Deep Learning-Based Arabic Sign Language Recognition for Medical Emergency Communication

Shaden Alsuhayan
Dept. of Computer Engineering
Imam Abdulrahman Bin Faisal University
Dammam, Saudi Arabia
2220007144@iau.edu.sa

Haneen Alhabib
Dept. of Computer Engineering
Imam Abdulrahman Bin Faisal University
Dammam, Saudi Arabia
2220001259@iau.edu.sa

Mozoon Alkhalis
Dept. of Computer Engineering
Imam Abdulrahman Bin Faisal University
Dammam, Saudi Arabia
2220002873@iau.edu.sa

Haneen Alomri
Dept. of Computer Engineering
Imam Abdulrahman Bin Faisal University
Dammam, Saudi Arabia
2210003424@iau.edu.sa

Nora Aljomuh
Dept. of Computer Engineering
Imam Abdulrahman Bin Faisal University
Dammam, Saudi Arabia
2220004452@iau.edu.sa

Ritaj Alhamli
Dept. Computer Engineering
Imam Abdulrahma Bin Faisal University
Dammam, Saudi Arabia
2210002809@iau.edu.sa

Abstract— **This project explores the use of deep learning models for Arabic Sign Language (ArSL) recognition in medical emergency situations, where clear and fast communication is especially important. The goal is to support individuals who are deaf or hard of hearing during emergencies. A subset of emergency-related signs was selected from the KArSL-502 dataset, which includes a wide range of Arabic sign language words. The study focuses only on signs that are relevant to urgent medical communication. Five deep learning models were implemented and compared: ResNet, ResNet+BiLSTM, I3D, VideoMAE, and SignBart. Based on the experimental results, the SignBart model showed the most suitable performance for recognizing Arabic sign language in medical emergency contexts. This work aims to improve emergency communication by providing more practical and accessible solutions for sign language users.**

Keywords— Deep Learning, Artificial Intelligence, Sign Language Recognition, Arabic Sign Language (ArSL), KArSL-502

I. Introduction

Sign Language is the primary communication mean for deaf and hearing-impaired people, consisting of signs made using hand gestures along with facial expressions to convey meaningful sentences. There is no universal sign language as it varies across different countries and regions, such as The Word-Level American Sign Language (WLASL), The Russian Sign Language (SLOVO), The Arabic Sign Language (ArSL). In recent years, Sign Language Recognition (SLR) task has gained remarkable as Deep Learning (DL) models continue to improve. While significant research has been done on non-Arabic sign languages and some progress has been made on ArSL,

During a medical emergency, the ability to communicate medical-related information is critical and can have a substantial impact on patient outcomes. ArSL is essential to communicate in Arabic; however, the automated recognition of ArSL is underdeveloped, especially in the context of communicating during medical emergencies in real-time. The primary challenge in this area is developing models that can accurately interpret complex gestures that form a large part of the language of sign. This study uses a subset of the KArSL-502 dataset comprised of a wide variety of vocabulary. From this larger dataset, the subset that is most relevant to communicating during a medical emergency was selected.

Five advanced deep learning models were evaluated and compared in their ability to recognize ArSL in a medical emergency: VideoMAE, I3D, ResNet+BiLSTM, ResNet and SignBart. VideoMAE works using transformer-based architecture and employs a self-supervised manner to learn features both spatially and temporally from video data, which helps greatly in recognizing continuous-signs language gestures. I3D is an evolution of 2D Convolutional Networks; this new model allows capturing motion through time by extending the dimensions of the original models of time as the third dimension, ideal for action recognition problems, such as sign language. While ResNets has been known for solving the problem of vanishing gradients, the combination of ResNets with Bidirectional Long Short Term Memory networks (BiLSTM) provides ResNets with greater performance detail when extracting features from images without having to modify the architecture as previously applied. In addition, ResNets and BiLsMTMs help to understand the temporal context of the sign language video sequence. SignBART is a derivative of the Transformer architecture; the self-attention mechanism is applied to provide the deserved focus on the sequentially produced data to capture contextual relationships better, leading to more accurate recognition of continuous sign language.

II. Project Goals and objectives

The primary goal of this research is to develop an effective deep learning model for Arabic Sign Language recognition in the context of medical emergencies. The specific objectives of the study are:

1) To focus on a specific subset of KArSL vocabulary related to medical emergencies, the dataset is curated to include signs that are most relevant in urgent healthcare situations.

2) Evaluating different deep learning models for their effectiveness at recognizing KArSL vocabulary related to medical emergencies.
3) To provide comparative analysis and contrast results across different settings to show the model's ability to generalize.
4) Finally, determine the best deep learning model for its inclusion within a medical emergency communication system that will provide greater access and responsiveness to individuals utilizing KArSL for communication during emergencies.

