IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 28 July 2022, accepted 6 August 2022, date of publication 5 September 2022, date of current version 12 September 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3204110

RESEARCH ARTICLE

An Efficient Two-Stream Network for Isolated Sign Language Recognition Using Accumulative Video Motion

HAMZAH LUQMAN
Information and Computer Science Department, College of Computing and Mathematics, King Fahd University of Petroleum & Minerals, Dhahran 31261, Saudi Arabia
SDAIA-KFUPM Joint Research Center for Artificial Intelligence, KFUPM, Dhahran 31261, Saudi Arabia
e-mail: hluqman@kfupm.edu.sa

This work was supported in part by the Saudi Data and AI Authority (SDAIA), and in part by the King Fahd University of Petroleum & Minerals (KFUPM) through the SDAIA-KFUPM Joint Research Center for Artificial Intelligence under Grant JRC-AI-RFP-05.

ABSTRACT Sign language is the primary communication medium for persons with hearing impairments. This language depends mainly on hand articulations accompanied by nonmanual gestures. Recently, there has been a growing interest in sign language recognition. In this paper, we propose a trainable deep learning network for isolated sign language recognition, which can effectively capture the spatiotemporal information using a small number of signs' frames. We propose a hierarchical sign learning module that comprises three networks: dynamic motion network (DMN), accumulative motion network (AMN), and sign recognition network (SRN). Additionally, we propose a technique to extract key postures for handling the variations in the sign samples performed by different signers. The DMN stream uses these key postures to learn the spatiotemporal information pertaining to the signs. We also propose a novel technique to represent the statical and dynamic information of sign gestures into a single frame. This approach preserves the spatial and temporal information of the sign by fusing the sign's key postures in the forward and backward directions to generate an accumulative video motion frame. This frame was used as an input to the AMN stream, and the extracted features were fused with the DMN features to be fed into the SRN for the learning and classification of signs. The proposed approach is efficient for isolated sign language recognition, especially for recognizing static signs. We evaluated this approach on the KArSL-190 and KArSL-502 Arabic sign language datasets, and the obtained results on KArSL-190 outperformed other techniques by 15% in the signer-independent mode. Additionally, the proposed approach outperformed the state-of-the-art techniques on the Argentinian sign language dataset LSA64. The code is available at https://github.com/Hamzah-Luqman/SLR_AMN.

INDEX TERMS Sign language recognition, Arabic sign language, Argentinian sign language, KArSL, LSA64, gesture recognition, action recognition.

I. INTRODUCTION

Hearing loss has become a common problem globally. According to the World Health Organization [1], 2.5 billion people (i.e., one in four persons) are projected to have some degree of hearing loss by 2050; approximately 700 million of these will need hearing rehabilitation. This increases the dependence on sign language, which is the primary communication language for persons with various levels of hearing disabilities.

Sign languages are complete languages with their own grammar and syntax; however, their linguistic properties differ from those of natural languages [2]. Each sign language has its own dictionary that is usually limited in size in comparison with the dictionaries of natural languages. Consequently, signers use one sign to refer to several spoken synonymous words, such as home, house, and apartment. Similar to spoken languages, sign languages are diverse; several sign languages

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

are used worldwide, such as the American sign language (ASL), Chinese sign language (CSL), and Arabic sign language (ArSL) [3]. ArSL is one of the sign languages used in Arabic countries. This is a unified language of several sign languages used in Arabic countries [4]. ArSL was proposed in 1999 by the League of the Arab States and the Arab League Educational, Cultural and Scientific Organization. The ArSL dictionary consisting of 3200 sign words was published in two parts in 2000 [5] and 2006 [6]. The signs for the dictionary were mainly selected by finding the shared signs among the majority of the sign languages in Arab countries and in Arab Gulf countries. ArSL is mainly used in the Arab states of the Arab Gulf countries (e.g., Qatar and the United Arab Emirates). Further, in almost all Arab countries, ArSL is used at airports and by the news media, such as Al-Jazeera. The correlation between signs and spoken languages is complex and varies depending on the country more than the spoken language. Therefore, countries that share the same spoken language may have different sign languages, for example, although English is the spoken language of both the United Kingdom and the United States, they have different sign languages, namely, British sign language and ASL, respectively [7].

Sign language is a descriptive language that simultaneously utilizes manual and nonmanual gestures [8]. A majority of the sign words depend on manual movements that use hand motions for interpersonal communication [9]. These signs are accompanied usually by nonmanual gestures that consist of body postures and facial expressions. Nonmanual gestures play an important role in many sign languages to convey emotions and linguistic information that cannot be expressed by manual gestures. For example, facial expressions are used to express negations in ArSL, and they serve as adverbs and adjectives that modify manual signs [10]. Additionally, facial expressions are used to distinguish between signs that share the same manual gesture. For instance, the sign for ''brother'' is identical to the sign for ''sister'' in German sign language; however, their lip patterns differ [11].

Motion represents a basic component of sign gestures. Based on the motions involved, signs can be classified into two types: static and dynamic signs. Most of the sign language letters and digits are static signs in which the signs do not involve any motion. These signs depend mainly on the shapes and orientations of the hands and fingers [12]. Therefore, still images can adequately capture these types of signs, which justify the availability of most of the alphabet datasets in the form of images. By contrast, dynamic signs involve manual and/or nonmanual motions of body parts. These signs represent a majority of the sign words used in the sign language vocabulary [13]. Hence, a video stream is required to represent signs in which the motion component is basic.

Sign language interpretation involves recognition and translation. Recognition is the task of identifying sign gestures either in the images or videos of a sign language and returning their equivalent in a natural language. The output of this stage can be isolated words or sentences depending on the input provided. Isolated sign recognition systems accept a sign and output an equivalent word in a spoken language. Continuous sign language recognition systems identify a sequence of signs performed continuously and output a set of words in the form of sentences. These sentences have the structure and grammar of the source sign language, which are usually different from the structure and grammar of natural languages. Thus, a machine translation system is used to translate these sentences into the target natural language.

Several approaches have been proposed recently for sign language recognition. However, there are still some limitations that need to be addressed. Firstly, most of the sign recognition systems consider all signs' frames for sign learning and classification. This can result in degrading the recognition accuracy due to the variations between the signs performed by different signers. Therefore, there is a need for an approach to extract the main postures of the sign gesture and ignore less important postures. Secondly, most of the temporal learning techniques for dynamic sign gesture recognition could not learn the non-manual gestures efficiently. Thirdly, few techniques have been proposed for ArSL recognition compared with other sign languages. This can be attributed mainly to the lack of a benchmarked dataset. We aim in this work to address these limitations. The main contributions of this research are as follows:

- We propose a trainable deep learning network for sign language recognition that can effectively capture the spatiotemporal information with few frames of the signs
- We design a hierarchical sign learning model, which learns the spatial and temporal information of the sign gesture in three networks: dynamic motion network (DMN), accumulative motion network (AMN), and sign recognition network (SRN).
- We propose a technique to extract the dominant and important sign postures. This approach helps tackle the variations of the sign samples.
- We propose an accumulative video motion (AVM) technique to encode the sign motions in the video stream into a single image.
- We evaluated the proposed approach on the KArSL and LSA64 datasets and found that the proposed method outperformed other methods.
- We benchmarked the KArSL-502 dataset.

The rest of the paper is organized as follows. Section II reviews the related work dealing with sign language recognition. Section III presents the architecture of the proposed system, and Section IV describes the experimental work and the obtained results. Section V highlights the contributions of this research and concludes the paper.

II. RELATED WORK

Several techniques have been proposed in the last two decades for automatically recognizing various sign languages. Based on the acquisition devices, these techniques can be classified into two types: sensor-based and

vision-based techniques [14]. Most of the early recognition techniques depended on motion sensors for the detection and tracking of hand movements; however, recent techniques have abandoned these sensors, and sign captioning is facilitated by applying cameras.

A. SENSOR-BASED TECHNIQUES

Sensor-based techniques use motion sensors to acquire sign gestures [15]. These sensors can track the movements and shapes of fingers and hands. Electronic gloves are the most commonly used sensors in literature [16], [17], [18], [19], [20], [21], [22], [23]. Ritchings *et al.* [22] proposed a sign language recognition system using two bend sensors with push-button switches for motion tracking. This system was evaluated on 65 signs and an accuracy of 93% was reported. However, this system failed to track signs when the signers had small hands.

Dempster–Shafer Theory of Evidence was used by Mohandes and Deriche [23] to integrate the data obtained from a hand-tracking system and a glove sensor. A dataset consisting of 100 signs was collected using these sensors, and these signs were used to evaluate the proposed system. Twenty-eight signals were provided by both sensors for each hand. The accuracy reported using the hand-tracking system was 84.7%, whereas a better accuracy of 91.3% was obtained using the glove-based system. The authors found that data fusion at the classification level had a higher accuracy of 98.1% when compared with the data fusion accuracy of 96.2% at the features level. The glove-based system using a pair of DG5-VHand gloves was developed by Tubaiz *et al.* [24]. The information retrieved by these sensors was fed into the modified K-nearest neighbor for classification, and an accuracy of 98.9% was reported for a dataset comprising 40 sentences.

