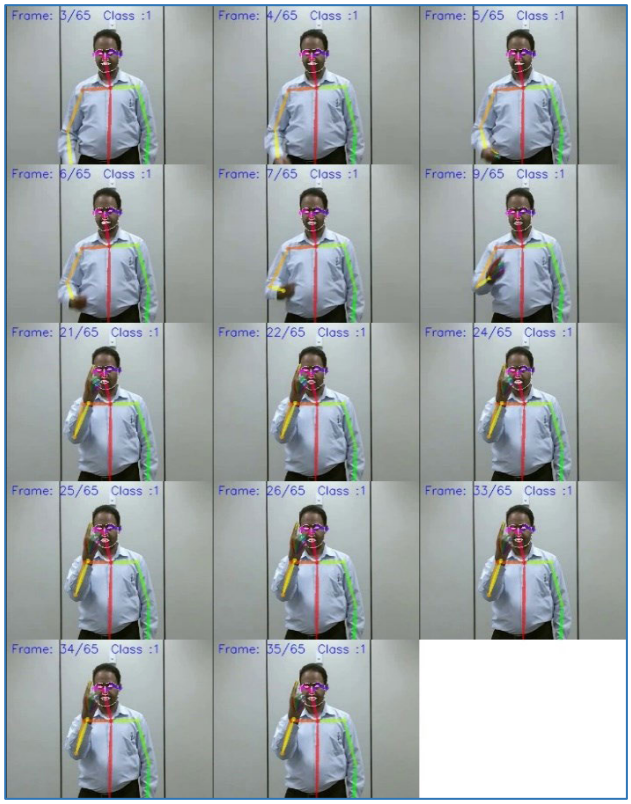

FIGURE 5. Example of the OPL output on the ArSL dataset, static sign—joints correctly identified.

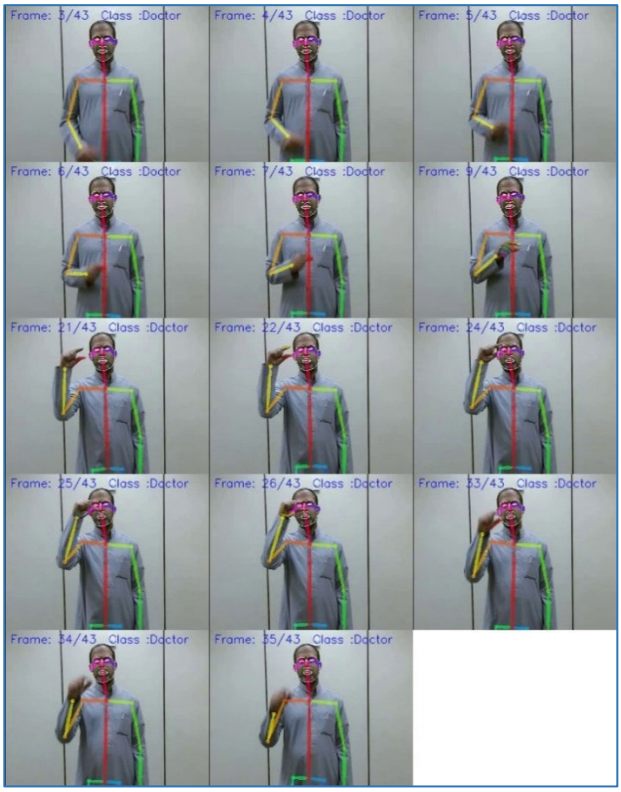

FIGURE 6. Hand skeleton generation errors, right hand fully/partially blurred (Dynamic Sign)— finger joints not identified.

TABLE 7. List of body Joints used as input to the SKN.

	Number of original OP Key-points: (X, Y, Confidence probability)	List of selected key points used in the SKN
Body	(0 to 24): 25 x3	(0,1,2,3,4,5,6,7, 15,16,17,18): 12
Hands	(0 to 20): 21 x 3 x 2 hands	(0 to 20) 21 x 2 x 2 hands:84
Face	(0 to 69): 70 x 3	-
Total	25x3 + 21 x 3 x 2+70 x 3 = 411	21 x 2 x2 +12 = **96 48 (X, Y) coordínate**

2) ENHANCED PARALLEL SKELETAL DEEP CNN

Originally inspired from [52], the authors of [15] proposed a skeletal CNN network (SKN) fed by three dimensional coordinates X, Y and Z, generated by the 3D Intel real sense depth camera software development kit. This type of camera generates 3D key points of the hands within a 3D space. Results from the paper [15] have shown an accuracy of 91.28% for 14 gesture classes from the DHG [53] dataset, but had limitations as it dropped by 7%, when doubling the number of classes to 28.

Our contribution to the modified SKN network dealt with two major points: Firstly, we simplified the network architecture to use 2D points instead of 3D points, which led to 1/3 less computation per network branch. Secondly, we extended the input layers to use the full body 48 key points, see Table 7, instead of only the two hands 3D points generated by the 3D Intel real sense.

Given the selected set of OPL key point coordinates, we fed these points to a second parallel network that inputs each key point to three parallel branches, a low-resolution, a high-resolution, and a pooling branch. The difference between the network proposed by [15] and the OPN network, is that SKN convolutions are point convolutions and no connection or concatenation occurs at mid-stages, a detailed view of the network is presented in Figure 7. Each of the channels of the network is a component of a multivariate time sequences. Variations over each channel are independent and do not participate in the update of the weights of the other components.

Each input channel size was modified to accept a two-dimensional X, Y coordinates generated by the OPL network. We do not provide nor use the depth information, as in the original design, but we enrich the input with some additional selected body joints, instead of the hands only, as per the original design. Details of our model joints are presented in Table 7.

IV. EXPERIMENTAL DESIGN & RESULTS

Our experimental methodology is driven by three evaluation criteria:

The first criterion is to obtain an adequate accuracy for the different signers, seen or unseen in the training session. This target is more associated to developing a model that can be efficient for new signers when deployed in demo mode, or to

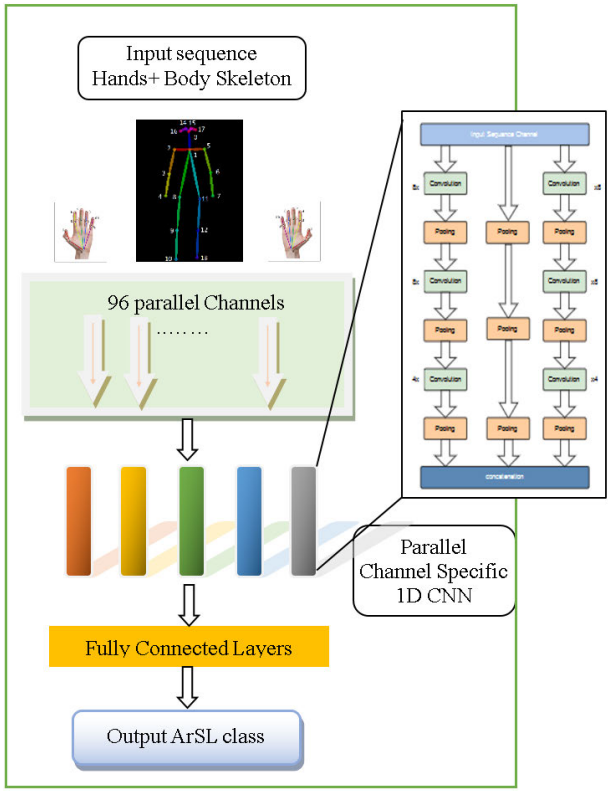

FIGURE 7. Enhanced architecture of the SKN.