III. Related Work

Alsulaiman et al. [1] introduced King Saud University Saudi Sign Language (KSU-SSL) dataset, the largest SSL dataset, consisting of 293 signs, 33 signers, and 145,035 samples recorded using RGB, infrared (IR), and mobile cameras across 10 domains includes healthcare, common, alphabets, verbs, pronouns and adverbs, numbers, days, kings, family, and regions. The authors proposed a novel framework by introducing a Convolutional Graph Neural Network (CGNN) architecture which is made up of small number of separable 3DGCN layers and augmented with a spatial attention mechanism, achieving an average accuracy of 97.25%. The dataset, however, was recorded with the use of colored gloves and painted hands to enhance the hand visibility which does not reflect real-world conditions. Furthermore, the authors have leveraged MediaPipe for perceiving human pose, hand tracking, and face landmarks on mobile devices that support feature extraction.

Al Khuzayem et al. [2] developed 'Efhamni', an android-based mobile application for Saudi Sign Language Recognition (SSLR) to translate isolated SSL to text and audio. The application was trained on KSU-SSL dataset which covers 16,000 videos of 40 words and the alphabet and numbers of the Arabic language, comprising 80 different signs performed by 40 signers, with each signer executing each sign five times. Their approach combines Convolutional Neural Network (CNN) with Bidirectional Long-Short Term Memory (BiLSTM), MediaPipe-based hand and body landmark extraction. The authors have trained variant models with different number of layers, batch sizes, epochs, and with data augmentation and without. Their best accuracy was obtained with three-layer model with data augmentation, 512 batch size, and 600 epochs. This best model yielding 99.98% training accuracy, 94.46% testing accuracy, 94.79% validation accuracy, precision of 94.61%, recall of 94.56%, an F1-score of 94.52%. The translation process took about 6 seconds on 30 frames resulting in 5 frames per second (FPS). However, the application is limited to isolated SSL, requiring the users to record or upload each word separately. Moreover, it is important to note that the version of KSU-SSL dataset used in this research differs from the 293-sign KSU-SSL utilized in Alsulaiman et al. [1] which explains the variation in dataset size.

Building on the need for more expressive communication systems, Elhassen et al. [3] developed KAU-CSSL, the first Continuous Saudi Sign Language dataset, including 5,810 medical-domain RGB video collected samples across 85 classes. The authors leveraged transfer learning by developing the KAU-SignTransformer model, a hybrid model integrating ResNet-18 for spatial, a Transform Encoder for temporal sequence modeling, and BiLSTM for better long-term dependencies. The model achieved 99.02% accuracy, 99% precision, 99% recall, and 99.01% F1-score in signer-dependent mode. However, in signer-independent mode, performance dropped, achieving 77.71% accuracy, 83.47% precision, 79.86% recall, and 78.30% F1-score, highlighting the difficulty in generalizing across variant signing styles, paces, and backgrounds. Additionally, limitations included short duration signs and occluded facial expressions caused by Niqab.

To combat the challenge of the data scarcity in ArSL recognition systems and bridge communication between deaf and hearing communities, Algethami et al. [4] conducted a pioneering study using a Time Convolutional Neural Network model for continuous sentence recognition in ArSL. Their research aimed at recognizing continuous Arabic sign language gestures from video sequences to understand the meaning of the sequence of signs rather than separate signs to allow normal, time-dependent communication. The researchers first developed a custom dataset, ArSLSR, consisting of 30 common sentences where MediaPipe technology was utilized to extract 258 key motion points; this reduced computational cost and enhanced generalizability. The TCN model performed very well and achieved 99.5% accuracy on ArSLSR and 99% on the external ArabSign dataset, outperforming the attention-enhanced RNN-BiLSTM model in terms of both computational efficiency and speed. Although the enhanced model achieved a similar 99% accuracy on ArSLSR, its performance on the external dataset was poor, 26%, before enhancements, thereby further underlining TCN as a potential promising system.

The article conducted by S. Aly and W. Aly [5] proposes a novel framework called DeepArSLR shown in Fig. 1, which combines DeepLabv3+ for semantic segmentation, a single-layer Convolutional Self-Organizing Map (CSOM) for hand shape feature extraction, and Bi-directional LSTM (BiLSTM) for classifying the sequence of extracted feature vectors. This architecture was tested using the ArSL dataset, which utilizes 23 gestures captured by three users. The DeepLabv3+ model alone achieved a 94.2% mean Intersection over Union (mIoU), demonstrating its ability to segment hands accurately. The proposed framework achieved an average accuracy of 89.5% using DeepLabv3+ hand semantic segmentation, which outperformed all state-of-the-art methods. The limited number of 3 users demonstrating the gestures is considered a limitation since it is likely that accuracy drops when exposed to unseen signers. Furthermore, the authors highlighted that their work can be extended to explore a wider scope, such as solving continuous sign language recognition problems for Arabic and similarly for other languages.