The sensor-based acquisition techniques helped in the tracking of hands and in handling environmental constraints, such as background removal; however, requiring the signer to wear gloves during the signing was difficult for real-time systems. Also, these sensors could not capture the nonmanual gestures that are a basic component of any sign language. Thus, a majority of the recent sign language recognition systems are vision-based where single or multiple camera devices are used for sign capturing. These systems require only video cameras for sign acquisition, which helps to integrate these systems easily using new technologies such as smartphones. Additionally, new cameras can provide depth information that helps to obtain more information about the performed signs.

B. VISION-BASED TECHNIQUES

Vision-based techniques can be broadly classified into classic and machine learning techniques. Classic techniques depend on extracting statistical and geometric features from sign gestures and feeding them into a classifier. Nai *et al.* [25] proposed a system to recognize ASL digits on depth images. A set of statistical features were extracted and classified using the random forest classifier, and an accuracy of 89.6% was reported. Depth images were also used by Ameida *et al.* [26] for Brazilian sign language recognition. A set of seven structural features were extracted from these images and fed into support vector machines (SVM) for classification to obtain an accuracy above 80% with 34 signs. Joshi *et al.* [27] applied a multilevel histogram of gradient (HOG) for recognizing Indian sign language letters in complex backgrounds. An accuracy of 92% has been reported on a dataset that consists of 26 signs. Nevertheless, this accuracy is low given the dataset size and the hand segmentation.

Hidden Markov model (HMM) was used by Zaki and Shaheen [28] for recognizing 50 signs of ASL. The principal component analysis (PCA) was used for features reduction and an accuracy of 89.1% was reported. A PCA with linear discriminant analysis was also used by Pan *et al.* [29]. The extracted features were classified using SVM, and the accuracies of 94% and 99.8% were reported using the 26 signs of ASL and CSL, respectively. Nguyen and Do [30] used HOG and local binary pattern (LBP) techniques for features extraction and SVM for classification.

Several frequency domain features have been used in literature for sign language recognition. The main frequency-based techniques used in the literature are dynamic time warping [31], [32], Fourier descriptors [3], [29], Hu moments [29], discrete wavelet transform [33], [34], and wavelet transform [35]. Makhashen *et al.* [7] used the Gabor transform for features extraction. These features were fed into a convolutional neural network (CNN), and an accuracy higher than 95% was reported for the ASL letters. However, Zernike moments outperformed Hu moments, PCA, and Fourier descriptors for sign language recognition [36], [37], [38].

Machine learning techniques have been extensively applied for sign language recognition during the last 10 years. These techniques can be classified into traditional machine learning techniques and deep learning techniques. Traditional machine learning techniques apply classical algorithms for sign language recognition, such as SVM, PCA, and HMM. These techniques have been used with different input representations including sensor-based features. Other researchers combined them with deep learning models, such as CNN and long short-term memory (LSTM). CNN was used by Jiang and Zhang [39] for CSL recognition, and an accuracy of 88.1% was reported using 1260 RGB images. Similarly, Barbhuiya *et al.* [40] proposed a CNN–SVM model for sign language recognition. The proposed model utilizes the pre-trained AlexNet and VGG16 models. The proposed model was used for features extraction, and the SVM was used for classification. This approach reported an accuracy of 99.8% on the static signs of ASL. For more details about the deep learning techniques utilized for sign language recognition, we refer to this survey [41].

Liu *et al.* [42] used the joint points of the signer's hand skeleton as the input to the LSTM model. Twelve joint points were obtained using the Kinect sensor and fed into LSTM with seven layers. This technique was evaluated on the two

CSL datasets with 25K and 125K images with reported accuracies of 86% and 64%, respectively. Sidig and Mahmoud [43] also used hand trajectories for ArSL recognition. The skeletons of the signer's hands were tracked using the Kinect camera, and the captured joint points were fed into KNN for classification. This technique was evaluated on 100 signs of the KArSL dataset, and accuracies of 99% and 64% were reported for signer-dependent and signer-independent modes, respectively.

Multimodality systems have been proposed by several researchers for sign language recognition. A combination of joint points with color and depth images was used by Huang *et al.* [44] as an input to a CNN model with eight layers. This model was evaluated on a dataset comprising 25 sign gestures, and an accuracy of 94.2% was reported. Color and depth images were also concatenated by Li *et al.* [45] and used for recognizing 24 letters of ASL. A sparse autoencoder model with CNN was used to extract the features from the color and depth images to be classified by SVM. This approach reported an accuracy of 99.05% on the dataset comprising 20K samples.

Although most of the surveyed studies on sign language recognition considered only manual gestures, several researchers have studied the importance of nonmanual features for sign language recognition. Luqman and El-Alfy [13] combined the facial expressions with hand gesture features; this fusion improved the accuracy by 3.6% when compared with the case when only manual gestures were performed in the signer-independent mode by four signers using 50 ArSL signs. The fusion of manual gestures with facial expressions was also performed by Sabyrov *et al.* [46]. The OpenPose library was used to extract the keypoints from the facial expressions and hand gestures. This hybrid system boosted the recognition accuracy by 7% when compared with only hand gestures. Kumar *et al.* [47] used Kinect with a Leap Motion Controller to capture facial expressions and hand gestures. Each of these modalities was separately fed into the HMM to be combined later at the classification level using the Bayesian classifier. An Indian sign language dataset with 51 signs was used to evaluate this approach, and the reported results showed that the fusion of the manual and nonmanual gestures improved the accuracy by 1.04% when compared with a single modality. Table 1 shows the summary of the surveyed papers.

III. PROPOSED METHOD

The objective of this research is to develop a trainable deep learning network for sign language recognition that can effectively capture the spatiotemporal information with few frames. To this end, we propose a sign recognition system that consists of a key postures extractor and sign learning networks, as shown in Figure 1. The key postures extractor is used to capture the main postures of the sign gesture by extracting the key frames in the sign video stream. We also propose the AVM technique to capture the motion of the signs' frames and represent the motion using a single image while preserving the spatiotemporal information of the sign gesture. The key postures and AVM frame are fed into a novel two-stream network for sign language recognition. The key postures are fed into the DMN to learn the spatiotemporal information in the sign gesture. The AVM frame is used as an input to the AMN that learns the motion in the AVM image. The extracted features from the two streams are concatenated and fed into the SRN for learning the fused features and performing the classification. In this section, we start by describing the key frames and AVM extraction techniques. Then, we discuss the proposed two-stream sign learning architecture and the fusion technique.

A. KEY POSTURES

Based on body motion, sign gestures can be classified into two types: static and dynamic. Static signs are motionless gestures, and they depend mainly on the shape, orientation, and articulation of the hands and fingers to convey meanings. By contrast, dynamic signs employ body movements during signing. Dynamic gestures represent a majority of signs used in sign languages, whereas static gestures are used mainly for letters, digits, and a few sign words.

Dynamic gestures are more challenging to recognize than static gestures. The recognition of static gestures depends only on spatial information, whereas the recognition of dynamic gestures requires spatial and temporal information. An additional challenge for recognizing such signs is the gesture variations among the different signers of the sign. These variations are obvious with signs that consist of more than one posture. Another challenge with the recognition of dynamic gestures is the large number of generated frames, especially when sign gestures are recorded at high frame rates. Some of these frames are often redundant, which increases the recognition time of the systems that process sign video frames for recognizing sign gestures. To address these problems, we extracted the key frames from each sign and used these frames as the input to the recognition system.

A key posture technique is used to extract the prominent sign postures in the sign video stream by extracting the corresponding frames in the sign's video stream. Inspired by [43], we extracted the key frames by employing the hand trajectories captured by tracking the hand joint points, which were returned by Kinect as part of the skeleton data. The points for the hand's joints can have some outliers that can significantly impact the extraction of key postures. To address this problem, we preprocessed these joint points by smoothing the hand locations using the median filter to remove the outliers. For occluded hands or lost joints, Kinect V2 is efficient in joint estimation while providing skeletal tracking that is more robust to occlusions [48]. However, if this estimation is noisy or inaccurate, our median filter will smooth it in the preprocessing stage. Then, we extracted the key frames by connecting the hand locations during signing to form a polygon, as shown in Figure 2.

The sharp changes in hand locations represent the vertices of the polygon. To keep the most important N vertices,

TABLE 1. Summary of the surveyed papers.