new signs when extending the dataset. The second criterion is to propose a reference benchmark for ArSL comparison. Our experimental results for the automatic ArSL system are presented in various metrics, such as class accuracy, f1-score, recall, and precision. The third criterion is to tune the system's parameters and improve the system's response to reach real-time recognition. These experiments have been motivated by the optimization of the time to generate a decision given an acceptable real-time accuracy. This tradeoff is very peculiar in the sense that OPL, when fed by a frame, makes a set of convolutions and concatenations that can be a bottleneck in real time. In addition, the second network when fed by the set of reduced points would also require some milliseconds to generate a decision. Thus, real time might not be achieved on the fly, but with a small delay, and though could be optimized through experimental setup.

A. PERFORMANCE METRICS

The problem of identifying signs, within a sequence of frames, is a classification problem, and the metrics needed to determine the accuracy on the testing set have included the precision, the recall and F1-score [54], as well as the recognition accuracy, but for most of the experiments, we will only display the recognition accuracy per video sequence, as a single value. This value is simpler to analyze and compare to other previous research papers. In addition, we introduce a new parameter called index inference score to find the optimal number of frames required by the system, in order to determine the relevant class in real time situation.

1) ACCURACY METRICS

The used metrics are defined as follows:

$$\text{precision} = \text{TP}/(\text{TP} + \text{FP}) \tag{1}$$

$$\text{recall} = \text{TP}/(\text{TF} + \text{FN}) \tag{2}$$

$$\text{F1-score} = 2 \times (\text{precision} \times \text{recall})/(\text{precision} + \text{recall}) \tag{3}$$

$$\text{Accuracy} = \text{TP}/(\text{TP} + \text{FN}) \tag{4}$$

where, TP, FP and FN are the true positive, false positive and false negative rates respectively.

2) INDEX INFERENCE SCORE (IES)

The ingredients to achieve real time or near real time sign recognition are all the composition of hardware and software that run and execute sequential and parallel processes in few hundreds of milliseconds. In most cases, the recognition has to be within the next few seconds. In our case, while input collecting and processing a certain number of frames, a decision needs to be made on some previous frames. To analyze the tradeoff of selecting the accurate number of frames required for the decision making, against keeping an optimal accuracy, we are exploring the index IES for this analysis.

We recall that in video recognition processing, sending too many frames to the system impacts additional latency, whereas sending fewer frames reduces accuracy [55]. This tradeoff of accuracy against speed problem has been discussed by [56], and established that the index efficiency score can be used under certain conditions. This index gives the best number of frames to send to our system, allowing an optimal accuracy.

The IES can be computed as per equation (5):

$$IES = \frac{RT}{1 - PE} = \frac{RT}{PC} \tag{5}$$

where PE represents the percentage error, PC is the percentage of correct answers, and RT is the response or reaction time. In our case, we propose to map RT to the number of frames upon which the system decides.

We are basing our consideration on the fact that frames have similar time to be propagated from the camera throughout the first OPL network.

The target of the real time can be achieved once the primary requirement of a powerful GPU is available, as per the OPL framework requirements for the best frame rates.

The full pipeline delay is the addition of each of the following stages:

a) Frame acquisition
b) OPL (image to key points)
c) Elongation
d) Key points frame stacking
e) SKN network sign decision

TABLE 8. Signer dependent data splitting.

	Train	Valid.	Test	Exp. ID
Sessions' ID	1, 2, 0	4	3	0
	3, 4, 1	2	0	1
	1, 2, 0	3	4	2
	3, 4, 1	0	2	3
# of Samples / Signer	3	1	1	
#signs	80	80	80	
# of signers	40	40	40	
80 Mixed Signs	**9600**	**3200**	**3200**	
28 Static Signs	**3360**	**1120**	**1120**	
52 Dynamic Signs	**6240**	**2080**	**2080**	

TABLE 9. Signer independent data splitting.

	Train	Validation	Test	Exp. ID
Signers' ID	1, 2 ,10 to 31	3 to 9, 40	32 to 39	0
	2 to 9, 25 to 40	10 to 17	1,18 to 24	1
	1, 2, 10 to 31	32 to 39	3 to 9, 40	2
	2 to 9, 25 to 40	1,18 to 24	10 to 17	3
# of samples/ signer	5	5	5	
# signs	80	80	80	
# of signers	24	8	8	
Total Samples/Exp.	**9600**	**3200**	**3200**	

At run-time, steps one to three have nearly fixed delays, while step four is directly related to the reaction time, where the more frames that are stacked, the slower the SKN. Step 5 generates a decision for the associated sign class.

Once the IEs is computed, it will determine the optimal number of frames to be fed to the network to decide and give a result. Less frames might be not sufficient; more frames would give a bigger delay in the ArSL recognition.

B. TRAINING AND TESTING DATA

The experiments were conducted on signer dependent mode, where different samples from the same signers are seen in training and testing, and in signer independent mode, where signers at training are not seen at testing. Experiments considered different temporal variations of the central selected frames for each sign, as explained in section 7.B.

In all the recorded videos, the static signs are mostly located at the central frames of each video recording, each sign lasts for one second, (approximately 30 frames), while dynamic sign range longer frames within the recorded videos. (approximately two to three seconds per sign). We opted for an automatic selection of the frames starting from the central frame of each recorded sign, and some frames around the central frame were gradually added, depending on the sign length and type.

When the OPL is fed with a sign video, with a single signer, it generates 137 triples key points per frame (see Table 7), where each triple value includes the (X, Y) image coordinates, of each joint and face key points, and the probability or confidence score of each corresponding key point.

In the signer dependent experiments, we used three recordings for training, a fourth recording for validation and the remaining recording was used for testing. Diverse variants of signers' swapping have also been investigated. The different configurations of signers and signers' sessions used in the training, validation, and testing are presented in tables Table 8 and Table 9, respectively.

We trained our signer dependent and independent networks with approximately the same proportion of samples (75% training, 25% testing), and we focused on the diversity of the signers to avoid training and validation with some selected best signers, as shown in the experiments Ids 0 to 3 of Table 8 and Table 9, where data is split in different manners.

In the independent signer experiments, we used 24 random speakers with all their five recordings for training, eight signers for validation, and eight other signers for testing, as detailed in Table 9.

C. EXPERIMENTAL RESULTS

Our proposed network requires a fixed length of inputs, whereas the ArSL does not permit the making of the signs in a fixed duration, i.e., one cannot make the sign in a specific fixed time duration, or a set of frames. Thus, we adjusted the time length of the recorded videos of the signs to have the same sequence length. We adopted a spline interpolation elongation method, where all the generated OPL sign sequences are set to a fixed number of frames by a replication technique. Given a sequence of frame values, the interpolation technique replicates the sequence in one of the following manners, using a spline filter either by reflection, addition by appending a constant value, approximating to a nearest value, mirroring and/or wrapping around central values. In our experimental design, we opted for reflection to replicate the values (a b c d) to a longer sequence of the form: (d c b a | a b c d | d c b a), where (a, b, c, d) are successive frame key points generated by OPL. The optimal experimental sequence for the total frames' elongation was found to be 100, allowing up to three seconds and more (>90 frames) as input to the SKN network.