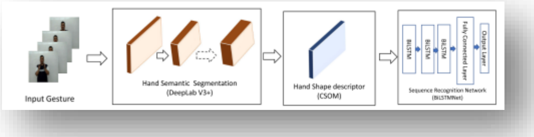

Fig. 1. Overview of the Framework process from input gestures to final classification. Reproduced from [5]

Additionally, Bencherif et al. [6] introduce an Arabic Sign Language Recognition System that combines OpenPose for selecting 2D hand and body key areas and a Skeletal CNN for classifying static and dynamic signs. The authors found that there is a lack of diversity in the existing ArSL public dataset; thus, they built a new dataset that comprises 80 Arabic signs acted by 40 signers. The proposed framework achieved nearly 90% overall accuracy, demonstrating strong performance in both static and dynamic sign recognition. However, the authors mentioned minor difficulties they faced during conducting this study, which include limited real-time speed, faced difficulties in recognizing points in blurred frames, and differentiating similar signs. For future work, the authors encouraged fixing these discussed limitations by providing better recognition and enhanced real-time performance.

Expanding on these developments, ASLDetect [7] is a model for Arabic Sign Language recognition that includes a U-Net-style segmentation component in addition to ResNet34, providing help for challenges that involve detailed backgrounds and lighting adjustments. The researchers trained ASLDetect using two datasets, ArASL2018 and ArASL2021, through transfer learning and selective augmentation with 99.35% and 86.84% accuracy respectively. It outperformed existing models like ResNet34, T-SignSys and UrSL-CNN and is considerably more robust and adjustable to the real world. However, it only recognizes alphabetic signs.Similarly, the Saudi Sign Language recognition system [8] used a Convolutional Neural Network (CNN) for a training set of 40 static Saudi signs in different lighting and background conditions with 99.47% testing accuracy and real-time recognition capabilities on desktop and mobile. However, it only analyzed static gestures and didn't recognize any dynamic gestures or emergency gestures.

Sign language recognition has become a critical key in assistive technology, aiming to help and bridge the communication gap for individuals with hearing and speech difficulties. researchers offered an advanced Arabic Alphabet Sign Language recognition system using transfer learning and transformer architectures such as Vision Transformer (ViT) and Swin Transformer while implementing clear steps for the system. Their study achieved almost 99% on both training and testing, using Arabic Sign Language datasets, demonstrating how deep learning is beneficial and performs wonderfully for accurate gesture interpretation and analysis. However, the limitation of this study is focusing mainly on alphabet-level recognition rather than real-life urgent scenarios situations [9].

To address the shortage of translators for a language used in Saudi Arabia, Noor et al. [10] developed a real-time hybrid system for Arabic (ArSL). This system is based on a multi-layered approach combining CNN and LSTM to capture both spatial and temporal references. To achieve this, the Google MediaPipe framework was used to implement 21 key points, which effectively demonstrated the efficiency model in real time. To support this implementation, a custom dataset of 4,000 images of ten static words (for CNN training) and 500 video segments of ten dynamic words (for LSTM training) was applied. The hybrid model demonstrated strong performance, with the fully independent CNN achieving 94.40% in static challenges and the fully independent LSTM achieving 82.70% in dynamic challenges.

According to Sok et al study developed an Emergency Care System using Deep Learning techniques to support people with disabilities in Cambodia. Their model used both Convolutional Neural Networks (CNN) and IoT integration to identify emergency gestures as illustrated in the figure below, and directly alert hospitals and caregivers of the emergency. This study highlights the potential of the emergency context but lacks linguistic sign language such as Arabic [11].

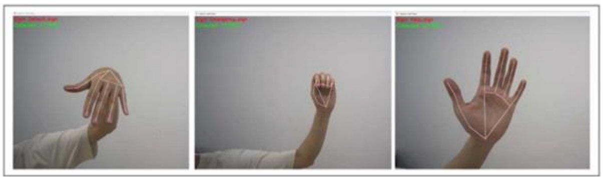

Fig. 2. Sample Hand Gestures Representing SOS Signals — (a) Default Gesture, (b) Help Gesture, and (c) Emergency Gesture reproduced from [11]

[12] built one of the first datasets for emergency use, collecting 824 short videos of eight Indian Sign Language gestures such as help, doctor, and accident. Their CNN-LSTM model achieved around 96% accuracy, proving that emergency signs can be recognized effectively when the dataset is clear and focused.

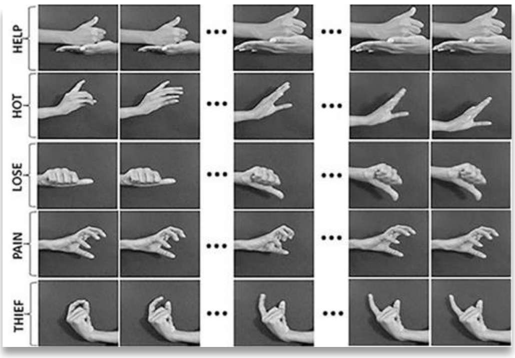

Fig. 3. Visual examples of emergency gestures in Indian Sign Language (ISL) from [12])

[13] developed SWL-LSE, a dataset of Spanish Sign Language gestures related to health and emergency contexts. They collected around 8,000 samples from 124 participants through an online platform and achieved about 92% accuracy using a pose-based deep learning model (ST-GCN).