Paper	Sign language	Features	Classifier	Number of Signs	Accuracy
Nai et al. [25]	American			10	89.6%
Zaki et al. [28]	American			50	89.1%
Pan et al. [29]	American			26	94.0%
	Chines				99.8%
Makhashen et al. [7]	American	Gabor, CNN	SVM	26	95.0%
Nguyen et al. [30]		HOG-LBP	SVM	24	
Jiang and Zhang [40]	Chinese	CNN			88.1%
Barbhuiya et al. [41]	American	CNN	SVM	26	99.8%
Sidig and Mahmoud [44]	Arabic	Hands Trajectory	KNN	100	64.0%
Huang et al. [45]	American	Hands Trajectory	CNN	25	94.2%
Li at al. [46]	American			24	99.1%
Luqman and Alfy [13]	Arabic	CNN-LSTM		50	72.4%
Liu et al. [43]	Chinese			500	63.3%
Sabyrov et al. [47]	Kazakh-Russian			20	73.0%
Kumar et al. [48]	Indian			51	94.3%

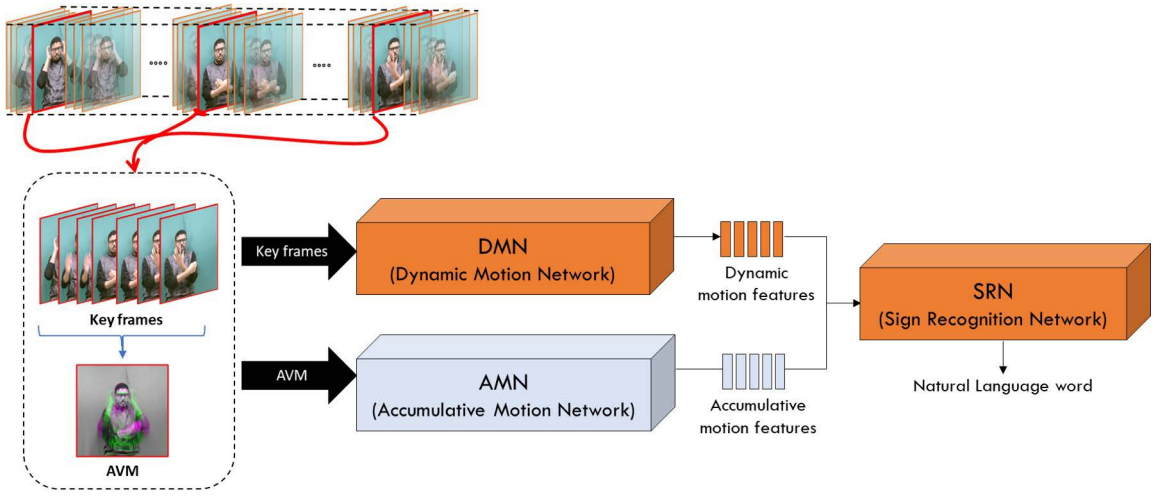

FIGURE 1. Framework of the sign recognition system.

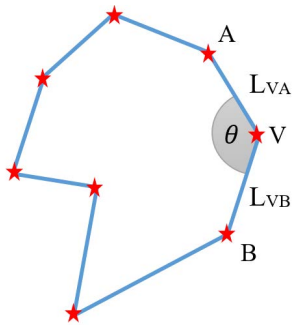

FIGURE 2. An illustration of how hand locations are connected to form a polygon to compute the importance of the vertices.

we applied a polygon approximation algorithm. This algorithm measures the importance of each vertex by taking the product of its edge's lengths and the angle between the edges of this vertex. As shown in Figure 2, the importance of the vertex V is computed as follows:

$$V_{importance} = L_{AV} \times L_{VB} \times \Theta \tag{1}$$

where L_{AV} and L_{VB} are the lengths from the vertex V to the vertices A and B, respectively, whereas θ is the angle between the vertex V and the two adjacent segments. The process is applied to all polygonal vertices, and the least important vertex is removed. This reduction algorithm is iteratively repeated to recompute the importance of the remaining vertices until N vertices remain, as shown in Figure 3; this figure shows the raw hand trajectory and the trajectory obtained after applying the algorithm. This algorithm was applied to all the color videos to extract N key postures.

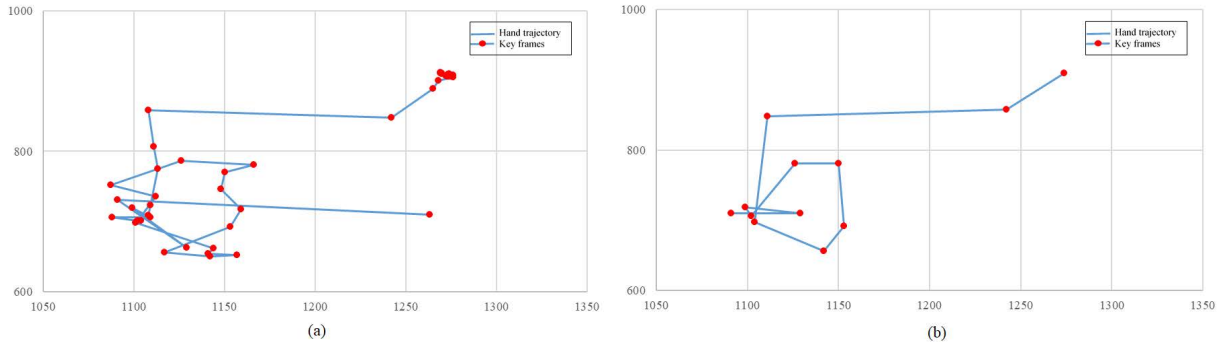

FIGURE 3. Result of applying the key postures algorithm to the "First aid" sign of ArSL. The X and Y axes represent the coordinates of the hand joint points projected to the color frames for (a) a raw hand trajectory and (b) a trajectory obtained after smoothing and applying the algorithm for extracting the key postures.

B. ACCUMULATIVE VIDEO MOTION

Motion is a primary component of dynamic sign gestures that represent a majority of the signs in the sign language dictionary. Encoding this motion into one still image helps in using spatial features extraction techniques to learn signs. Additionally, encoding helps overcome the problem of misclassifying static signs that do not include motion. These signs differ based on the shapes and orientations of the hands and fingers. These variations cannot be captured easily by time-series techniques, such as the LSTM model. To address this challenge, we propose the AVM technique that encodes the sign with its motion into a single image. This approach is inspired by the accumulative temporal difference (ATD) technique proposed by Shanableh *et al.* [49]. ATD represents the sign video as a single binary still image using the thresholded accumulated difference between consecutive frames. In contrast, the AVM approach proposed in this work utilizes the accumulated summation between the sign's frames and produces an RGB image representing the whole sign. In addition, our approach preserves the spatial information between frames even if there is no motion in the sign, whereas the ATD technique preserves only the motion and removes the static features between frames, which makes it inefficient for recognizing static gestures.

This technique creates a bidirectional composite image (Bi-AVM) by fusing the key frames in the forward and backward directions, as follows:

$$\text{Bi-AVM} = \sum_{i=1}^{KP} KeyFrame_i + \sum_{i=KP}^{1} KeyFrame_i \qquad (2)$$

where KP is the number of key frames that correspond to the number of key postures. The forward AVM (FWD-AVM) fusion creates a composite image by fusing the images starting from the first frame till the last frame. The backward AVM (BWD-AVM) fusion starts the fusion in the reverse order from the last key frame. Figure 4 shows a sample of the AVM image.

C. SIGN LEARNING SYSTEM

The proposed system consists of three networks for sign recognition as shown in Figure 1. The first network, DMN, learns the spatiotemporal information on the key frames of the sign gesture. The AMN stream accepts the AVM image as an input to learn the spatial information of this image. The outputs of both streams are fused at the features level and fed into the SRN model for learning and classification.

1) DYNAMIC MOTION NETWORK

Sign language recognition is a time-series problem that depends on two sources of information for each sign gesture: spatial and temporal. The spatial information represents the sign using fingers, hands, and body shapes and rotations. The temporal information represents the motion used by all the dynamic signs. Motion is a primary component in sign language, and it involves changing the position and/or shape of the hands during gesturing.

To learn and extract the spatial and temporal information from the key frame of the sign gesture, a combination of CNN and LSTM is applied. Figure 5 shows the architecture of the proposed network. CNN has been extensively employed for several pattern recognition problems, and its efficiency in extracting the spatial features is well established. We fine-tuned four pre-trained models (viz., VGG16 [50], Xception [51], ResNet152V2 [52], and MobileNet [53]) for extracting the spatial information from each key frame. These three models have been trained on ImageNet for large-scale image classification with 14,197,122 images and 21,841 subcategories. Although these models have been trained on the same dataset, the specifications and structure of the models made them fit well for different pattern recognition problems in the literature.