1) DYNAMIC SIGNS RESULTS

The dynamic signs are all the signs that require the continuous movement of one or two hands. Sometimes the head movements or face expressions can also be part of the sign. The results of the dynamic signs accuracies are presented in Table 10 for various number of central frames taken from each sign video. The best attained accuracy is 98.39% for 61 successive central frames, for approximately two seconds video duration, before elongation.

In Table 11, the signer independent results are presented, and it clearly shows that experiment 0 gives best results, allowing an accuracy of 97.23% for the 100 successive frames, approximately three seconds time duration at 30 fps.

TABLE 10. Dynamic Signs: Signer dependent accuracies (%).

Exp. ID	11	21	31	41	51	61	71	81	91	100
0	81.51	93.3	95.08	97.85	98.21	98.39	98.39	98.12	97.41	96.87
1	88.48	92.05	95.71	96.69	96.33	96.78	97.05	96.87	96.6	96.96
2	87.05	94.37	96.51	96.87	96.51	97.5	97.85	98.03	97.41	97.41
3	85.89	92.58	93.12	95.62	94.91	93.66	95.53	94.55	95.26	95.8
Avg. Acc. / # frames	85.73	93.08	95.11	96.76	96.49	95.98	97.21	96.89	96.67	96.76
Average for IES	95.07									

TABLE 11. Dynamic Signs: Signer independent accuracies (%).

Exp. ID	11	21	31	41	51	61	71	81	91	100
0	85.89	89.19	95.53	96.25	94.73	96.69	94.19	96.51	95.08	97.23
1	85.71	91.78	94.19	95.17	96.25	95.8	93.3	96.07	95.44	96.6
2	82.94	90.8	93.48	93.75	91.78	93.92	93.83	92.67	92.32	92.76
3	88.12	91.15	94.46	93.83	93.57	92.67	91.6	94.46	96.07	94.55
Avg. Acc. / # frames	85.67	90.73	94.42	94.75	94.08	94.77	93.23	94.93	94.73	95.29
Average for IES	93.26 % NEW									

TABLE 12. Static Signs: Signer dependent accuracies (%).

Exp. ID	7	13	21	31	41	51	61	71	81	91	100
0	83.22	86.49	85.62	87.31	84.72	84.23	84.23	86.87	85.81	81.15	86.49
1	84.66	85.19	86.15	86.49	88.6	86.87	86.05	87.21	85.67	87.21	88.36
2	82.21	86.44	85.76	85.91	85.91	86.68	86.63	88.89	87.21	88.12	87.11
3	85.14	85.48	85.86	87.21	84.9	84.37	83.46	86.34	85	85.24	86.15
Avg. Acc. / # frames	83.81	85.9	85.85	86.73	86.47	85.54	85.09	87.33	85.92	85.43	87.03
Average for IES	85.92 %										

The second best accuracy occurs at 61 frames, which can be more useful if speed is an important factor in the decision.

Most dynamic signs in the signer dependent mode are recognized with a high accuracy, whereas few signs have shown lower recognition results, as presented in the confusion matrix (CM) of FIGURE 8, as confusion appears between the signs ''Doctor'', '' Hello'', and ''English language'', where the signs have similar sign parts in common, this lead to the full indistinctness.

For the signer dependent mode, the networks over all the experiments were very sensitive to the selection of the signers. Additional experiments in section 4, will allow us to focus on the signers that improved or reduced accuracy.

2) STATIC SIGNS RESULTS

The static signs are all the signs that require a single gesture or a fixed position of one or both hands in a normal SL dialog. Therefore, by intuition, the fewer the frames are presented to the system, the faster the decision can be made.

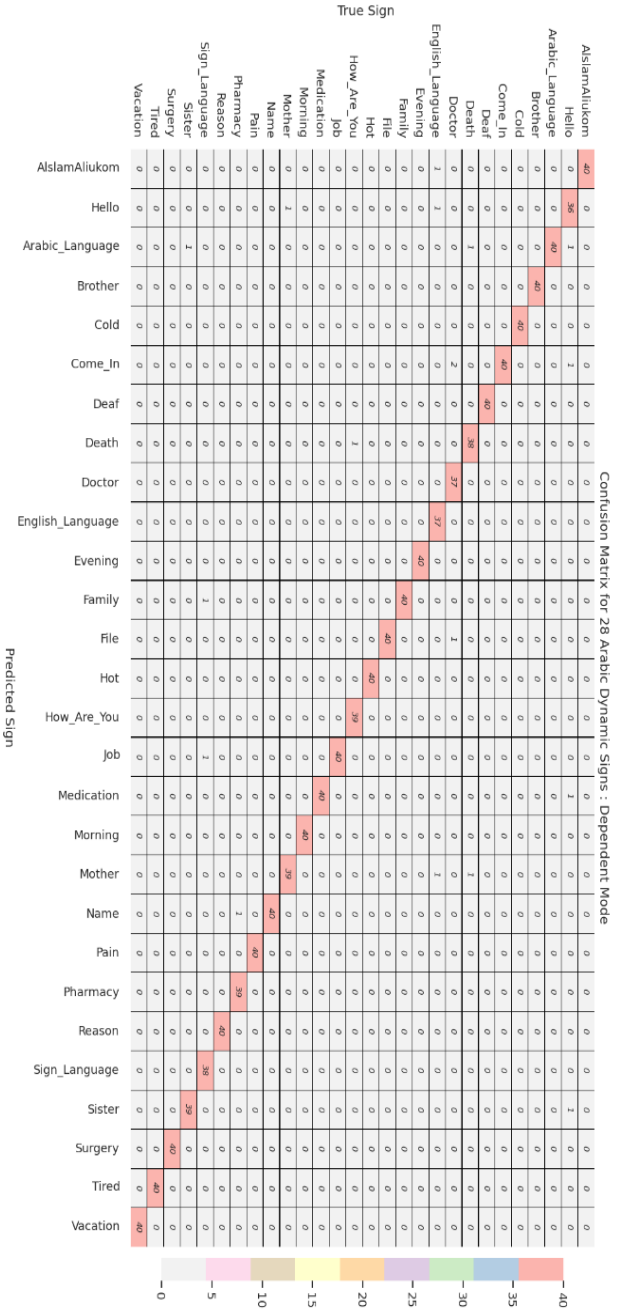

FIGURE 8. Confusion matrix of the dynamic signs in the signer dependent mode: Best testing accuracy of (98.39%).

The results of the static ArSL recognition are presented in Table 12, where it shows that from the average of all experiments, more than two seconds can be an optimal frame number for the system to better identify the sign; Best result occurs at 71 frames with a value of 88.89%, results show light fluctuations, as accuracy values range from 81.15% to 88.89%, with a standard deviation of +/−1.59%.