[14] Building on this idea, [14] introduced the Korean Disaster Safety Information Sign Language Dataset, which contains thousands of videos covering emergency situations such as fires, earthquakes, and evacuations. Transformer-based models were used to develop an effective translation system for real-world disaster communication.

Finally, [15] presented ArabSign, the first continuous Arabic sign language dataset, which includes more than 9,000 samples collected using RGB, depth, and skeleton data.

TABLE I. SUMMARY OF KEY STUDIES RELATED TO DATASETS.

Study	*Language*	*Domain Focus*	*Dataset Size*	*Model Used*	*Accuracy / Key Results*
[12]	Indian	Emergency gestures	824 videos (8 signs)	CNN–LSTM	~96%
[13]	Spanish	Health & emergency	8,000 samples (300 signs)	ST–GCN	~92%
[14]	Korean	Disaster / emergency	Thousands of videos	Transformer	BLEU ≈ 32,

Study	*Language*	*Domain Focus*	*Dataset Size*	*Model Used*	*Accuracy / Key Results*
					WER ≈ 0.48
[15]	Arabic	General communication	9,000+ samples (50 sentences)	Encoder–Decoder GRU	WER ≈ 0.50

IV. Data Acquisition and Preprocessing

Due to the limitation in accessing Saudi Sign Language datasets with the project timeline, the KArSL-502 Arabic Sign Language dataset was selected as an alternative [16]. This dataset is publicly available and was obtained from Kaggle. KArSL-502 is a large video dataset for word-level Arabic Sign Language (ArSL) consisting of 502 isolated words (signs) collected using Microsoft Kinect V2. Each sign of the dataset was performed by three professional signers, with 50 repetitions per signer, resulting in a total of 75,300 samples (502 × 3 × 50).

The dataset is organized into three main directories labeled "01", "02", and "03", corresponding to the three signers. Within each directory, a nested folder with same signer identifier exists. Each signer folder holds two subfolders: "train" and "test", representing the original dataset spilt. Within each split, there are 502 folders, each relates to a specific sign (e.g. "0001", "0002" … "0502"). Under the "test" split, each sign folder contains 8 repetition folders, while the remaining repetition folders are located under the "train" split. Each repetition folder holds a sequence of RGB image frames representing the sign video.

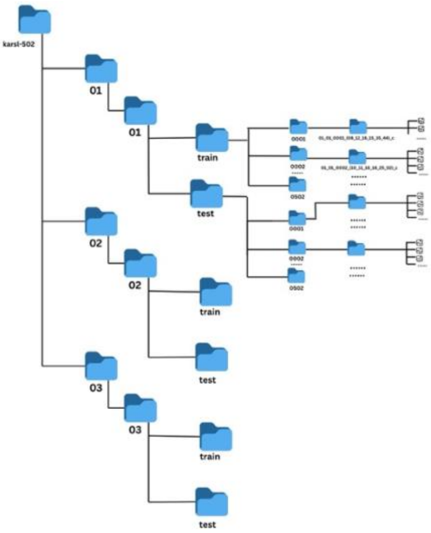

Fig. 4. KArSL-502 Dataset Structure Diagram

To remain aligned with the project main motivation, a subset of signs related to medical emergency and urgent communication was selected from the full KArSL-502 dataset. A total of 131 medically relevant signs were manually chosen based on their importance in critical scenarios. For this subset, each sign was performed by three signers and repeated 50 times, resulting in 19650 samples (131 × 3 × 50).

To manage and utilize the selected subset, a custom Excel file was created (see Appendix A). It served as the label-mapping reference and included the following columns:

- **ClassIndex:** A sequential class label ranging from 0 to 130, used as the final target label during model training and testing.
- **SignFolder:** We have added this column which contains the formatted original SignID (71,72, etc.) to a 4 digit, zero-padded version of ID (0071,0072. etc.). This has been done to match the actual folder names in the dataset directory structure, providing direct mapping between the Excel sheet and corresponding sign folder.
- **SignID:** The original sign ID as defined by the dataset author.
- **Sign-Arabic:** The Arabic meaning of the sign.
- **Sign-English:** The English translation of the sign, can be for clarity to non-Arabic readers.

After uploading the dataset to Kaggle, we explored and validated its structure, confirming the nested directory format for each signer and identifying the correct paths to access training and testing repetitions. A code was implemented to load the mapping file, convert folder IDs to the required zero-padded format, and search the dataset for available repetitions of selected signs. We also verified the presence and readability of RGB frame sequences for each repetition and visually inspected sample frames. Fig. 5shows sample frames extracted from a single repetition of a randomly selected sign.

Sign: 0101 - The sign in English: blood analysis - The sign in Arabic: مرد ليلحت
frame 1
frame 2
frame 3
frame 4
frame 5
frame 6
frame 7
frame 8
frame 9
frame 10

Fig. 5. Ten RGB frames extracted from a single repetition of a randomly selected sign in the KArSL-502 dataset.