As shown in Figure 5, the extracted features using the pre-trained models are fed into a stacked LSTM. The LSTM consists of two LSTM layers with 2048 neurons each. The output of these layers is fed into a fully connected layer with 1024 neurons followed by a rectified linear (ReLU) activation function. This function handles the nonlinearity by zeroing the negative values. This function is computationally powerful and helps to reduce the possibility of gradient vanishing [54]. To reduce the overfitting, a dropout layer of 60% is used after the activation function. For classification, a Softmax layer is added as the last layer in the DMN stream to

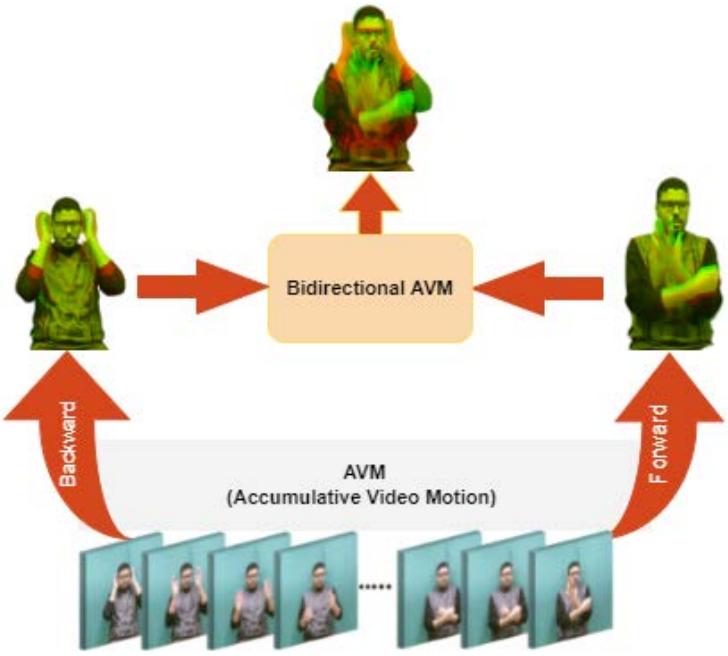

FIGURE 4. An example of accumulative video motion fusion approach (the frame background has been removed for clarity).

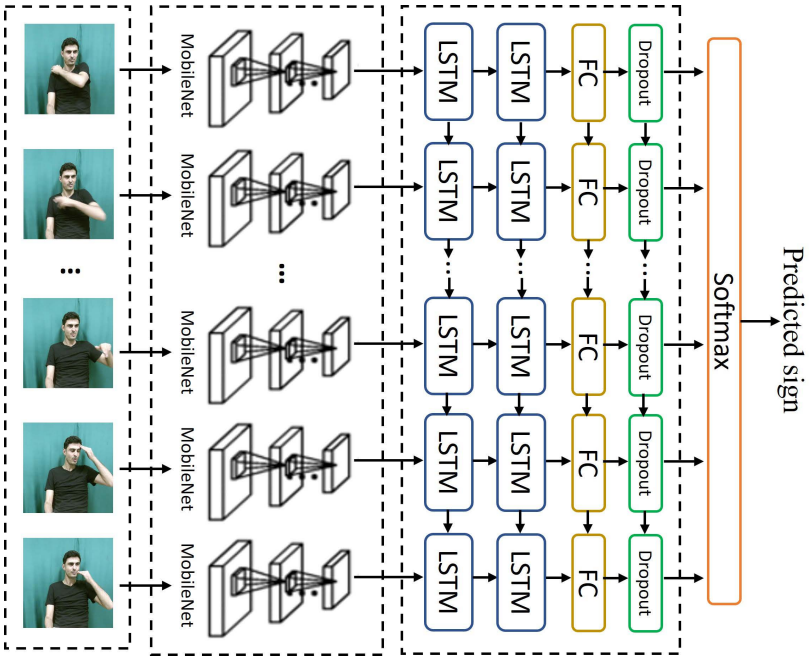

FIGURE 5. Framework of dynamic motion network model.

assign a probability value to each predicted sign. Therefore, the number of neurons in this layer matches the number of signs in the dataset. We also used the cross-entropy loss function during the model training.

2) ACCUMULATIVE MOTION NETWORK

Recognizing sign gestures depends on how the spatial and temporal information of the sign gestures is recognized. As discussed in Section III-B, we represented the motion of the sign in a single image using the AVM approach. This image encodes spatial and temporal information. It also helps to recognize the static sign gestures that do not involve motion. These signs are a challenge for DMN because the variations between some static signs are at the level of finger shapes, which cannot be captured easily by the DMN networks.

AMN learns the signs represented by AVM. AVM fusion is performed in three ways: forward, backward, and bidirectional, as discussed in Section III-B. Each AVM is fed into the AMN that uses a CNN network fine-tuned on a pre-trained MobileNet network. This network is used to extract 1024 features from each AVM image by applying global average pooling to the output of the last layer before the classification layer of the MobileNet network. These features are fed into a dropout layer with 50% probability and the output of this layer is fed into the classification layer.

3) SIGN RECOGNITION NETWORK

The dynamic and accumulative motion features extracted from DMN and AMN, respectively, are fused and fed into the SRN stream. These features are concatenated to form one vector which is used as an input to the SRN stream as shown in Figure 1. SRN is a convolutional network that consists of four stacked layers. The first layer is a batch normalization layer that is used to normalize the input features to this layer and consequently reduce the model training time [55]. In addition, it helps in addressing the internal covariate shift problem that results from the distribution change of the network activations during the model training [56].

The output of the batch normalization layer is fed into a convolutional layer. This layer utilizes 256 neurons for learning with a kernel size of seven, selected empirically. To handle the nonlinearity of the features extracted using the convolutional layer, we employed a ReLU activation function. The resulting output of this layer is fed into a dropout layer with a probability of 60% selected empirically. This layer helps in reducing the possibility of overfitting. The last layer is the classification layer that uses a Softmax classifier with the number of neurons equal to the number of signs. The whole recognition model is trained with a cross-entropy loss function and an Adam optimizer with a learning rate of 10^{-4}, which was selected empirically.

IV. EXPERIMENTAL WORK

A. DATASETS

We evaluated our approach using two datasets, namely, KArSL [57] and LSA64 [58], which are the Arabic and Argentinian sign language datasets, respectively. KArSL is a multimodality ArSL dataset recorded using Kinect V2 at a rate of 30 frames per second. The dataset comprises 502 signs performed by three signers, and each sign is repeated 50 times by each signer; finally, we had 75,300 samples. Each sign was available in three modalities: RGB, depth, and skeleton joint points. We employed joint points to extract the key postures of the sign gestures and used the corresponding frames in the video stream as the input to the proposed models. The dataset comprised different types of sign gestures, which included digits and numbers (30 signs), letters (39 signs), and sign words (433 sign words). All of these signs are available in RGB video format. For alphabet letters and digits, the KArSL dataset contains more signs representing the combination between some letters or digits of ArSL, such as Alif letter with Hamza, 10, 100, and 200 signs. Figure 6 (a) shows samples from KArSL dataset.

We used two sets of the KArSL dataset: KArSL-190 and KArSL-502. KArSL-190 is the pilot version of the KArSL dataset, and it consists of 190 signs that comprise 30 digit signs, 39 letter signs, and 121 word signs. We used this set to evaluate the proposed techniques and compared our work with other studies that used this set. We also evaluated our approach on more signs using KArSL-502, which included all the signs (502 signs) of the KArSL dataset. The results reported for KArSL-502 can also be used to benchmark the KArSL dataset because it is the first study to use the whole KArSL dataset.

LSA64 is an Argentinian sign language dataset that contains 3200 videos of 64 signs performed by ten signers. Each sign is repeated five times by each signer. The dataset was collected using an RGB color camera. The signers who performed the dataset signs wore colored gloves to ease the detection and tracking of their hands. However, we used the signs without performing any segmentation. Figure 6 (b) shows samples from LSA64 dataset.

B. RESULTS AND DISCUSSION

Several experiments have been conducted with different configurations to evaluate the efficiency of the proposed sign language recognition systems. Experiments were conducted in two modes: signer dependent and signer independent. In the signer-dependent mode, we tested the model on samples of the signers who were involved in the training of the model. By contrast, in the signer-independent mode, we tested the system on the signs performed by the signers who were not present for the model training. For the signer-dependent mode, four sets of experiments were performed on the KArSL dataset—three sets corresponded to each of the three signers in the KArSL dataset, and one set contained the signs of all the signers. The signer-independent experiments were conducted using three sets corresponding to each signer tested for the dataset. For example, in the set used for Signer 01 in the signer-independent mode, two signers (Signer 02 and Signer 03) were used for training, and one signer (Signer 01) was used for testing.

