An LSTM-based System for Dynamic Arabic Sign Language Recognition*

Slimane Oulad-Naoui[1], Habiba Ben-Abderrahmane[2,*], Assia Chagha[1], and Abderrahmane Cherif[1]

[1] Lab. des Mathématiques et Sciences Appliquées, University of Ghardaia, Algeria
{ouladnaoui,chagha.assia,cherif.abderrahmane.cs}@univ-ghardaia.dz
[2] Laboratoire d'Informatique et des Mathematiques, University of Laghouat, Algeria
* Corresponding author: Habiba Ben-Abderrahmane
habiba.benabderrahmane@lagh-univ.dz

Abstract. Recognizing sign language is a vital task that assists in demystifying communication with deaf-mute persons. Many previous works tackle this problem by considering a static point of view, such as isolated single alphabet symbols or digit detection. This paper introduces an Arabic Sign Language (ArSL) recognition system using a deep learning technique. Our work raises the recognition level in two aspects: first, it deals with the dynamic nature of the problem and hence admits input from videos; and second, it experiments with a recent Arabic sign language dataset. Since only particular parts of the input convey the desired message, the proposed model pays attention to only the main regions in the input video and thus relies mainly on the use of keypoints of zones of interest tracked from video frame sequences. Regarding the sequence nature of the input data, the extracted keypoints are fed to an LSTM-based architecture specifically tailored to discover sentences from the input. Experiments on the ArabSign dataset reveal that our model succeeds in reproducing sentences existing in a sign video with an accuracy rate that surpasses 88.5%. The obtained results validate undeniably the effectiveness of the proposed model in recognizing a multitude of ArSL gestures.

Keywords: Arabic Sign Language Recognition, LSTM, Video Processing, Keypoint Tracking

1 Introduction

The World Federation of the Deaf records approximately 70 million deaf and more than 400 million individuals with hearing disability and the prevalence of over 300 sign languages [1]. Meanwhile, existing communication technologies predominantly cater to spoken or written language and often neglect the deaf

* This work is supported by the Directorate-General for Scientific Research and Technological Development (DGRSDT) - Algeria, and performed under the PRFU Project: C00L07UN470120230001.

C. A. Kerrache et al. (eds.), *Proceedings of the International Conference on Emerging Intelligent Systems for Sustainable Development (ICEIS 2024)*, Advances in Intelligent Systems Research 184,
https://doi.org/10.2991/978-94-6463-496-9_24

and dumb community, which still encounters communication difficulties and usability challenges. The integration of sign language (SL) into communication systems enriches daily interactions among deaf, speech-impaired, and hearing communities. Recognizing sign language as a structured form of manual and visual expression emphasizes the need for automated sign language recognition (SLR) systems.

Sign languages entail multiple both intrinsic and extrinsic traits that make their recognition an inextricable problem. Despite strides in improving SLR systems with, to some extent, satisfactory achievements and results, the field still confronts several major challenges.

Foremost among these is the lack of universal SL. Many SL are used all over the world. Even in the same region, the deaf community uses disparate sign languages. The absence of standards complicates the attempts to build efficient SLR systems. Moreover, certain signs particularly those that associate intricate nuances of hand movements, facial expressions, and body postures are extremely hard to catch by an automatic process. Likewise, the presence of noise due to environmental factors, sub-optimal recording conditions, and variations in signing gestures even renders the detection of meaningful signs a more difficult task. Additionally, the dynamic nature of SL including spatiotemporal dependencies adds more complexity to the recognition process. Besides the recognition central task, the substantial data required to train recent SLR models poses an additional hurdle. Effectively addressing all these nuanced aspects requires sophisticated modeling techniques.

Regarding the input data, work on SLR systems may be either vision-based or sensor-based. Whereas, when addressing the task point of view, we can distinguish three major classes of work: fingerspelling recognition, isolated symbol recognition, and continuous SLR. Primary works in SLR typically seek fingerspelling [5] or isolated single-symbol recognition [3,8,9], thereby ignoring the temporal interrelationship hidden within the sign gesture.