V. EXPERIMENTS

This section will present a sequence of experiments that were conducted systematically to explore different deep learning models for video-based recognition. The evaluated approaches are explained in a sequential manner, starting from the simplest architecture and moving toward more sophisticated models. Through this structured process it allowed us to perform a clear analysis of multiple models and motivates the selection of our final model based on practical evidence.

A. ResNet

ResNet is implemented as a single model to evaluate the success of spatial feature learning without explicit temporal modeling. It's widely used and known as a deep convolutional neural network that's could of extracting rich and complex visual representations from single frames using residual connections. It benefits our project since it enables us to start stable training of deep architecture and mitigate the vanishing gradient problem. By using ResNet alone, the experiment isolates the contribution of frame-level spatial information and serves as a strong baseline for our second experiment ResNet and Bi-LSTM.

1) Signer-Independent Experiment

In the signer-independent experiment using the ResNet-only model, evaluation is performed on a completely unseen signer, ensuring strict separation between training and testing subjects. The model is trained using frame-level images extracted from a subset of signers, any frame belonging to one signer is entirely excluded from testing. In this setup, ResNet processes each frame alone and learns how to differentiate spatial features related to hand shape, posture and visual appearance. This approach of evaluation is really challenging due to the absence of both signer overlapping and motion modeling, achieving 26.06% as an overall accuracy. In addition, the signer-independent ResNet experiment serves as a strong baseline that shows the limitations of frame-based recognition under high inter-signer variability and highlights the need to use temporal modeling which leads us to our second experiment implementing ResNet + Bi-LSTM.

2) *Mixed Signer Experiment*

In the mixed signer experiment, ResNet model is evaluated under a partially signer-independent setting, where training and testing samples may come from the same signers but from different video instances. Specifically, 1st and 2nd train folders from the data used in training, and 3rd folder of train used in validation. Moreover, the test is being held on all 3 test folders. By implementing this approach, the model achieved around 82% illustrating that the model learned signer-specific spatial features, resulting in higher performance compared to the fully signer-independent setting due to reduced inter-signer variability. Nevertheless, this result doesn't represent real-world deployment, this experiment provides a baseline for assessing the contribution of temporal modeling in comparison to the ResNet + Bi-LSTM architecture.

B. ResNet and Bi-LSTM

ResNet and Bi-LSTM architecture are excellent candidates for our project because it wonderfully models both spatial and temporal information present in our frame-based dataset. ResNet acts as a powerful spatial feature extractor, capturing outstanding discriminative visual patterns from each frame while benefiting from residual connections that allow deeper networks to be trained without degradation. These deep representations are mainly important for handling variations in the environment, such as different angles. backgrounds and illumination. In addition to these impressive frame-level features, the Bi-LSTM models show temporal dynamics of the sequence by learning and analyzing dependencies across time in both forward and backward directions. This bidirectional architecture enables the network to analyze contextual information from the entire sequence rather than relying solely on individual frames. Therefore, the combination provides a balanced framework that captures fine-grained visual details while preserving motion and temporal consistency, making it a strong and widely known choice for sequence-based recognition tasks.

1) *Signer-Independent Experiment*

In the signer-independent experiment, the model is evaluated on completely unseen data during training to ensure a strict separation between training and testing subjects. Specifically, the ResNet and Bi-LSTM architecture is trained using video sequences from a subset of signers, while all samples from one signer are excluded completely from the testing. Each frame is represented as a fixed-length sequence of frames. The spatial features are extracted from the dataset frame using a pretrained ResNet backbone, and temporal dependencies are modeled using a bidirectional LSTM to capture motion patterns across the entire sequence. The model achieved under these conditions 45.38% as a test accuracy, which may look modest at first glance, but it's vital to interpret that the result in the context of a 131-class signer-independent classification problem. Under a random-guess baseline, the expected accuracy would be below 1%, making the achieved performance more than 45× higher than chance level. This evaluation setting is designed to assess the model's ability to generalize across different signers and ensure a fair examination. As a result, the performance under this setup reflects a realistic scenario.

2) *Mixed Signer Experiment*

In the mixed experiment, the evaluation follows a partially signer-independent setting, where the 1st and 2nd train folders from the data used in training, and the 3rd folder of train used in validation. Moreover, the test is being held on all 3 test folders. This setup allows the model to learn signer-specific visual and motion characteristics during training while still being evaluated on unseen samples. As expected, this setting results in higher performance, achieving around 83.64% compared to the fully signer-independent experiment, since inter-signer variability is reduced and the task focuses more on recognizing sign patterns rather than adapting to unseen execution styles.

C. I3D (Inflated 3D Convolutional Network)

I3D is a deep learning model built to understand videos specifically. Unlike other DL architectures that process frames independently, I3D utilizes 3D convolutions to learn information, for example, hand shape, position, and posture. Additionally, the model learns temporal information that includes motion progressing across frames; these features are essential when it comes to the sign language recognition task, since it not only focuses on the hand gesture, but also on how they move over time. I3D was pretrained on large-scale video datasets, which allowed us to start with effective motion-aware features, and then we can fine-tune for our chosen sign language dataset. Moreover, I3D has been extensively used as a great baseline for gesture recognition.