In these experiments, we started by evaluating each component of the proposed system independently. We evaluated the DMN stream using different pre-trained networks on 18 key postures selected empirically. The CNN component of this network was fine-tuned using three pre-trained models for sign recognition, namely, VGG16, Xception, ResNet152 and MobileNet. The feature vectors resulting from these networks were fed into the stacked LSTM, as discussed in Section III-C1. Then, we evaluated the AMN stream using three configurations: forward (FWD-AMN), backward (BWD-AMN), and bidirectional (BWD-AMN). This stream accepts the AVM image as an input and employs a pre-trained MobileNet network for features extraction, as discussed in Section III-C2. Finally, we evaluated the SRN network that accepts the dynamic and accumulative motion

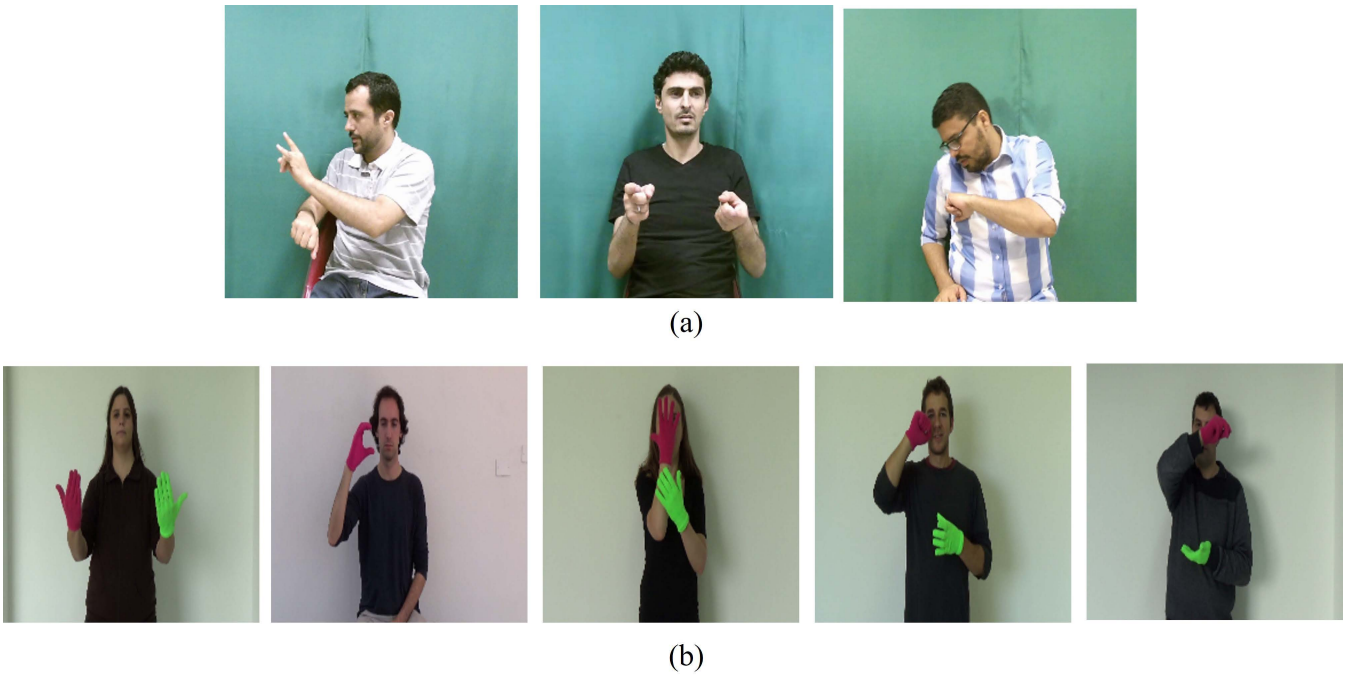

FIGURE 6. Samples from the used datasets: (a) KArSL and (b) LSA64.

features extracted from the DMN and AMN streams, respectively. All these experiments were performed using Tensorflow 2.5 on a workstation with Nvidia GeForce RTX 2080TI graphics processing unit (GPU) with 11 GB GPU memory and 64 GB RAM memory.

The DMN stream consists of stacked LSTM layers as discussed in Section III-C1. The LSTM component of this stream has been selected empirically as shown in Table 2. This table shows the recognition accuracies of the DMN stream with LSTM and GRU components using different pre-trained models on KArSL-502 dataset. Clearly, the DMN stream with LSTM and MobileNet model outperformed the other pre-trained models. Therefore, we conducted all other experiments in this work using the DMN stream with LSTM and MobileNet pre-trained model.

Table 3 shows the obtained results for the proposed models in the signer-dependent mode using KArSL-190 and KArSL-502 datasets. It is also noticeable that the AMN stream with all the fusion configurations outperformed the DMN stream. This can be attributed to the ability of AMN to capture the static sign gestures with minor differences encoded by the AVM technique, which is challenging for DMN. The highest accuracies of AMN fusions are obtained with bidirectional AMN (Bi-AMN) that considers fusion in both directions. We also showed the results of the SRN stream. The features extracted using DMN with MobileNet were fused with the features extracted using AMN to form the input for SRN. DMN with MobileNet was selected because it performed better than other pre-trained models. We evaluated this fusion with forward, backward, and bidirectional AMNs, which are shown in Table 2 as FWD-SRN, BWD-SRN, and Bi-SRN, respectively. Table 3 also shows that the obtained accuracies using the SRN network outperformed the DMN stream for the KArSL-190 and KArSL-502 datasets. By contrast, there was no noticeable improvement over the AMN stream except for Signer 01 of KArSL-502. However, the obtained results with SRN were high in the signer-dependent mode.

Although the results obtained for the proposed networks with the signer-dependent mode can be considered satisfactory, the more challenging type of sign language recognition is with the signer-independent mode. This type of recognition is related to the real-time systems that are tested on signers who are different from the signers involved in system training. To this end, we used two signers from the KArSL dataset for training and a third signer for testing. We followed the same experimental settings used for the signer-dependent experiments. Comparing Tables 3 and 4 shows that the signer-independent recognition was more challenging than the signer-dependent recognition. It is clear from Table 4 that the accuracies of all the configurations of the AMN stream on both datasets were significantly higher than the accuracies of the configurations of the DMN stream. The greatest improvement in all the AMN fusions is with the bidirectional AMN. Fusing this stream with DMN-MobileNet and feeding them into SRN helped to improve the results on KArSL-190 for all the signers. For KArSL-502, the fusion of DMN and AMN improved the accuracy of all the signers as compared with FWD-AMN and BWD-AMN. However,

TABLE 2. Recognition accuracies of the DMN stream on KArSL-502.

	Model	Signer-dependent				Signer-independent			
		Signer 1	Signer 2	Signer 3	All	Signer 1	Signer 2	Signer 3	Average
LSTM	DMN-VGG16	0.973	0.984	0.992	0.985	0.253	0.156	0.217	0.209
	DMN-Xception	0.976	0.991	0.99	0.986	0.198	0.139	0.141	0.159
	DMN-ResNet152	0.981	0.995	0.996	0.988	0.229	0.15	0.207	0.195
	DMN-MobileNet	**0.981**	**0.994**	**0.996**	**0.993**	**0.267**	**0.194**	**0.236**	**0.232**
GRU	DMN-VGG16	0.981	0.996	0.997	0.982	0.224	0.158	0.212	0.198
	DMN-Xception	0.982	0.997	0.999	0.991	0.239	0.148	0.154	0.180
	DMN-ResNet152	0.981	0.993	0.997	0.988	0.257	0.164	0.243	0.221
	DMN-MobileNet	0.985	0.992	0.998	0.992	0.228	0.151	0.216	0.180

TABLE 3. Signer-dependent recognition results on KArSL-190 and KArSL-502.

Model	KArSL-190				KArSL-502			
	Signer 01	Signer 02	Signer 03	All	Signer 01	Signer 02	Signer 03	All
DMN-MobileNet	0.967	0.993	0.997	0.981	0.981	0.994	0.996	0.993
FWD-AMN	0.974	0.995	0.999	0.991	0.993	0.998	0.999	0.995
BWD-AMN	0.978	0.997	0.998	0.988	0.990	0.997	0.999	0.992
Bi-AMN	**0.980**	**1.000**	**0.997**	**0.991**	0.991	**0.998**	**0.999**	**0.996**
FWD-SRN	0.979	0.990	0.993	0.984	0.991	0.989	0.988	0.990
BWD-SRN	0.974	0.988	0.985	0.985	**0.996**	0.987	0.996	0.980
Bi-SRN	0.971	0.992	0.993	0.990	0.991	0.990	0.996	0.988

TABLE 4. Signer-independent recognition rates.

Model	KArSL-190				KArSL-502			
	Signer 01	Signer 02	Signer 03	Average	Signer 01	Signer 02	Signer 03	Average
DMN-MobileNet	0.167	0.166	0.183	0.172	0.267	0.194	0.236	0.232
FWD-AMN	0.368	0.343	0.180	0.297	0.394	0.285	0.179	0.286
BWD-AMN	0.333	0.294	0.300	0.309	0.289	0.228	0.252	0.256
Bi-AMN	0.408	0.329	0.413	0.383	**0.390**	**0.295**	**0.343**	**0.343**
FWD-SRN	0.334	0.330	0.390	0.351	0.352	0.298	0.326	0.325
BWD-SRN	0.307	0.356	0.336	0.333	0.258	0.230	0.213	0.234
Bi-SRN	**0.363**	**0.423**	**0.419**	**0.402**	0.358	0.269	0.355	0.327

the accuracy of only Signer 03 improved with Bi-SRN as compared with Bi-AMN.