We notice, from Table 13, that static signs in the independent mode require less frames to be processed, best result occur at 31 frames (one second duration), with a value of 86.34%. Results show very high fluctuations, as accuracy

TABLE 13. Static Signs: Signer independent accuracies (%).

Exp. ID	7	13	21	31	41	51	61	71	81	91	100
0	82.45	81.39	82.88	84.66	83.36	77.93	83.12	82.59	78.17	78.84	85.52
1	82.64	85.33	84.85	86.34	84.9	81.1	84.08	81.05	78.36	84.03	76.63
2	75.19	81.29	81.92	82.54	82.93	80.09	77.69	76.82	80.52	82.98	82.45
3	76.53	78.41	78.41	79.23	73.31	79.08	77.4	75.33	76.25	72.5	75
Avg. Acc. / # frames	79.2	81.61	82.02	83.19	81.13	79.55	80.57	78.95	78.33	79.59	79.9
Average for IES	80.37 %										

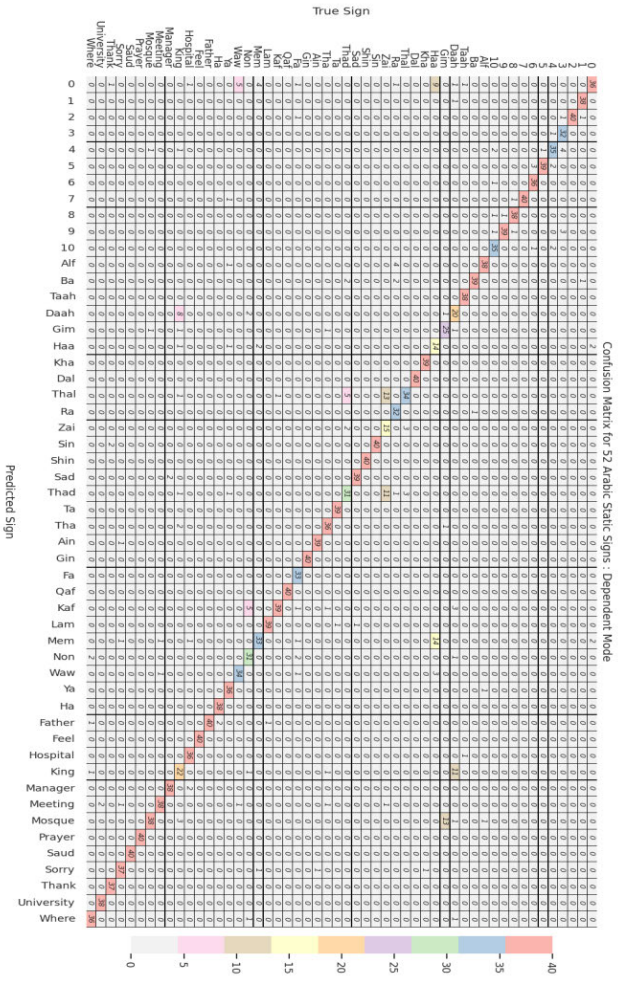

FIGURE 9. Confusion matrix of the static signs in the signer dependent mode, (Best testing accuracy of 88.89%).

values range from 72.50% to 86.34%, with a standard deviation of +/−3.52%. This is very related to the high similarity of the shapes of the signs, where some numbers and letters appear to be very similar in many video sequences. The confusion matrix of best signer dependent mode is illustrated in FIGURE 9, alphabet letters like ''Haa'', ''Mem'', and ''Zero'', and ''Zai'', ''Thad'' and ''Thal'', with similar spatial shapes lead to the decrease in the global accuracy.

TABLE 14. IES of the signer dependent and signer independent modes.

	Signer Dependent		Signer Independent	
	Static Signs	Dynamic Signs	Static Signs	Dynamic Signs
Average Frames	51.73 (~51frames)	55.90 (~55frames)	51.73 (~51frames)	55.90 (~55frames)
Average Accuracy(%)	85.92	95.07	93.26	80.37
IES – Avg. Acc.	0.601	0.588	0.643	0.599

TABLE 15. 80 mixed signs accuracies (%).

Exp. ID	0	1	2	3	Average Values
Signer Dependent Accuracy (%)	89.62	89.62	88.71	88.40	89.09
Signer Independent Accuracy (%)	88.09	83.31	87.31	82.75	85.36
Accuracy Difference (%)	1.53	6.31	1.4	5.65	**3.72**

3) MIXED STATIC AND DYNAMIC SIGNS RESULTS

An optimal system is made of sub-optimal systems; once the static and dynamic networks are optimal, the concatenation of both systems is hoped to be optimal. The IES metric is used in this scenario to find the best trade-off accuracy/speed (where speed is related to the number of frames fed to the system upon which this latter generates a recognized sign). From Table 14, we notice that the smallest IES occurs at the signer dependent mode, with a value of 0.588, Unfortunately, the dependent mode cannot be driven to real time, as the recorded dataset signers are not available all the time while deploying the system elsewhere.

Thus, the optimal candidate IES value is taken from the signer independent mode, with a value of 0.599. This would allow the system to be used by unseen new signers and optimally use approximately 55 successive frames per sign in all the next experiments.

When both static and dynamic signs are used together for training, the static signs have affected the recognition of the dynamic accuracies, which led to the decrease in the average accuracy of all the signs. This is best presented in Table 15 where we can clearly that our system shows some signer independency, as the average difference in accuracy in both signer dependent and sign independent is less than 4%, over all the experiments.

To improve and tune up the system and depict the difficult signs and signers that were not actively participating in increasing the accuracies, we compare the best configurations in the signer dependent mode. The comparison of the average static and dynamic signs per class test metrics are shown in Table 16, where clearly dynamic signs are more prone to be recognized compared to static signs. We confirm also that alphabet letters have very low metrics, for instance the precision metric of the numbers and the alphabet are less than the precision of the dynamic signs by 0.10, spatial similarity of the numbers and the alphabet contributed to this difference.

TABLE 16. Average metrics (%) for successive 55 central frames mixed signs – dependent mode.

Class	precision	recall	f1-score
Numbers (Static)	0.83	0.84	0.84
Alphabet (Static)	0.80	0.77	0.77
Other Static Signs	0.93	0.96	0.94
Dynamic Signs	0.97	0.97	0.97

TABLE 17. Test results for signers' shuffling (52 static signs – 55 frames).

Experiment ID	Signer ID's (Test)	Test Recognition Accuracy (%)	Experiment ID	Signer ID's (Test)	Test Recognition Accuracy (%)
EXPERIMENT ID=0	32	81.42	EXPERIMENT ID=2	3	85.0
	33	81.42		40	85.71
	34	92.14		4	92.14
	35	85.64		5	90.0
	36	89.27		6	90.0
	37	87.14		7	95.0
	38	89.27		8	85.0
	39	91.42		9	89.28
EXPERIMENT ID=1	18	90.71	EXPERIMENT ID=3	10	82.14
	19	83.57		11	83.57
	1	90.71		12	93.57
	20	85.71		13	85.71
	21	82.85		14	87.85
	22	86.42		15	90.0
	23	88.57		16	87.85
	24	87.85		17	87.14

Both ArSL alphabet and numbers are signs occurring at the palm and fingers level, more investigation needs to be done at this area of the image. In the next section, we will show the results per signer in the signer independent mode. This will allow us to determine which signers have positively participated in increasing/decreasing the accuracy in the sign recognition process.