1) *Signer-Independent Experiments*

First, we begin with evaluating the I3D model on a pure signer-independent split conducted on a subset of the KARSL-502 that focuses on emergency signs, the training is on signer 1 and 2 "train" files, and the testing on the completely unseen signer 3. Our first attempt was conducted to study and explore the model's abilities. So, we did not use any data augmentation techniques, and a strict data split that does not allow any overlapping. After training the model, we evaluated and considered it as our controlled baseline. It achieved an overall accuracy of 50.4%. This accuracy demonstrates that the model can learn useful information from the dataset, proving its capability, and is worth the time and effort required. In the second attempt, we added to the discussed attempt data augmentation to the training pipeline only; to ensure correct results, we made sure the validation and test data are not augmented. We applied simple and straightforward augmentations that will not affect and change the meaning of sign language gestures, yet they are still valuable in terms of helping the model to generalize better and to increase the dataset amount. The data augmentation added includes horizontal flip, random brightness change, random contrast change, and random temporal crop.

However, the model achieved 48.8% test accuracy, which is extremely similar to the first baseline. This attempt showed that data augmentation alone is not enough to increase the model's evaluation metrics. It showed that signer-independent sign language recognition is still a challenging task, especially with a small dataset.

2) Mixed Signer Experiment

Building on earlier signer-independent experiments, we choose to continue with evaluating the model's performance on a different data split with additional improvements. The training is on signer 1 and 2 "train" files, the validation on signer 3 "train" file, and the testing is on all three "test" files for signer 1,2, and 3. This approach is considered semi-signer independent. This split was chosen to help us deepen our understanding of how the model behaves when signers are more exposed to the model, but still not to training itself; signer 3 is not a part of training at all. Using this data split, the pre-trained I3D model was fine-tuned with five epochs, which is sufficient because I3D is a large and heavy model, so extending the number of epochs would ultimately lead to overfitting. We chose a batch of size two, which provided steady training. For the optimizer, we used the Adam optimizer, which provides adaptive learning and a stable convergence, which is particularly important, especially with training a deep 3D convolution network. The learning rate was 1e-4, so it ensures that the model can be fine-tuned to our emergency sign language dataset without losing any useful knowledge it has from being pre-trained. For the regularization methods, we used a weight decay of 1e-4, which penalizes large weights to prevent overfitting. Also, for data augmentation, we intentionally removed the horizontal flip function used before, because we suspected that the flip may alter and change the meaning of the sign. Nonetheless, all other data augmentation functions used in the "*Signer-Independent Experiments*" section were also used here. After training, we saved the best model based on validation accuracy. And then evaluated it on the test set. The training and evaluation process showed that the model has a stable and useful learning process. Throughout the five epochs, the model's training loss kept decreasing, starting from 1.2361 to 0.0417. This proved that the model is capable to learn the pattern in data. Validation accuracy also improved overall and achieved its highest value at the final fifth epoch. For the final evaluation, the accuracy on the test set reached 85.46%. The precision, recall, and F1-score are also all very similar and consistent in value, which proves the model's ability in performing well across many different sign classes and does not focus on only a few simple signs. To further show the model's capabilities, we constructed a confusion matrix reported in the Appendix. The dark blue points that construct the diagonal line illustrate how many times the model classified each class correctly. Overall, the I3D model performed impressively on both the signer-independent and the mixed signer experiments.

D. VideoMAE

The next major advancement in our research has been the implementation of a Video Masked Autoencoder (VideoMAE), which uses Vision Transformers (ViTs), as opposed to the traditional convolutional neural network (CNN) approach. The self-supervised training method used in VideoMAE is called tube masking, where a random portion of the spatiotemporal space of a video is masked out, and then the model will have to predict the masked portions using the surrounding frames before the mask was applied. Through this method, VideoMAE learns to encode robust motion dynamics and temporal relations.

We employed a VideoMAE backbone pretrained on the Kinetics-400 dataset and fine-tuned it for our Sign Language Recognition task. To mitigate overfitting and ensure the model generalizes well to the variability inherent in real-world recordings, we implemented a specific Data Augmentation pipeline during the training phase. This included applying random rotation (±10 degrees) to account for variations in camera tilt, utilizing color jitter (brightness and contrast factor of 0.2) to simulate different lighting conditions, and employing random resized cropping (scale 0.8–1.0).

1) Signer-Independent Experiment

We implemented a strict Signer Independent evaluation procedure in order to assess the ability of the models to generalize to a new singer not seen during training. The two original signers' (01 and 02) data was used exclusively to train the model, while Signer 03's data was not included in training and was exclusively reserved for evaluation of how well models performed with an entirely new signer. By creating this distinct separation of the training and evaluation data, we prevent the models from memorizing the specific visual features or backgrounds associated with a signer. The results of the evaluation of the Signer Independent Evaluation revealed an extremely high level of test accuracy, 91.41% of the VideoMAE model.