To evaluate the performance of the proposed networks on each sign category, we show in Table 5 the recognition accuracies of each stream separately on the three categories of KArSL signs (numbers, letters, and sign words) in the signer-independent mode. The accuracies shown in Table 5 are for the bidirectional AMN and DMN with the MobileNet pre-trained model because both models outperformed other settings. Table 5 also shows that for all the models, the signs of type *number* were the most challenging to recognize, followed by the *letters* signs. This can be attributed to the lack of motion in these signs. In addition, most of these signs are static and the differences between some of these static signs are marginal. For example, 'Dāt' and 'Sād' ArSL letters are almost similar and differ only on the position of the thumb finger. Additionally, certain number signs have only marginal variations, which cannot be captured easily with the recognition models. In contrast, the highest recognition rates were obtained with sign words that can be attributed to the variation between sign words and the use of motion with these signs. It is also noticeable in the confusion matrix that the AMN stream can recognize the static signs more efficiently than the DMN stream due to its ability in capturing the spatial features encoded by the AVM technique. The fusion of the DMN and AMN streams through the SRN stream improved the accuracies of all sign types for all signers except Signer 01 of KArSL-190. Furthermore, the SRN stream outperformed DMN with all sign types of KArSL-502.

To better investigate the misclassifications, we used a pie chart (Figure 7) of the misclassification signs of KArSL-502 for each network stream organized by the sign chapter (the KArSL dataset contains signs from 11 chapters of the ArSL dictionary). The signs involved in this analysis are those that could not be recognized by the network streams in the signer-independent mode for the three signers. As shown in the figure, most of the signs that could not be recognized by all the network streams belong to the characteristics chapter.

TABLE 5. Recognition accuracies of the proposed models per sign category in the signer-independent mode.

Signer	Signs	KArSL-190			KArSL-502		
		DMN	Bi-AMN	Bi-SRN	DMN	Bi-AMN	Bi-SRN
Signer 01	Numbers	0.062	0.383	0.174	0.118	0.351	0.225
	Letters	0.092	0.447	0.415	0.086	0.436	0.415
	Sign words	0.223	0.428	0.395	0.296	0.359	0.363
	Average	**0.126**	**0.420**	**0.328**	**0.167**	**0.382**	**0.334**
Signer 02	Numbers	0.033	0.143	0.297	0.104	0.196	0.166
	Letters	0.095	0.338	0.461	0.041	0.434	0.360
	Sign words	0.222	0.373	0.442	0.214	0.301	0.268
	Average	**0.117**	**0.285**	**0.400**	**0.119**	**0.310**	**0.265**
Signer 03	Numbers	0.024	0.202	0.409	0.046	0.146	0.163
	Letters	0.038	0.405	0.462	0.007	0.359	0.408
	Sign words	0.275	0.468	0.412	0.272	0.336	0.334
	Average	**0.112**	**0.358**	**0.428**	**0.108**	**0.281**	**0.302**

FIGURE 7. Misclassified signs by each network stream on KArSL-502 organized by the sign chapter.

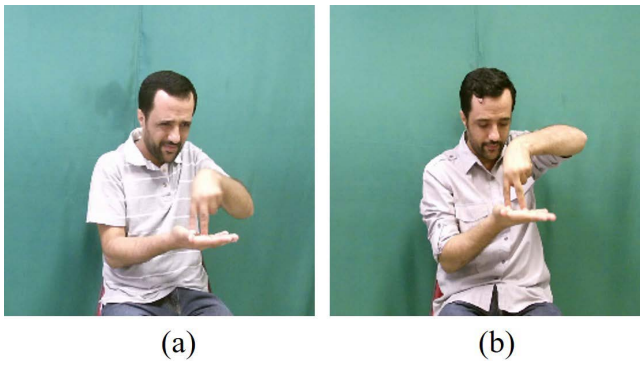

FIGURE 8. Two signs that share the same manual gestures but have different facial expressions: (a) afraid and (b) stand.

This chapter contains characteristic signs such as happy, poor, and selfish. Most of these signs have identical manual gestures and differ only in the facial expressions as shown in Figure 8. This figure shows how the ''afraid'' and ''stand'' signs of ArSL share the same gesture and motion but are accompanied by different facial expressions. Figure 7 also shows how the AMN stream could recognize almost all the static signs, unlike the DMN stream.

C. COMPARISON WITH OTHER WORKS

To evaluate the efficiency of the proposed approach, we compared the obtained results with the state-of-the-art techniques in the literature for the two datasets KArSL-190 and LSA64. KArSL-502 was published in 2021; therefore, no earlier work was available for comparison. Consequently, the reported results of this work could be used as a benchmark for the KArSL dataset. Sidig *et al.* [57] proposed four techniques for ArSL recognition. Three types of features were extracted from the skeleton's joint points provided by the Kinect sensor and fed into the HMM: (i) the joint points of the signers' hands, (ii) the hand shape represented using HOG, and (iii) a combination of joint points and the shapes of the signers' hands. Additionally, they formed a single image from all the frames of the signs and used a CNN model with VGG-19 for classification. Table 6 compares the results of these techniques with our results using KArSL-190. As shown in the table, the obtained results of the proposed AMN and SRN streams in the signer-dependent and signer-independent modes outperformed other techniques. In addition, the improvements in accuracy over the Sidig and Mahmoud [57] results with Bi-SRN were approximately 11% and 15% in the signer-dependent and signer-independent modes, respectively. These results confirm the efficiency of our proposed networks for sign recognition.

The LSA64 dataset, which is an Argentinian dataset consisting of 64 signs performed by ten signers, was also used to evaluate the generalization of our approach to other sign languages. We evaluated the proposed approach in the signer-dependent and signer-independent modes. For the signer-dependent mode, we split the data randomly into the train (80%) and test (20%) sets; we repeated each experiment

TABLE 6. Comparison with other works using KArSL-190.

Model	Signer-dependent	Signer-independent			
	Average	Signer 01	Signer 02	Signer 03	Average
Joint points + HMM [58]	0.843	0.160	0.080	0.117	0.119
HOG + HMM [58]	0.881	0.190	0.150	0.177	0.172
Joint points + HOG + HMM [58]	0.853	0.156	0.080	0.116	0.117
VGG-19 [58]	0.76	0.280	0.267	0.222	0.256
DMN-MobileNet	0.985	0.167	0.166	0.183	0.172
FWD-AMN	0.990	0.368	0.343	0.180	0.297
BWD-AMN	0.990	0.333	0.294	0.300	0.309
Bi-AMN	**0.992**	0.408	0.329	0.413	0.383
FWD-SRN	0.988	0.334	0.330	0.390	0.351
BWD-SRN	0.985	0.307	0.356	0.336	0.333
Bi-SRN	0.987	**0.363**	**0.423**	**0.419**	**0.406**

TABLE 7. Comparison with other works using ISA64 dataset (* evaluated on 46 sign gestures of ISA64).

Model	Signer-dependent	Signer-independent
3D CNN [61]	0.939	–
Skeletal features + LSTM [63]	0.981	–
Statistical features + Multiclassifiers [60]	0.974	0.917
CNN-LSTM [62]	0.952*	–
ConvNet [64]	0.978	–
CSD + SVM [65]	–	0.850
DMN-MobileNet	**0.991**	0.258
FWD-AMN	0.976	0.858
BWD-AMN	0.968	0.848
Bi-AMN	0.985	**0.918**
FWD-SRN	0.949	0.818
BWD-SRN	0.964	0.784
Bi-SRN	0.975	0.885

TABLE 8. Signer-independent recognition accuracies of the bidirectional accumulative motion network on the LSA64 dataset.

Signer	01	02	03	04	05	06	07	08	09	10	Average
Accuracy	0.947	0.906	0.834	0.966	0.919	0.931	0.938	0.897	0.934	0.913	**0.918**

five times. For the signer-independent mode, nine signers were used for model training, and the 10th signer was used as an unseen signer for testing. We compared the results of our approach with the results obtained by Ronchetti *et al.* [59], Neto *et al.* [60], Masood *et al.* [61], Konstantinidis *et al.* [62], and Imran *et al.* [63]. Ronchetti *et al.* [59] proposed a probabilistic model that combines the outputs of three classifiers trained on a set of statistical features. Neto *et al.* [60] proposed a 3D CNN architecture for sign recognition. Konstantinidis *et al.* [62] proposed an LSTM model to classify the signs based on the hand and body skeletal features. Rodriguez *et al.* [64] used cumulative shape difference (CSD) with SVM for sign-independent recognition. Masood *et al.* [61] applied a CNN-LSTM model for sign video classification wherein the CNN model was trained on a pre-trained Inception model. This approach was evaluated on 46 gestures of the LSA64 dataset. Imran *et al.* [63] proposed three motion templates to encode the hand movements of the sign gestures. These representations were fed into the pre-trained CNN for gestures learning and classification. The comparative results are presented in Table 7. Clearly, our approach outperformed other approaches in the signer-dependent and signer-independent experiments. The highest accuracy in the signer-independent mode was obtained using Bi-AMN. In this experiment, the lowest accuracies were obtained with Signer 02, Signer 03, and Signer 08 (see Table 8). These signers were nonexpert signers, and they introduced certain movements that were not part of the sign language, such as head motions and returning hands to their resting positions before signing. These observations align with the challenges reported for the LSA64 dataset in [64].