4) FOCUS ON SIGNER INDEPENDENT MODE

To validate the results, we developed some additional experiments, focusing only on the independent signers. This is the typical model for a real scenario when deploying our sign language recognition solution. The diverse experiments were oriented to see and find out how some signers have negatively impacted the quality of the whole dataset recording. In other words, why the classification deep network could not reach higher rates of recognition and consequently determine the signs and signers that were hard to recognize [57]. In some cases, hard samples are important for research because finding solutions to these cases makes the system more robust. Splitting ID's in independent signer's modes was already described in Table 9. The per signer test accuracy results are presented in Table 17, where signers' splits are detailed for each experiment individually. The signers used for validation in some experiments were used as testing in other experiments and vice versa.

TABLE 18. Test results for signers' shuffling (28 dynamic signs – 55 frames).

Experiment ID	Signer ID's (Test)	Test Recognition Accuracy (%)	Experiment ID	Signer ID's (Test)	Test Recognition Accuracy (%)
EXPERIMENT ID=0	32	97.85	EXPERIMENT ID=2	3	95.71
	33	99.28		40	97.14
	34	97.85		4	98.57
	35	97.14		5	97.85
	36	97.85		6	96.42
	37	99.28		7	98.57
	38	99.28		8	100
	39	98.57		9	98.57
EXPERIMENT ID=1	18	98.57	EXPERIMENT ID=3	10	95.71
	19	98.57		11	95.71
	1	97.14		12	95.0
	20	96.42		13	97.14
	21	98.57		14	97.14
	22	97.14		15	98.57
	23	93.57		16	94.28
	24	96.42		17	90.71

TABLE 19. Test results for signers' shuffling (80 signs – 55 frames).

Experiment ID	Signer ID's (Test)	Test Recognition Accuracy (%)	Experiment ID	Signer ID's (Test)	Test Recognition Accuracy (%)
EXPERIMENT ID=0	32	87.14	EXPERIMENT ID=2	3	87.14
	33	94.28		40	86.42
	34	84.28		4	87.14
	35	87.85		5	85.08
	36	89.28		6	85.7
	37	89.28		7	90.0
	38	86.42		8	90.71
	39	90.71		9	90.0
EXPERIMENT ID=1	18	91.42	EXPERIMENT ID=3	10	83.57
	19	86.42		11	87.14
	1	87.85		12	86.42
	20	92.14		13	90.71
	21	90.0		14	93.57
	22	90.71		15	86.42
	23	87.85		16	95.0
	24	92.85		17	89.28

As expected, some signers such as signers 32, 33, 19, 21, 11, have not been well recognized, leading to a global decrease of the average accuracy. These signers would require additional pre-processing and/or frames discarding.

For the dynamic signs results, shown in Table 18, only signer 23 and 16 have shown accuracies below 95%, other signers were recognized at very high rates. Example of the test instances of signer 8, which were fully recognized.

When both static and dynamic signs are mixed, experiment 1 shows better average recognition values with a minimum value of 86.42% for signer 38 and 92.85% for signer 24, and a standard deviation of +/−2.30%, while experiment 0 shows a minimum of 84.28% for signer 34 and a maximum of 94.28% for signer 33, with a standard deviation of +/−3.01%.

TABLE 20. Comparison with previous results.

Research Work	Number of Signs	Signer Dependent Accuracy (%)	Signer Independent Accuracy (%)
3D-CNN [32]	40	98.12	84.38
MLP [58]	40	98.62	87.69
Auto-Encoder [58]	40	98.75	84.89
Our Model using the 40 Signs of [32] &[58]	40	**96.71**	**93.29**
Our best model (Static)	52	**88.89**	**86.34**
Our best model (Dynamic)	28	**98.39**	**96.69**
Our best model (Mixed Signs)	**80**	**89.62**	**88.09**

The maximum of all signers occurs at experiment 3 with a value of 95%, but the worst signer 10 with an accuracy of 83.57%, increased the range between the maximum and minimum value to 11.43%, which shows a large sparsity in the results, and a decrease in the overall average.

5) COMPARISON WITH PREVIOUS WORKS ON THE SAME ARSL DATASET

In our previous research papers [32] and [58] on the same dataset, 3D CNN ArSL recognition systems were developed, but on a restricted set of signs.

These approaches were based on feeding CNN networks with 2D images as a sequence, with different time stamps from each video. The result accuracies dealt with only 40 signs of the 80 recorded ones. A comparison, on the same ArSL dataset is shown in Table 20, from which we can notice: firstly, that our new proposed system is more robust to sign independency, as accuracy drops between dependent and independent systems with an average of 2.3%, which shows that the models, with unseen data generalize quite good, compared to the previous work of [32] and [58], where the drop between dependent and independent models reached 12.84%. Secondly, our model generalizes well in terms of adding new signs, as the doubling of the signs from 40 to 80 affected our model by a drop of 7.09% for the dependent mode and 5.2% for the independent mode. Let us recall that the static signs include the numbers and the alphabet, which are very hard signs to be recognized in video sequences containing full body information, as they represent just few pixels in the image, and differ from each other at the level of the palm and/or fingers.

V. PROBLEMS & DIFFICULTIES

In this section, we will summarize the diverse problems that faced us during the dataset preparation, the recordings, as well as the factors that influenced the recognition rates of some experiments.

The dataset was recorded in using three separate cameras, in an office with a gray background. The field of view of each camera was too large and instead of recording only the signer, we additionally recorded the whole scene, which induced extra information in the horizontal direction, leading to a 2/3 of useless sides pixel information. This could have been avoided if a focus-like system was used.

In addition, some of the signers that were trained by a sign language trainer, weeks before the recording sessions, had difficulties in synchronizing their hands and making the gestures, identically over the 5 sessions.

The skeleton data of the KinectV2 were not very useful because the skeleton's process increased latency—and many skeletons could not be generated at the same frame synchronization rate, leading to delays in saving the frames and the skeleton files—and induced some interruptions while recording. The depth of information from both Kinect cameras required alignments with the RGB, given the calibrated camera parameters, but we mostly relied on the original factory calibration parameters that were sub-optimal.

For many blurred frames, OPL did not generate accurate finger key points, mostly for the transient frames (frames before and after the central sign frames), this was due mainly to some blurred pixels in the original frames, a faster frame rate would prevent such blurring and improve OPL finger detection.

VI. CONCLUSION & FUTURE WORK

This research paper proposed an Arabic sign language automatic recognition framework, which consists of using a new ArSL Dataset recorded at our university premises. The dataset was recorded with three cameras, a Kinect V1, a Kinect V2, and a Sony handy cam. The 40 signers recorded each 80 signs, five times, resulting in a multimodality dataset that comprises RGB images, Depth images and body skeletons from the Kinect V2.