Recently, with the success gathered by AI-powered systems, numerous works envisage powerful SLR systems using machine learning (ML) [2] or deep learning (DL) techniques [23]. The early intelligent SLR systems rely on handcraft models. They use standard image engineering tasks: acquisition, preprocessing, feature extraction, and lastly classification. Although the obtained result was acceptable, dependency on manual processes makes these solutions more difficult, resource-intensive, and time-consuming, which limits their efficacy.

In this work, we develop a deep learning-based SLR system devoted to Arabic signs. The system goal is to extract, end-to-end, the phrase embedded by an Arabic signer from video data where a sequence of signs represents a sentence.

The rest of this paper is organized as follows: We recall the elementary points on SLR systems in Section 2. Section 3 reviews the main techniques used in SLR. The proposed model is discussed in Section 4. The detail of the experimentation is illustrated with the obtained result in Section 5. We conclude in Section 6 and outline possible further extensions to the present work.

2 Background

A SL is a set of signs formed using different finger and hand movements, facial expressions, and body postures. A SLR system aims to identify and translate SL gestures into speech or text. It facilitates communication for individuals with hearing/speech impairment.

SLR systems follow a typical knowledge discovery pipeline [2]. They begin by collecting and preprocessing data. Key features are then extracted to help in the ultimate goal which is a recognition task by classifying the output obtained from a well-selected learning model.

Inputs to SLR systems may vary depending on the data acquisition technique and the chosen architecture and recognition approach. They exhibit different modalities when representing SL gestures. Some popular inputs include strain, tactile or EMG sensors, skeletal data, for sensor-based SLR; and images or videos in the case of vision-based SLR.

After the acquisition phase, the input data are prepared for further analysis. Frequent preprocessing tasks include resizing and cropping, normalization, noise reduction, data augmentation, and gesture segmentation in the case of continuous input. When the data is well-prepared a model is applied. Modeling involves designing hand-crafted or ML/DL models to translate input data into meaningful SL gestures or sequences. The modeling process contains a well-chosen model depending on the nature of the input, the complexity of the recognition task, and the available computational resources.

Once the model is ready, it is exercised in train-validate/test cycles. Finally, the system is leveraged when its evaluation is satisfactory. This evaluation assesses the performance of the constructed system in accurately recognizing SL gestures. Well-known evaluation metrics such as precision, recall, F1-score, and Word Error Rate(WER) are used. The last one computes how far is the recognized sequence from the ground truth sequence in terms of the needed edit operations.

In computer vision (CV), keypoints are essential locations within an image. They are crucial for various CV tasks such as object detection, motion tracking, etc. Given that SLR is mainly about special zone movement identification, keypoint detection constitutes a significant preprocessing task, particularly in vision-based SLR. MediaPipe [18] is an open-source multiplatform framework developed by Google for ML applications specifically for audio/visual and sensor data manipulation. It provides keypoints detection module that can be easily integrated with AI applications. We use this framework in the preprocessing stage of the present work.

Recurrent Neural Network (RNN) are a class of neural network specialized for processing sequence data such as those found in NLP applications, speech recognition, etc. Long Short Term Memory (LSTM) [15] is a type of RNN introduced to solve the vanishing gradients problem where the state of a cell is split into two parts: the cell state and the output state. LSTM cells provide three gates that control the flow of information. In an LSTM cell, the forget gate is used to erase information, the input gate is responsible for the information to update, and an output gate that selects the state information to output.

3 Related Work

This section explores diverse gesture recognition techniques within the context of Arabic SL. We investigate traditional approaches to ArSLR, passing by ML-based methods, and ultimately delving into DL-based techniques.