2) Mixed Signer Experiment

In this mixed signer experiment, the VideoMAE Model was assessed under a partially signer-independent scenario. Training data from the VideoMAE Model included the training folders from Signers 01 and 02, with Signer 03's Train folder used for validation. The testing phase for this experiment included all three testing folders. This method produces an outstanding 96.82% accuracy of the model's learning ability and the ability to adapt to individual signer characteristics due to the model having learned signer-specific spatiotemporal patterns. This resulted in higher performance compared to the fully signer-independent setting due to reduced inter-signer variability. Nevertheless, while this result doesn't represent a strictly unseen-user deployment, this experiment provides a baseline for assessing the model's maximum learning capacity when adapting to characteristics of known users.

VI. MODEL DEVELOPMENT AND TRAINING

A. Data Splitting Approach

The original KArSL dataset is organized into signer-specific directories, where each signer contains predefined train and test subfolders. However, this original split is not suitable as it does not provide a validation set. To address this limitation, the original data split was reorganized as follows:

- **Training set:** Samples from the original train folders belonging to Signer 01 and Signer 02
- **Validation set:** Samples from the original train folder belonging to Signer 03
- **Test set:** Samples from the original test folders belonging to all signers and not used during training or validation

This reorganization ensures no data leakage between the training, validation, and test sets. The validation set was utilized exclusively for model selection and early stopping, while the test set was kept totally unseen and reserved for final performance evaluation.

Such a splitting approach is particularly significant for sign language recognition tasks, where variation in signing style, execution speed, hand dominance, and motion patterns across different signers can notably influence model generalization. By prioritizing signer-independent testing, the evaluation process more accurately reflects real-world usage scenarios.

B. Proposed Methodology

The proposed methodology follows a skeleton-based sign language recognition pipeline, focusing on efficiency, privacy protection, and suitability for real-time scenarios in medical emergency communication.

Sign video frames were processed using MediaPipe framework, a platform developed by Google for extracting body poses and hand landmarks. This step converts raw video frames into compact skeletal representations, decreasing input dimensionality while preserving essential motion information.

Due to the high computational cost associated with MediaPipe extractions, especially when processing long video sequences repeatedly across multiple signers, the number of repetitions was intentionally reduced. Specifically, 10 repetitions were extracted from the training folder for each signer, while 8 repetitions were extracted from the test folder for all signers. This controlled extraction strategy allowed the pipeline to be conducted within a limited timeline.

The extracted landmarks were organized into fixed-length temporal sequence and fed into a transformer-based recognition model (SignBart). The model learns spatiotemporal patterns in skeleton-based motion and maps directly to medical-emergencies sign classes.

Each extracted repetition was represented as a serialized sequence of skeletal keypoints. Each frame contains 75 two-dimensional landmarks, corresponding to body pose, left-hand, and right-hand joints encoded using (x, y) coordinates. An instance of the extracted skeleton keypoints for a single frame is shown in Fig. 6, illustrating the spatial distribution of landmarks.

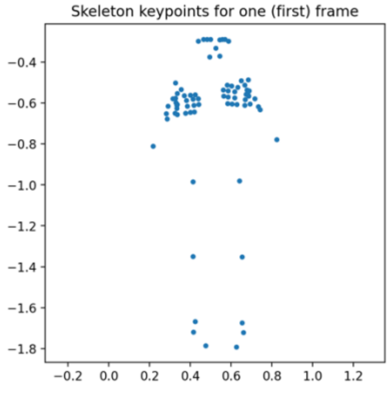

Fig. 6. Example of extracted skeletal keypoints for a single frame showing body pose, left-hand, and right-hand landmarks.

To capture temporal dynamics, joint trajectories were analyzed across sequential frames. Fig. 7demonstrates the behavior of randomly selected join x-coordinates over time, highlighting motion patterns captured during sign execution.

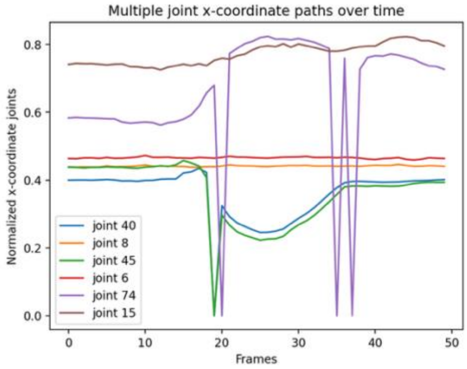

Fig. 7. Some joints x-coordinates trajectories across consecutive frames

Since repetitions vary in length, all sequences were standardized to a fixed temporal length of 50 frames. Sequences longer than this length were truncated, while shorter ones were padded by repeating the last frame. The result standardized representation is visualized in Fig. 8, where skeletal keypoints are shown across the 50 frames. Different colors are used to distinguish joints.