In this paper, we investigated exclusively the RGB images of the Kinect V2 by proposing the concatenation in serial of two parallel networks, a 2D CNN network for key-points estimation and a second 1D CNN skeleton network. The automatic ArSL results are very promising, as our best network configuration recognized 98.39% for dynamic signs and 88.89% for static signs in the signer dependent mode, and an accuracy of 96.69% for dynamic signs and 86.34% for static signs in the signer independent mode. When the same network is trained by both dynamic and static signs, a test accuracy of 89.62% for the signer dependent and 88.09% for the signer independent mode were recorded. The use of the inverse efficiency score showed that with an optimal number of sufficient frames, as input to our system, the trade-off between accuracy and speed could be enhanced, if such models are deployed on production.

A complementary solution would be to add a sign boundary detector or a network that can decide whether the actual shape is transient gesture or a sign. More investigation will be axed on the frames between effective signs and their properties in the continuous ArSL. To enhance the accuracy of each signer in both dependent and independent modes, more research would be made on detecting the minimal set of distinguishing

frames and/or key points that represent a sign by clustering the generated frame key points into representative reduced clusters, so that the deep model can be compact and lighter on mobile devices. Another additional point is to improve delay removal in the whole pipeline via convolution suppression and optimal data propagation, aiming to reduce the network size and optimize the classification speed. Some other improvements would be to zoom on fingers and add an automatic technique to detect the palm orientations because we noticed that the drop in accuracy was mainly due to the difficulty to detect the similar signs that shapely resemble each other, but differ by one finger or by the palm orientation.

ACKNOWLEDGMENT

The authors would like to extend their appreciation to the Deanship of Scientific Research at King Saud University for funding this work through research group no RG-1437-018. They also would like to thank the Deanship of Scientific Research and RSSU at King Saud University for their technical support.

REFERENCES

[1] T. Shanableh and K. Assaleh, "Telescopic vector composition and polar accumulated motion residuals for feature extraction in Arabic Sign Language recognition," *EURASIP J. Image Video Process.*, vol. 2007, p. 10, 2007, Art. no. 87929, doi: 10.1155/2007/87929.

[2] G. Latif, N. Mohammad, J. Alghazo, R. AlKhalaf, and R. AlKhalaf, "ArASL: Arabic alphabets sign language dataset," *Data Brief*, vol. 23, Apr. 2019, Art. no. 103777, doi: 10.1016/j.dib.2019.103777.

[3] S. M. Shohieb, H. K. Elminir, and A. M. Riad, "Signs World Atlas; a benchmark Arabic sign language database," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 27, pp. 68–76, Jan. 2015, doi: 10.1016/j.jksuci.2014.03.011.

[4] M. Alfonse, A. Ali, A. S. Elons, N. L. Badr, and M. Aboul-Ela, "Arabic sign language benchmark database for different heterogeneous sensors," in *Proc. 5th Int. Conf. Inf. Commun. Technol. Accessibility (ICTA)*, Dec. 2016, pp. 1–9, doi: 10.1109/ICTA.2015.7426902.

[5] M. Mohandes, S. I. Quadri, and M. Deriche, "Arabic sign language recognition an image-based approach," in *Proc. 21st Int. Conf. Adv. Inf. Netw. Appl. Workshops (AINAW)* May 2007, pp. 272–276, doi: 10.1109/AINAW.2007.98.

[6] M. El Badawy, A. S. Elons, H. Sheded, and M. F. Tolba, "A proposed hybrid sensor architecture for Arabic Sign Language recognition," in *Intelligent Systems*. Cham, Switzerland: Springer, 2015, pp. 721–730.

[7] Z. Zafrulla, H. Brashear, T. Starner, H. Hamilton, and P. Presti, "American sign language recognition with the Kinect," in *Proc. 13th Int. Conf. Multimodal Interfaces*, 2011, pp. 279–286, doi: 10.1145/2070481.2070532.

[8] S. Aliyu, M. Mohandes, M. Deriche, and S. Badran, "Arabie sign language recognition using the Microsoft Kinect," in *Proc. 13th Int. Multi-Conf. Syst., Signals Devices (SSD)*, Mar. 2016, pp. 301–306, doi: 10.1109/SSD.2016.7473753.

[9] M. Ahmed, M. Idrees, Z. Ul Abideen, R. Mumtaz, and S. Khalique, "Deaf talk using 3D animated sign language: A sign language interpreter using Microsoft's Kinect v2," in *Proc. SAI Comput. Conf. (SAI)*, Jul. 2016, pp. 330–335, doi: 10.1109/SAI.2016.7556002.

[10] F. Soares, J. S. Esteves, V. Carvalho, C. Moreira, and P. Lourenco, "Sign language learning using the hangman videogame," in *Proc. Int. Congr. Ultra Mod. Telecommun. Control Syst. Workshops*, Oct. 2015, pp. 231–234, doi: 10.1109/ICUMT.2015.7382433.

[11] M. A. Almasre and H. Al-Nuaim, "Recognizing Arabic sign language gestures using depth sensors and a KSVM classifier," in *Proc. 8th Comput. Sci. Electron. Eng. Conf.*, Sep. 2017, pp. 146–151, doi: 10.1109/CEEC.2016.7835904.

[12] P. Kumar, H. Gauba, P. P. Roy, and D. P. Dogra, "A multimodal framework for sensor based sign language recognition," *Neurocomputing*, vol. 259, pp. 21–38, Oct. 2017, doi: 10.1016/j.neucom.2016.08.132.

[13] N. A. Sarhan, Y. El-Sonbaty, and S. M. Youssef, "HMM-based Arabic sign language recognition using Kinect," in *Proc. 10th Int. Conf. Digit. Inf. Manage. (ICDIM)*, Oct. 2015, pp. 169–174, doi: 10.1109/ICDIM.2015.7381873.

[14] 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 *Proc. 6th Int. Adv. Comput. Conf. (IACC)*, Feb. 2016, pp. 451–458, doi: 10.1109/IACC.2016.90.

[15] G. Devineau, F. Moutarde, W. Xi, and J. Yang, "Deep learning for hand gesture recognition on skeletal data," in *Proc. 13th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG)*, May 2018, pp. 106–113, doi: 10.1109/FG.2018.00025.

[16] F. Hu, P. He, S. Xu, Y. Li, and C. Zhang, "FingerTrak: Continuous 3D hand pose tracking by deep learning hand silhouettes captured by miniature thermal cameras on wrist," *Proc. ACM Interact., Mobile, Wearable Ubiquitous Technol.*, vol. 4, no. 2, pp. 1–24, Jun. 2020, doi: 10.1145/3397306.

[17] N. Tubaiz, T. Shanableh, and K. Assaleh, "Glove-based continuous Arabic sign language recognition in user-dependent mode," *IEEE Trans. Human-Mach. Syst.*, vol. 45, no. 4, pp. 526–533, Aug. 2015, doi: 10.1109/THMS.2015.2406692.

[18] B. Hisham and A. Hamouda, "Arabic static and dynamic gestures recognition using leap motion," *J. Comput. Sci.*, vol. 13, no. 8, pp. 337–354, Aug. 2017, doi: 10.3844/jcssp.2017.337.354.

[19] A. S. Elons, "GPU implementation for Arabic sign language real time recognition using multi-level multiplicative neural networks," in *Proc. 9th Int. Conf. Comput. Eng. Syst. (ICCES)*, Dec. 2014, pp. 360–367, doi: 10.1109/ICCES.2014.7030986.