3.1 Traditional Methods

Authors of [21] provide a comprehensive image-based system to identify ArSL. The approach includes four steps: first, it implements the Gaussian skin color model to detect the dynamic face; second, hand movements, related to the face center, are tracked using the growth of the area for analysis; then, from the discovered hand areas, they select the main features that represent a specific ArSL gesture; Finally, Hidden Markov Model (HMM) are used for recognition. The evaluation of the system on a set of 300 marks achieved a great accuracy close to 93%. This work emphasizes the effectiveness of integrating face detection, tracking hands, extracting features, and the suitability of HMM in ArSL recognition.

In 2009, [4] introduced one of the earlier vision-based Arabic isolated sign recognition systems. The authors collect their own dataset using a single video camera and use again HMM as a recognizer technique. The system demonstrated notable word recognition rates of 98.13%, 96.74%, and 93.8% on offline training data, offline test data and online test data respectively. In location-independent scenarios, it achieved a word recognition rates of 94.2% and 90.6% in offline and online modes, respectively. This research developed an ArSL recognition system, regardless of individual signs. However, there is still a major challenge in recognizing uninterrupted sentence sequences.

In [12] the authors conducted a detailed comparison between two continuous recognition techniques for ArSLR. The study examined a modified kNN approach suitable for sequential data and HMM implemented with two different tool sets. Results indicated that the HMMs solution outperformed the modified kNN approach in terms of computational time required for classification. This thorough investigation not only advances understanding in the field of ArSL recognition but also offers valuable datasets and insights for researchers developing SLR systems.

3.2 Machine Learning-based Methods

A combination of subtractive clustering with least-squares estimator and a set of networks appeared in [3]. Each network is responsible for recognizing one ArSL symbol from input images. The whole system constitutes a fuzzy inference system that reaches 93.55% of accuracy in identifying 30 Arabic manual alphabets.

[14] introduced an ArSLR system utilizing the Leap and Latte Panda motion controllers, integrating kNN and SVM algorithms. Ada-Boosting is applied to enhance the accuracy, and Dynamic Time Wrapping (DTW) is evaluated. With 30 hand gestures examined, including single and dual-handed gestures, DTW achieved 88% of accuracy for single-handed and 86% for dual-handed gestures.

After AdaBoos, recognition rates reach 92.3% and 93% for single and dual-handed gestures respectively.

[10] proposed a visual ArSLR system that uses c-mean fuzzy clustering and neutrosophic for segmenting 28 symbols of ArSL. The input RGB-image was first resized, gray-scaled, and prepared via the Gaussian filter as a way to eliminate noise. It was then converted to the neutrosophic format where its characteristics are extracted. A rule based inference is done in the final stage. The system classification accuracy attained 91%.

In 2018, [7] addressed the necessity for automatic recognition of ArSL alphabets through an image-based approach. The study investigates various visual descriptors to develop an accurate recognizer for ArSL alphabet. These descriptors are then fed into a single-versus-all support SVM. Analysis revealed that the Histogram of Oriented Gradients descriptor outperforms other studied descriptors. The model achieved a recognition rate of 63.5% for Arabic alphabet letter gestures.

3.3 Deep Learning-based Methods

The limitations of ML-based methods, such as hand-crafted feature extraction and unsequenced data processing, insist on making use of another approach that handles these issues. Numerous works in ArSL recognition recourse to DL techniques [23] for more effective results.

One of the early works that use both feed-forward (FF) and RNN for hand gesture recognition appeared in [20]. The authors used a colored digital camera and colored gloves to produce images each one of them was resized to 256×256. Next, a vector of thirty features is derived from each image using fingertips and the relative positions and orientations. The FFNN used is composed of 3 layers whereas in the RNN both hidden and output layers are recurrent; a recurrent link exists also between the output and the input layer. The later architecture is used in sign identification from videos. The obtained result was 79% and 95% for the FFNN and RNN respectively.