V. CONCLUSION AND FUTURE WORK

In the last decade, sign language recognition has gained popularity and attracted the interest of researchers worldwide. Several approaches that differ in the sign's acquisition method, recognition technique, target language, and number of recognized signs have been proposed for isolated sign

language recognition. In this research, three deep learning models (namely, DMN, AMN, and SRN) have been proposed for sign language recognition. The DMN stream learns the spatiotemporal information of the sign's key postures. In this research, we propose a technique to extract key postures for handling the variations between the sign's samples. This technique uses the dominant postures that represent the key motion changes of the sign. We also proposed the AVM approach to encode the sign motion into a single image. This image was used as the input to the second proposed network, namely, AMN. The third proposed network was SRN, which fused the features extracted from the DMN and AMN streams and used them as the input. These networks were evaluated on two datasets, and the obtained results proved that the AMN is efficient for sign language recognition compared with other streams and it outperformed the state-of-the-art techniques.

Signer-independent recognition is more challenging than signer-dependent, and the number of signers used for model training plays a vital role in the model's accuracy. Models trained on a large number of signers are expected to have higher signer-independent accuracy compared with models trained on a small number of signers. This can be noticed in our results when we used the KArSL dataset with 3 signers and the ISA64 dataset with 10 signers. This has also been noticed in the literature where models trained on a large number of signers reported high signer-independent accuracy [34], [65], [66], [67], whereas models trained on a small number of signers usually reported lower accuracies [13], [43], [68].

As a future work, other models can be used for sign language recognition, such as attention mechanism and Transformers. In addition, we will use other modalities for sign language recognition.

REFERENCES

[1] (Apr. 30, 2022). *Hearing Loss Statistics*. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss

[2] H. Luqman and S. A. Mahmoud, "Automatic translation of Arabic text-to-Arabic sign language," *Universal Access Inf. Soc.*, vol. 18, no. 4, pp. 939–951, 2018.

[3] A. A. I. Sidig, H. Luqman, and S. A. Mahmoud, "Transform-based Arabic sign language recognition," *Proc. Comput. Sci.*, vol. 117, pp. 2–9, Nov. 2017.

[4] A. A. I. Sidig, H. Luqman, and S. A. Mahmoud, "Arabic sign language recognition using optical flow-based features and HMM," in *Proc. Int. Conf. Reliable Inf. Commun. Technol.*, 2017, pp. 297–305.

[5] *LAS: Second Part of the Unified Arabic Sign Dictionary*, League Arab States Arab League Educ., Cultural Sci. Org., Tunisia, 2006.

[6] *LAS: First Part of the Unified Arabic Sign Dictionary*, League Arab States & Arab League Educ., Cultural Sci. Org., Tunisia, 2000.

[7] G. M. B. Makhashen, H. A. Luqman, and E.-S.-M. El-Alfy, "Using Gabor filter bank with downsampling and SVM for visual sign language alphabet recognition," in *Proc. 2nd Smart Cities Symp. (SCS)*, 2019, pp. 1–6.

[8] A. V. Nair and V. Bindu, "A review on Indian sign language recognition," *Int. J. Comput. Appl.*, vol. 73, no. 22, pp. 33–38, 2013.

[9] 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.

[10] N. E. Goldstein and R. S. Feldman, "Knowledge of American sign language and the ability of hearing individuals to decode facial expressions of emotion," *J. Nonverbal Behav.*, vol. 20, no. 2, pp. 111–122, Jun. 1996.

[11] U. Von Agris, M. Knorr, and K.-F. Kraiss, "The significance of facial features for automatic sign language recognition," in *Proc. 8th IEEE Int. Conf. Autom. Face Gesture Recognit.*, Sep. 2008, pp. 1–6.

[12] P. K. Pisharady and M. Saerbeck, "Recent methods and databases in vision-based hand gesture recognition: A review," *Comput. Vis. Image Understand.*, vol. 141, pp. 152–165, Dec. 2015.

[13] H. Luqman and E.-S.-M. El-Alfy, "Towards hybrid multimodal manual and non-manual Arabic sign language recognition: MArSL database and pilot study," *Electronics*, vol. 10, no. 14, p. 1739, Jul. 2021.

[14] M. A. Ahmed, B. B. Zaidan, A. A. Zaidan, M. M. Salih, and M. M. B. Lakulu, "A review on systems-based sensory gloves for sign language recognition state of the art between 2007 and 2017," *Sensors*, vol. 18, no. 7, p. 2208, 2018.

[15] Q. Zhang, D. Wang, R. Zhao, and Y. Yu, "MyoSign: Enabling end-to-end sign language recognition with wearables," in *Proc. 24th Int. Conf. Intell. User Interfaces*, Mar. 2019, pp. 650–660.

[16] R.-H. Liang and M. Ouhyoung, "A real-time continuous gesture recognition system for sign language," in *Proc. 3rd IEEE Int. Conf. Autom. Face Gesture Recognit.*, 1998, pp. 558–567.

[17] G. Fang, W. Gao, X. Chen, C. Wang, and J. Ma, "Signer-independent continuous sign language recognition based on SRN/HMM," in *Proc. Int. Gesture Workshop*, 2001, pp. 76–85.

[18] C. Wang, W. Gao, and Z. Xuan, "A real-time large vocabulary continuous recognition system for Chinese sign language," in *Proc. Pacific-Rim Conf. Multimedia*. Berlin, Germany: Springer, 2001, pp. 150–157.

[19] W. Gao, G. Fang, D. Zhao, and Y. Chen, "Transition movement models for large vocabulary continuous sign language recognition," in *Proc. 6th IEEE Int. Conf. Autom. Face Gesture Recognit.*, 2004, pp. 553–558.

[20] G. Yao, H. Yao, X. Liu, and F. Jiang, "Real time large vocabulary continuous sign language recognition based on OP/Viterbi algorithm," in *Proc. 18th Int. Conf. Pattern Recognit. (ICPR)*, 2006, pp. 312–315.

[21] W. W. Kong and S. Ranganath, "Towards subject independent continuous sign language recognition: A segment and merge approach," *Pattern Recognit.*, vol. 47, no. 3, pp. 1294–1308, 2014.

[22] T. Ritchings, A. Khadragi, and M. Saeb, "An intelligent computer-based system for sign language tutoring," *Assistive Technol.*, vol. 24, no. 4, pp. 299–308, 2012.

[23] M. Mohandes and M. Deriche, "Arabic sign language recognition by decisions fusion using Dempster-Shafer theory of evidence," in *Proc. Comput., Commun. IT Appl. Conf. (ComComAp)*, Apr. 2013, pp. 90–94.

[24] 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.

[25] W. Nai, Y. Liu, D. Rempel, and Y. Wang, "Fast hand posture classification using depth features extracted from random line segments," *Pattern Recognit.*, vol. 65, pp. 1–10, May 2017.

[26] S. G. M. Almeida, F. G. Guimarães, and J. A. Ramírez, "Feature extraction in Brazilian sign language recognition based on phonological structure and using RGB-D sensors," *Exp. Syst. Appl.*, vol. 41, no. 16, pp. 7259–7271, 2014.

[27] G. Joshi, S. Singh, and R. Vig, "Taguchi-TOPSIS based HOG parameter selection for complex background sign language recognition," *J. Vis. Commun. Image Represent.*, vol. 71, Aug. 2020, Art. no. 102834.

[28] M. M. Zaki and S. I. Shaheen, "Sign language recognition using a combination of new vision based features," *Pattern Recognit. Lett.*, vol. 32, no. 4, pp. 572–577, 2011.

[29] T.-Y. Pan, L.-Y. Lo, C.-W. Yeh, J.-W. Li, H.-T. Liu, and M.-C. Hu, "Real-time sign language recognition in complex background scene based on a hierarchical clustering classification method," in *Proc. IEEE 2nd Int. Conf. Multimedia Big Data (BigMM)*, Apr. 2016, pp. 64–67.

[30] H. B. D. Nguyen and H. N. Do, "Deep learning for American sign language fingerspelling recognition system," in *Proc. 26th Int. Conf. Telecommun. (ICT)*, Apr. 2019, pp. 314–318.