[20] M. S. P. Kumar, V. Lathasree, and S. N. Karishma, "Novel contour based detection and GrabCut segmentation for sign language recognition," in *Proc. Int. Conf. Wireless Commun., Signal Process. Netw. (WiSPNET)*, Mar. 2017, pp. 738–742, doi: 10.1109/WiSPNET.2017.8299859.

[21] A. Mittal, P. Kumar, P. P. Roy, R. Balasubramanian, and B. B. Chaudhuri, "A modified LSTM model for continuous sign language recognition using leap motion," *IEEE Sensors J.*, vol. 19, no. 16, pp. 7056–7063, Aug. 2019, doi: 10.1109/JSEN.2019.2909837.

[22] S. Aly, B. Osman, W. Aly, and M. Saber, "Arabic sign language fingerspelling recognition from depth and intensity images," in *Proc. 12th Int. Comput. Eng. Conf. (ICENCO)*, Dec. 2016, pp. 99–104, doi: 10.1109/ICENCO.2016.7856452.

[23] B. Hisham and A. Hamouda, "Arabic dynamic gestures recognition using microsoft Kinect," *Sci. Visualizat.*, vol. 10, no. 5, pp. 140–159, Dec. 2018, doi: 10.26583/sv.10.5.09.

[24] N. C. Camgoz, O. Koller, S. Hadfield, and R. Bowden, "Sign language transformers: Joint end-to-end sign language recognition and translation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10023–10033, doi: 10.1109/cvpr42600.2020.01004.

[25] S. K. Liddell and M. Metzger, "Gesture in sign language discourse," *J. Pragmat.*, vol. 30, pp. 657–697, Dec. 1998, doi: 10.1016/s0378-2166(98)00061-7.

[26] D. Konstantinidis, K. Dimitropoulos, and P. Daras, "A deep learning approach for analyzing video and skeletal features in sign language recognition," in *Proc. IEEE Int. Conf. Imag. Syst. Techn. (IST)*, Oct. 2018, pp. 1–6, doi: 10.1109/IST.2018.8577085.

[27] C. Zhang, F. Bahmaninezhad, S. Ranjan, H. Dubey, W. Xia, and J. H. L. Hansen, "UTD-CRSS systems for 2018 NIST speaker recognition evaluation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 5776–5780, doi: 10.1109/ICASSP.2019.8683097.

[28] A. K. Sahoo, G. S. Mishra, and K. K. Ravulakollu, "Sign language recognition: State of the art," *ARPN J. Eng. Appl. Sci.*, vol. 9, no. 2, pp. 116–164, 2014.

[29] M. J. Cheok, Z. Omar, and M. H. Jaward, "A review of hand gesture and sign language recognition techniques," *Int. J. Mach. Learn. Cybern.*, vol. 10, pp. 131–153, Aug. 2019, doi: 10.1007/s13042-017-0705-5.

[30] B. L. Loeding, S. Sarkar, A. Parashar, and A. I. Karshmer, "Progress in automated computer recognition of sign language," in *Proc. Int. Conf. Comput. Handicapped Persons*. Berlin, Germany: Springer, Jul. 2004, pp. 1079–1087.

[31] S. Mitra and T. Acharya, "Gesture recognition: A survey," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 37, no. 3, pp. 311–324, May 2007, doi: 10.1109/TSMCC.2007.893280.

[32] M. Al-Hammadi, G. Muhammad, W. Abdul, M. Alsulaiman, M. A. Bencherif, and M. A. Mekhtiche, "Hand gesture recognition for sign language using 3DCNN," *IEEE Access*, vol. 8, pp. 79491–79509, 2020.