[11] emplyed a 3DCNN architecture. The system successfully recognizes 25 gestures from the ArSL dictionary with an accuracy of 98% for observed data and an average accuracy of 85% for new data, the approach demonstrates robust performance in gesture recognition. In 2019, [13] introduced a novel system based on CNN utilizing a real dataset for ArSLR. The study highlights the superior performance of the CNN-based method compared to traditional approaches relying on KNN and SVM. The validation results emphasize the potential of the proposed system to enhance inclusivity and accessibility for individuals within the deaf community, however, the system does not deal with video inputs. The authors of [16] introduced a vision-based system utilizing CNNs for Arabic character recognition based on hand signals. The system achieved a notable accuracy of 90% in recognizing handwritten Arabic letters, highlighting its reliability. However, and despite the quality of the results, it is noted that the dataset size was small.

In [6], the authors proposed an ArSLR framework based on the DeepLabv3+ semantic segmentation model. The later architecture was used to extract hand regions from video frames and Convolutional Self-Organizing Map for hand-shape features. Sequence recognition was performed using a bidirectional LSTM. The work achieved an average accuracy of 89.5% on an ArSL dataset composed of 23 isolated Arabic word signs.

To detect isolated ArSL characters from voluntary videos recorded using smartphones, The work of [8] employs two separate CNNs that it combines and pass to a BiLSTM. The system reaches 92% of accuracy.

[5] use MobilNet architecture which is a lightweight and special kind of CNN tailored to mobile or embedded devices. The input data are RGB images which are grayscaled and resized to 64×64 corresponding to 32 ArSL characters. The experiment conducted on a simple CPU PC with 8G of RAM reached 94.46% of recognition rate.

An ArSLR system based on RGB videos is presented in [22]. It uses two datasets: a raw dataset and a segmented dataset focusing on the face and hand region. The study introduced a novel multilayer-based operational perceptron called "SelfMLP" integrated into CNN-LSTM-SelfMLP models for recognition. Six models were developed using CNN backbones MobileNetV2 and ResNet18, alongside three SelfMLPs, with performance comparison. The best precision of 87.69% was achieved by MobileNetV2-LSTM-SelfMLP on the segmented dataset, showing superiority in all metrics.

[17] introduces a simple yet effective SL translation model from videos. The approach includes a stage of 55 keypoint extraction and a normalization followed by a stochastic augmentation to select the appropriate frames to be considered. The last phase is a GRU model with an encoder/decoder and an attention mechanism to enhance the translation. Experiments conducted on both high and low-resolution datasets demonstrate the model's effectiveness.

4 Proposed Model

Our initial approach to ArSL system design involves a model based on 3DCNN, aiming to capture both the spatial and temporal features of the input. Unfortunately, we encounter during the experiment substantial resource constraints and heavy training time due to the large size of the dataset and the limited-resource environment. To address this concern, we manage to reduce both the memory usage and the training time. Based on the observation that only specific parts of the signer body are crucial and contribute really to the sign gestures, this leads us to consider keypoint tracking. The goal is to optimize the model's complexity while retaining the essential information.

The crux of our preprocessing phase lies in the extraction and preservation of these keypoints. Instead of retaining entire video frames, we store only the extracted keypoints in a compressed .npy format. Using the MediaPipe tool, we retain 10×66 numpy vectors corresponding to ten frames per video each of which records 33 landmarks with the common x and y coordinates.

The overall pipeline of the constructed system is shown in Figure 1.

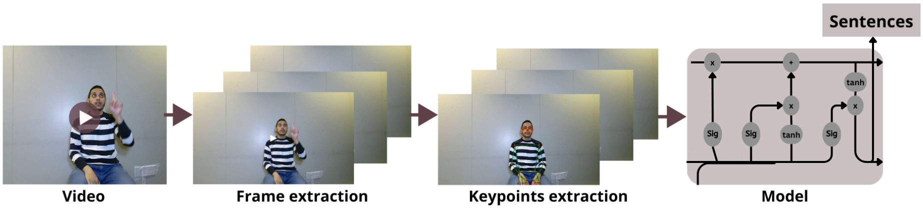

Fig. 1. Pipeline of the proposed system

It includes three main stages: preprocessing, learning, and classification. The mission of the first stage is to convert the input video into a sequence of ten frames. Next, frame keypoints are extracted and then fed to a recurrent neural network of LSTM type tailored to learn the temporal dynamic of the dataset as depicted by Figure 2. We stack two LSTM layers comprising 512 neurons each.