[31] S. Celebi, A. S. Aydin, T. T. Temiz, and T. Arici, "Gesture recognition using skeleton data with weighted dynamic time warping," in *Proc. VISAPP*, 2013, pp. 620–625.

[32] G. Plouffe and A.-M. Cretu, "Static and dynamic hand gesture recognition in depth data using dynamic time warping," *IEEE Trans. Instrum. Meas.*, vol. 65, no. 2, pp. 305–316, Feb. 2016.

[33] G. A. Rao and P. V. V. Kishore, "Selfie video based continuous Indian sign language recognition system," *Ain Shams Eng. J.*, vol. 9, no. 4, pp. 1929–1939, Dec. 2018.

[34] K. Pattanaworapan, K. Chamnongthai, and J.-M. Guo, "Signer-independence finger alphabet recognition using discrete wavelet transform and area level run lengths," *J. Vis. Commun. Image Represent.*, vol. 38, pp. 658–677, Jul. 2016.
[35] A. Thalange and S. K. Dixit, "COHST and wavelet features based static ASL numbers recognition," *Proc. Comput. Sci.*, vol. 92, pp. 455–460, Jan. 2016.
[36] M. A. Aowal, A. S. Zaman, S. M. M. Rahman, and D. Hatzinakos, "Static hand gesture recognition using discriminative 2D Zernike moments," in *Proc. TENCON IEEE Region Conf.*, Oct. 2014, pp. 1–5.
[37] R. Sabhara, "Comparative study of Hu moments and Zernike moments in object recognition," *Smart Comput. Rev.*, vol. 3, no. 3, pp. 166–173, Jun. 2013.
[38] K. Otiniano-Rodríguez, G. Cámara-Chávez, and D. Menotti, "Hu and Zernike moments for sign language recognition," in *Proc. Int. Conf. Image Process., Comput. Vis., Pattern Recognit.*, 2012, pp. 1–5.
[39] X. Jiang and Y.-D. Zhang, "Chinese sign language fingerspelling via six-layer convolutional neural network with leaky rectified linear units for therapy and rehabilitation," *J. Med. Imag. Health Informat.*, vol. 9, no. 9, pp. 2031–2090, Dec. 2019.
[40] A. A. Barbhuiya, R. K. Karsh, and R. Jain, "CNN based feature extraction and classification for sign language," *Multimedia Tools Appl.*, vol. 80, no. 2, pp. 3051–3069, 2021.
[41] E.-S.-M. El-Alfy and H. Luqman, "A comprehensive survey and taxonomy of sign language research," *Eng. Appl. Artif. Intell.*, vol. 114, Sep. 2022, Art. no. 105198.
[42] T. Liu, W. Zhou, and H. Li, "Sign language recognition with long short-term memory," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2016, pp. 2871–2875.
[43] A. A. I. Sidig and S. A. Mahmoud, "Trajectory based Arabic sign language recognition," *Int. J. Adv. Comput. Sci. Appl.*, vol. 9, no. 4, pp. 283–291, 2018.
[44] J. Huang, W. Zhou, H. Li, and W. Li, "Sign language recognition using 3D convolutional neural networks," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2015, pp. 1–6.
[45] S.-Z. Li, B. Yu, W. Wu, S.-Z. Su, and R.-R. Ji, "Feature learning based on SAE–PCA network for human gesture recognition in RGBD images," *Neurocomputing*, vol. 151, pp. 565–573, Mar. 2015.
[46] A. Sabyrov, M. Mukushev, and V. Kimmelman, "Towards real-time sign language interpreting robot: Evaluation of non-manual components on recognition accuracy," in *Proc. CVPR Workshops*, 2019, pp. 1–8.
[47] P. Kumar, P. P. Roy, and D. P. Dogra, "Independent Bayesian classifier combination based sign language recognition using facial expression," *Inf. Sci.*, vol. 428, pp. 30–48, Feb. 2018.
[48] Q. Wang, G. Kurillo, F. Ofli, and R. Bajcsy, "Evaluation of pose tracking accuracy in the first and second generations of Microsoft Kinect," in *Proc. Int. Conf. Healthcare Informat.*, Oct. 2015, pp. 380–389.
[49] T. Shanableh, K. Assaleh, and M. Al-Rousan, "Spatio-temporal feature-extraction techniques for isolated gesture recognition in Arabic sign language," *IEEE Trans. Syst. Man, Cybern. B, Cybern.*, vol. 37, no. 3, pp. 641–650, Jun. 2007.
[50] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.
[51] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1251–1258.
[52] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 630–645.
[53] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*.
[54] Y. Chen, L. Zhu, P. Ghamisi, X. Jia, G. Li, and L. Tang, "Hyperspectral images classification with Gabor filtering and convolutional neural network," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 12, pp. 2355–2359, Dec. 2017.
[55] S. Santurkar, D. Tsipras, A. Ilyas, and A. Madry, "How does batch normalization help optimization?" in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 2483–2493.
[56] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," 2015, *arXiv:1502.03167*.
[57] A. A. I. Sidig, H. Luqman, S. Mahmoud, and M. Mohandes, "KArSL: Arabic sign language database," *ACM Trans. Asian Low-Resource Lang. Inf. Process.*, vol. 20, no. 1, pp. 1–19, Jan. 2021.
[58] F. Ronchetti, F. Quiroga, C. Estrebou, L. Lanzarini, and A. Rosete, "LSA64: A dataset of Argentinian sign language," in *Proc. 22nd Congreso Argentino de Ciencias de la Computación (CACIC)*, 2016, pp. 794–803.
[59] F. Ronchetti, F. Quiroga, C. Estrebou, L. Lanzarini, and A. Rosete, "Sign languague recognition without frame-sequencing constraints: A proof of concept on the Argentinian sign language," in *Advances in Artificial Intelligence (IBERAMIA)* (Lecture Notes in Computer Science), M. M. Y. Gómez, H. J. Escalante, A. Segura, and J. d. D. Murillo, Eds. Cham, Switzerland: Springer, 2016, pp. 338–349.
[60] G. M. R. Neto, G. B. Junior, J. D. S. de Almeida, and A. C. de Paiva, "Sign language recognition based on 3D convolutional neural networks," in *Image Analysis and Recognition*. Cham, Switzerland: Springer, 2018, pp. 399–407.
[61] S. Masood, A. Srivastava, H. C. Thuwal, and M. Ahmad, "Real-time sign language gesture (word) recognition from video sequences using CNN and RNN," in *Intelligent Engineering Informatics*. Singapore: Springer, 2018, pp. 623–632.
[62] D. Konstantinidis, K. Dimitropoulos, and P. Daras, "Sign language recognition based on hand and body skeletal data," in *Proc. Conf., True Vis. Capture, Transmiss. Display 3D Video (DTV-CON)*, Jun. 2018, pp. 1–4.
[63] J. Imran and B. Raman, "Deep motion templates and extreme learning machine for sign language recognition," *Vis. Comput.*, vol. 36, no. 6, pp. 1233–1246, Jun. 2020.
[64] J. Rodríguez and F. Martínez, "Towards on-line sign language recognition using cumulative sd-vlad descriptors," in *Proc. Colombian Conf. Comput.*, 2018, pp. 371–385.
[65] S. Ravi, M. Suman, P. V. V. Kishore, K. Kumar, and A. Kumar, "Multi modal spatio temporal co-trained CNNs with single modal testing on RGB–D based sign language gesture recognition," *J. Comput. Lang.*, vol. 52, pp. 88–102, Jun. 2019.
[66] M. Al-Rousan, K. Assaleh, and A. Tala'a, "Video-based signer-independent Arabic sign language recognition using hidden Markov models," *Appl. Soft Comput.*, vol. 9, no. 3, pp. 990–999, 2009.
[67] S. G. Azar and H. Seyedarabi, "Trajectory-based recognition of dynamic persian sign language using hidden Markov model," *Comput. Speech Lang.*, vol. 61, May 2020, Art. no. 101053.
[68] V. Ranga, N. Yadav, and P. Garg, "American sign language fingerspelling using hybrid discrete wavelet transform-Gabor filter and convolutional neural network," *J. Eng. Sci. Technol.*, vol. 13, no. 9, pp. 2655–2669, 2018.

HAMZAH LUQMAN received the bachelor's degree in computer science from Ahgaff University, Yemen, in 2006, and the master's degree in computer science and the Ph.D. degree in computer science and engineering from the King Fahd University of Petroleum & Minerals (KFUPM), Saudi Arabia, in 2013 and 2018, respectively. He is currently an Assistant Professor with the Department of Information and Computer Science, KFUPM. He has several publications in sign language recognition and translation published in reputed journals and conferences. His research interests include machine learning and computer vision techniques for visual recognition tasks, such as gesture recognition, medical imaging, and Arabic text recognition.