[33] M. M. Kamruzzaman, "Arabic sign language recognition and generating Arabic speech using convolutional neural network," *Wirel. Commun. Mob. Comput.*, vol. 2020, May 2020, Art. no. 3685614.
[34] M. ElBadawy, A. S. Elons, H. A. Shedeed, and M. F. Tolba, "Arabic sign language recognition with 3D convolutional neural networks," in *Proc. 8th Int. Conf. Intell. Comput. Inf. Syst. (ICICIS)*, Dec. 2017, pp. 66–71.
[35] G. Latif, N. Mohammad, R. AlKhalaf, R. AlKhalaf, J. Alghazo, and M. Khan, "An automatic Arabic sign language recognition system based on deep CNN: An assistive system for the deaf and hard of hearing," *Int. J. Comput. Digit. Syst.*, vol. 9, no. 4, pp. 715–724, 2020.
[36] Y. Saleh and G. F. Issa, "Arabic sign language recognition through deep neural networks fine-tuning," *Int. J. Online Biomed. Eng.*, vol. 16, no. 5, pp. 71–83, 2020.
[37] S. Aly and W. Aly, "DeepArSLR: A novel signer-independent deep learning framework for isolated Arabic sign language gestures recognition," *IEEE Access*, vol. 8, pp. 83199–83212, 2020.
[38] G. D. Fathy, E. Emary, and H. N. ElMahdy, "Supporting Arabic Sign Language recognition with facial expressions," in *Proc. 7th Int. Conf. Inf. Technol. (ICIT)*, 2015, pp. 164–170. [Online]. Available: http://icit.zuj.edu.jo/ICIT15, doi: 10.15849/icit.2015.0024.
[39] N. B. Ibrahim, M. M. Selim, and H. H. Zayed, "An automatic Arabic sign language recognition system (ArSLRS)," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 30, no. 4, pp. 470–477, Oct. 2018, doi: 10.1016/j.jksuci.2017.09.007.
[40] A. A. I. Sidig, H. Luqman, and S. A. Mahmoud, "Transform-based Arabic sign language recognition," *Procedia Comput. Sci.*, vol. 117, pp. 2–9, Nov. 2017.
[41] M. Hassan, K. Assaleh, and T. Shanableh, "Multiple proposals for continuous Arabic sign language recognition," *Sens. Imag.*, vol. 20, no. 1, p. 4, 2019, doi: 10.1007/s11220-019-0225-3.
[42] M. Alsulaiman, G. Muhammad, M. A. Bencherif, A. Mahmood, and Z. Ali, "KSU rich Arabic speech database," *Information*, vol. 16, no. 6, pp. 4231–4253, 2013.
[43] M. M. Alsulaiman, G. Muhammad, M. A. Bencherif, A. Mahmood, Z. Ali, and M. Aljabri, "Building a rich Arabic speech database," in *Proc. 5th Asia Modelling Symp.*, May 2011, pp. 100–105, doi: 10.1109/AMS.2011.29.
[44] *King Saud University Arabic Speech Database—Linguistic Data Consortium*. Accessed: Sep. 11, 2020. [Online]. Available: https://catalog.ldc.upenn.edu/LDC2014S02
[45] A. Al-Nasheri, G. Muhammad, M. Alsulaiman, and Z. Ali, "Investigation of voice pathology detection and classification on different frequency regions using correlation functions," *J. Voice*, vol. 31, no. 1, pp. 3–15, 2017, doi: 10.1016/j.jvoice.2016.01.014.
[46] T. A. Mesallam, M. Farahat, K. H. Malki, M. Alsulaiman, Z. Ali, A. Al-Nasheri, and G. Muhammad, "Development of the Arabic voice pathology database and its evaluation by using speech features and machine learning algorithms," *J. Healthcare Eng.*, vol. 2017, Oct. 2017, Art. no. 8783751, doi: 10.1155/2017/8783751.
[47] P. Dreuw, C. Neidle, V. Athitsos, S. Sclaroff, and H. Ney, "Benchmark databases for video-based automatic sign language recognition," in *Proc. 6th Int. Conf. Lang. Resource Eval. (LREC)*. Marrakech, Morocco: European Language Resources Association, 2008.
[48] D. Avola, M. Bernardi, L. Cinque, G. L. Foresti, and C. Massaroni, "Exploiting recurrent neural networks and leap motion controller for the recognition of sign language and semaphoric hand gestures," *IEEE Trans. Multimedia*, vol. 21, no. 1, pp. 234–245, Jan. 2019, doi: 10.1109/TMM.2018.2856094.
[49] *The Unified Arabic Sign Language Dictionary for the Deaf*. Accessed: Sep. 26, 2020. [Online]. Available: https://menasy.com/arabDictionaryforthedeaf2.pdf
[50] T. Guzsvinecz, V. Szucs, and C. Sik-Lanyi, "Suitability of the Kinect sensor and leap motion controller—A literature review," *Sensors*, vol. 19, no. 5, p. 1072, 2019, doi: 10.3390/s19051072.
[51] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "OpenPose: Realtime multi-person 2D pose estimation using part affinity fields," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 1, pp. 172–186, Jan. 2021, doi: 10.1109/tpami.2019.2929257.
[52] P. Molchanov, S. Gupta, K. Kim, and J. Kautz, "Hand gesture recognition with 3D convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2015, pp. 1–7, doi: 10.1109/CVPRW.2015.7301342.
[53] Q. De Smedt, H. Wannous, and J.-P. Vandeborre, "Skeleton-based dynamic hand gesture recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2016, pp. 1206–1214, doi: 10.1109/CVPRW.2016.153.
[54] C. Goutte and E. Gaussier, "A probabilistic interpretation of precision, recall and F-score, with implication for evaluation," in *Proc. Eur. Conf. Inf. Retr.* Berlin, Germany: Springer, Mar. 2005, pp. 345–359.
[55] X. Zhu, Y. Xiong, J. Dai, L. Yuan, and Y. Wei, "Deep feature flow for video recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2349–2358, doi: 10.1109/CVPR.2017.441.
[56] R. Bruyer and M. Brysbaert, "Combining speed and accuracy in cognitive psychology: Is the inverse efficiency score (IES) a better dependent variable than the mean reaction time (RT) and the percentage of errors (PE)?" *Psychologica Belgica*, vol. 51, no. 1, p. 5, Feb. 2011, doi: 10.5334/pb-51-1-5.
[57] R. Cui, G. Hua, A. Zhu, J. Wu, and H. Liu, "Hard sample mining and learning for skeleton-based human action recognition and identification," *IEEE Access*, vol. 7, pp. 8245–8257, 2019, doi: 10.1109/ACCESS.2018.2889797.
[58] M. Al-Hammadi, G. Muhammad, W. Abdul, M. Alsulaiman, M. A. Bencherif, T. S. Alrayes, H. Mathkour, and M. A. Mekhtiche, "Deep learning-based approach for sign language gesture recognition with efficient hand gesture representation," *IEEE Access*, vol. 8, pp. 192527–192542, 2020, doi: 10.1109/access.2020.3032140.

MOHAMED A. BENCHERIF received the Engineering degree in control from INELEC, Boumerdes, Algeria, in 1992, and the master's degree in signals and systems and the Ph.D. degree in the classification of remote sensing images from Saad Dahleb University, Blida, Algeria, in 2005 and 2015, respectively. He worked in diverse industrial projects in project planning and management. He is currently working with the Center of Smart Robotics Research, King Saud University. His research interests include artificial intelligence, robotics, speech classification, sign language recognition, cloud computing, and hardware design.

MOHAMMED ALGABRI received the master's degree from King Saud University, where he is a currently pursuing the Ph.D. degree with the Computer Science Department, College of Computer and Information Sciences. His research interests include speech processing, pronunciation error detection, deep learning, and soft computing techniques.

MOHAMED A. MEKHTICHE was born in Medea, Algeria, in 1987. He received the B.S. and M.S. degrees in electronic engineering from the University of Blida, in 2010 and 2012, respectively.

From 2014 to present, he worked as a Researcher with the Center of Smart Robotic Research, King Saud University, Saudi Arabia. His current research interests include image processing, robot and drone design and implementation, autonomous navigation, and humanoid robotics.

MOHAMMED FAISAL received the master's and Ph.D. degrees from King Saud University, in 2012 and 2016, respectively. He currently works as an Assistant Professor and supervises the unit of Innovation and Entrepreneurship, College of Applied Computer Science, King Saud University, where he is a Robotics Consultant with the Center of Smart Robotics Research. In recent years, he has published scores of scientific research in refereed and classified scientific journals and conferences. His current research interest includes the use of robots and artificial intelligence to improve the quality of life and find innovative solutions. He won the President of the Republic of Yemen Youth Award for the Applied Sciences Branch for the year 2013 and many scientific and research awards and medals.

MANSOUR ALSULAIMAN received the Ph.D. degree from Iowa State University, USA, in 1987. Since 1988, he has been with the Computer Engineering Department, King Saud University, Riyadh, Saudi Arabia, where he is currently a Professor with the Department of Computer Engineering and the Director of the Center of Smart Robotics Research. His research interests include automatic speech/speaker recognition, automatic voice pathology assessment systems, computer-aided pronunciation training systems, and robotics. He was the Editor-in-Chief of the *Journal of King Saud University—Computer and Information Sciences* Section.

HASSAN MATHKOUR received the Ph.D. degree from The University of Iowa, USA. He is a currently a Professor with the Department of Computer Science, College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia. He held several administration posts, including the Dean, the Associate Dean, the Department Chair, the Director of the Research Center, and the Head of the joint Ph.D. Program. He also serves as an IT Consultant. He has more than 100 research papers in journals and conferences. His research interests include intelligent systems, peer-to-peer systems, modeling and analysis, database management systems, data mining, knowledge management, e-learning, and bioinformatics.

MUNEER AL-HAMMADI (Member, IEEE) received the Ph.D. degree from the Department of Computer Engineering, College of Computer and Information Sciences (CCIS), King Saud University (KSU), Riyadh, Saudi Arabia. He is currently with the Center of Smart Robotics Research, CCIS, KSU. His research interests include image and video processing, and deep learning. He was a recipient of the Best Graduate Student Research Award from KSU.

HAMID GHALEB received the master's degree in software engineering from King Saud University, Saudi Arabia, where he is currently pursuing the Ph.D. degree with the Software Engineering Department, College of Computer and Information Science. He is also working as a Researcher with the Center of Smart Robotics Research, King Saud University. His research interests include recommender systems, robotics, software engineering, and project management.