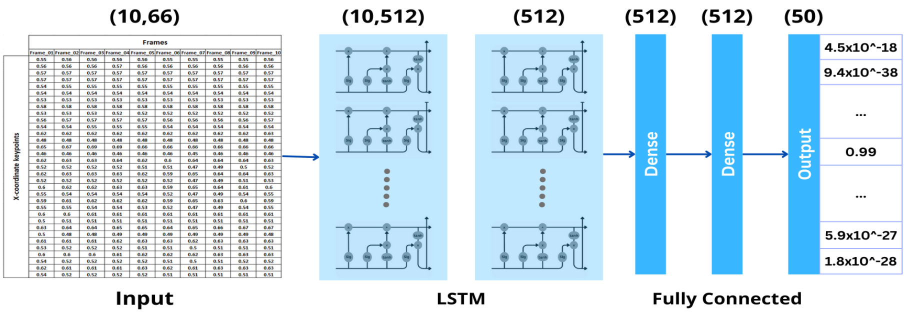

Fig. 2. Model architecture

This component learns intricate temporal patterns and dependencies within the data. The output layer of our model comprises 50 neurons reflecting the number of classes (sentences) of the target space. Preceded by a flatten and two dense layers, the outcome layer yields predictions that align with the desired output format. The model totalizes 3.8 million parameters. We manually adjust our hyperparameters and retain 10 frames per input video and 512 nodes for the two LSTM layers as a good tradeoff between model accuracy and its computation time.

5 Experiment

We implement our model with the Keras API with TensorFlow backend. Tests are performed on an Intel Xeon Silver 4112 processor running Ubuntu 22.04 LTS, equipped with 64GB of RAM, extended by an NVIDIA RTX 4060 GPU

Table 1. The dataset characteristics

Total Duration	10 hours 13 min.
Signer Age	21-30
Total Size	18GB
# of Samples	9,335
# Sentences	50
Average sentence length	3.1
Total # of frames	200,000
Frames per sentence	130.3

with 8GB of RAM. The recent ArabSign dataset [19] is used in the experiment. Table 1 lists the main attributes of this dataset. After the common 80/20 data split, we monitored the training process using categorical cross-entropy loss and the Adam optimizer with a learning rate of 0.001 and a batch size of 32, aiming to improve accuracy over 100 epochs. Figure 3 shows the training process started with the model performing poorly, with an accuracy below 10%. However, as training progresses, the accuracy gradually improves, reaching around 80% by the 50th epoch and surging to approximately 94% by the last epoch. tests on

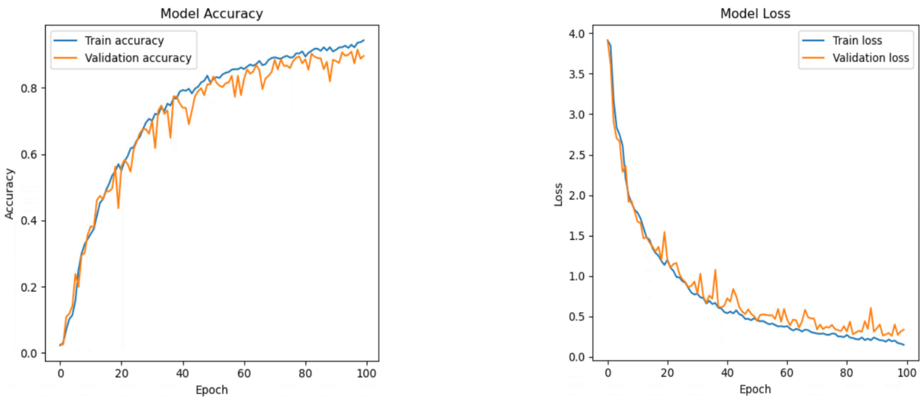

Fig. 3. Model accuracy and loss

unseen data showed promising results, with an accuracy of 88.75%, indicating good generalization capability. Despite initial challenges, the model demonstrated resilience and adaptability, showcasing the effectiveness of the approach and the potential of deep learning in handling complex sequential data tasks.

6 Conclusion

In this paper, we have devised a deep learning-based Arabic sign language recognition system. The present work developed a good model that translates

videos containing sign language gestures into Arabic phrases.
The system relies on an LSTM-based recurrent neural network that catches the temporal dependencies among a sequence of frames, and on the signer body keypoints extracted via the MediaPipe library. We experiment with the ArabSign dataset and reach 94% and 88.75% of train and test accuracy respectively. Although the obtained result is satisfactory, the present system version delivers a sentence from a short input video only. There is still a lot of room for improvement to the actual version of the system.
Refinement of the preprocessing stage in many aspects constitutes a predominant extension. Firstly, it would be interesting to precisely evaluate the impact of the framing approach on the subsequent stages. Additionally, the actual keypoint extraction module uses the standard version of the pose model of the MediaPipe tool, more investigation is needed to discern the most distinctive and influential landmarks in sign gestures. Focusing on more narrow and specific regions of interest for keypoint extraction could enhance the model's effectiveness and its robustness to noise. Another line of work that deserves further exploration could be the integration of the present model into an entire continuous real-time ArSL translator.

References

1. World health organization homepage. `https://www.who.int/ar/news-room/fact-sheets/detail/deafness-and-hearing-loss`, accessed: 2024/04/30
2. Adeyanju, I.A., Bello, O.O., Adegboye, M.A.: Machine learning methods for sign language recognition: A critical review and analysis. Intell. Syst. Appl. **12**, 200056 (2021), `https://doi.org/10.1016/J.ISWA.2021.200056`
3. Al-Jarrah, O., Halawani, A.: Recognition of gestures in arabic sign language using neuro-fuzzy systems. Artificial Intelligence **133**(1), 117–138 (2001), `https://doi.org/10.1016/S0004-3702(01)00141-2`
4. Al-Rousan, M.A., Assaleh, K., Tala'a, A.: Video-based signer-independent arabic sign language recognition using hidden markov models. Appl. Soft Comput. **9**(3), 990–999 (2009), `https://doi.org/10.1016/J.ASOC.2009.01.002`
5. Aldhahri, E., Aljuhani, R., Alfaidi, A., Alshehri, B., Alwadei, H., Aljojo, N., Alshutayri, A., Almazroi, A.: Arabic sign language recognition using convolutional neural network and mobilenet. Arabian Journal for Science and Engineering **48**(2), 2147–2154 (2023), `https://doi.org/10.1007/s13369-022-07144-2`
6. Aly, S., Aly, W.: Deeparslr: A novel signer-independent deep learning framework for isolated arabic sign language gestures recognition. IEEE Access **8**, 83199–83212 (2020), `https://doi.org/10.1109/ACCESS.2020.2990699`
7. Alzohairi, R., Alghonaim, R., Alshehri, W., Aloqeely, S.: Image based arabic sign language recognition system. International Journal of Advanced Computer Science and Applications **9**(3) (2018), `https://doi.org/10.14569/IJACSA.2018.090327`
8. Balaha, M.M., El-Kady, S., Balaha, H.M., Salama, M., Emad, E., Hassan, M., Saafan, M.M.: A vision-based deep learning approach for independent-users arabic sign language interpretation. Multim. Tools Appl. **82**(5), 6807–6826 (2023), `https://doi.org/10.1007/S11042-022-13423-9`
9. Boukdir, A., Benaddy, M., Ellahyani, A., Meslouhi, O.E., Kardouchi, M.: Isolated video-based arabic sign language recognition using convolutional and recursive

neural networks. Arabian Journal for Science and Engineering **47**(2), 2187–2199 (2022), https://doi.org/10.1007/s13369-021-06167-5

10. Elatawy, S.M., Hawa, D.M., Ewees, A.A., Saad, A.M.: Recognition system for alphabet arabic sign language using neutrosophic and fuzzy c-means. Educ. Inf. Technol. **25**(6), 5601–5616 (2020), https://doi.org/10.1007/S10639-020-10184-6
11. ElBadawy, M., Elons, A.S., Shedeed, H.A., Tolba, M.F.: Arabic sign language recognition with 3d convolutional neural networks. In: 2017 Eighth International Conference on Intelligent Computing and Information Systems (ICICIS). pp. 66–71 (2017), https://doi.org/10.1109/INTELCIS.2017.8260028
12. Hassan, M., Assaleh, Khaled andShanableh, T.: Multiple proposals for continuous arabic sign language recognition. Sensing and Imaging **20**(4) (2019), https://doi.org/10.1007/s11220-019-0225-3
13. Hayani, S., Benaddy, M., El Meslouhi, O., Kardouchi, M.: Arab sign language recognition with convolutional neural networks. In: 2019 International Conference of Computer Science and Renewable Energies (ICCSRE). pp. 1–4 (2019), https://doi.org/10.1109/ICCSRE.2019.8807586
14. Hisham, B., Hamouda, A.: Arabic sign language recognition using ada-boosting based on a leap motion controller. International Journal of Information Technology **13**(3), 1221–1234 (2021), https://doi.org/10.1007/s41870-020-00518-5
15. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997), https://doi.org/10.1162/neco.1997.9.8.1735
16. Kamruzzaman, M.M.: Arabic sign language recognition and generating arabic speech using convolutional neural network. Wirel. Commun. Mob. Comput. **2020**, 3685614:1–3685614:9 (2020), https://api.semanticscholar.org/CorpusID:219485742
17. Kim, Y., Baek, H.: Preprocessing for keypoint-based sign language translation without glosses. Sensors **23**(6), 3231 (2023)
18. Lugaresi, C., Tang, J., Nash, H., McClanahan, C., Uboweja, E., Hays, M., Zhang, F., Chang, C., Yong, M.G., Lee, J., Chang, W., Hua, W., Georg, M., Grundmann, M.: Mediapipe: A framework for building perception pipelines. CoRR **abs/1906.08172** (2019), http://arxiv.org/abs/1906.08172
19. Luqman, H.: Arabsign: A multi-modality dataset and benchmark for continuous arabic sign language recognition. In: 17th IEEE International Conference on Automatic Face and Gesture Recognition, FG 2023, Waikoloa Beach, HI, USA, January 5-8, 2023. pp. 1–8. IEEE (2023), https://doi.org/10.1109/FG57933.2023.10042720
20. Maraqa, M., Abu-Zaiter, R.: Recognition of arabic sign language (arsl) using recurrent neural networks. In: 2008 First International Conference on the Applications of Digital Information and Web Technologies (ICADIWT). pp. 478–481 (2008), https://doi.org/10.1109/ICADIWT.2008.4664396
21. Mohandes, M.A., Quadri, S.I., Deriche, M.A.: Arabic sign language recognition an image-based approach. In: 21st International Conference on Advanced Information Networking and Applications (AINA 2007), Workshops Proceedings, Volume 1, May 21-23, 2007, Niagara Falls, Canada. pp. 272–276. IEEE Computer Society (2007), https://doi.org/10.1109/AINAW.2007.98
22. Podder, K.K., Ezeddin, M., Chowdhury, M.E.H., Sumon, M.S.I., Tahir, A.M., Ayari, M.A., Dutta, P., Khandakar, A., Mahbub, Z.B., Kadir, M.A.: Signer-independent arabic sign language recognition system using deep learning model. Sensors **23**(16) (2023), https://doi.org/10.3390/s23167156
23. Yanqiong Zhang, X.J.: Recent advances on deep learning for sign language recognition. Computer Modeling in Engineering & Sciences **139**(3), 2399–2450 (2024), https://doi.org/10.32604/cmes.2023.045731