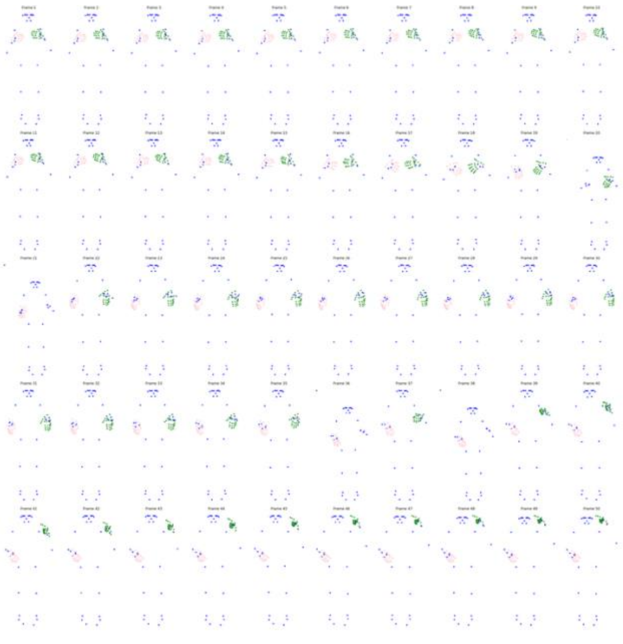

Fig. 8. Skeleton visualization across 50 frames for a sample sign

C. Proposed Architecture

The proposed system is based on the SignBrat model [17], a transformer-based encoder-decoder architecture originally proposed for sign language recognition using skeletal keypoints Each input sample consists of a temporal sequence of 50 frames, where each frame contains 75 two-dimensional keypoints, representing 33 body pose landmarks, 21 left-hand landmarks, and right-hand landmarks.

Within the SignBart architecture, the input skeletal keypoints are internally projected into a latent embedding space and combined with positional encodings to keep temporal order information before transformer-based processing.

An attention mask was used for each sequence to ensure that padded frames do not influence model learning during training by enabling the transformer to focus only on meaningful temporal information. The encoder captures spatiotemporal dependencies across frames, modeling both hand motion and body posture dynamics, while the decoder aggregates the encoded representations into a fixed-length embedding for classification. The original SignBart classification head was adapted to output 131 classes, matching the selected medical emergency subset. The overall architecture is illustrated in Fig. 9.

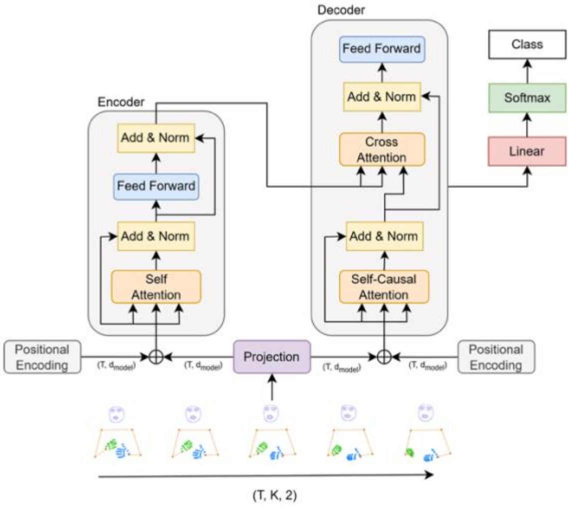

Fig. 9. Overview of the SignBart encoder–decoder architecture, adapted from [16]

D. *Training Configuration and Validation Strategy*

The proposed model was trained from scratch. Training and validation were both designed to ensure stable convergence and to prevent overfitting. Model optimization was performed using a mini-batch training scheme, with validation conducted after each epoch to ensure early stopping and best model selection.

Training was implemented using PyTorch deep learning framework. The implementation follows the official SignBart codebase released by the authors, which was adapted to support the selected subset and the modified classification head.

TABLE II. SUMMARY OF TRAINING AND VALIDATION SETUP

Framework	PyTorch
Optimizer	AdamW
Learning Rate	3×10^{-4}
Weight Decay	1×10^{-2}
Batch Size	64
Maximum Epochs	200
Learning Rate Scheduler	Cosine Annealing
Gradient Clipping	Max gradient norm = 1.0
Loss Functio	Cross-Entropy Loss
Validation Frequency	After each training epoch
Early Stopping	Patience of 15 epochs
Model Selection	Best validation accuracy checkpoint

Model parameters were optimized using the AdamW optimizer, which is well suited for transformer-based architectures due to its improved regularization behavior as it decouples weight decay from gradient update and apply it directly to the weights. A Cosine Annealing learning rate scheduler was applied as well. Gradient clipping with a maximum norm of 1.0 was applied to stabilize training and avoid exploding gradients. The cross-entropy loss function was used for multi-class classification.

Validation performance was monitored after each training epoch. Early stopping was applied when the validation accuracy failed to improve for 15 consecutive epochs, and the model checkpoint achieving the highest validation accuracy of 0.6748 was saved for final evaluation.

VII. EVALUATION AND ANALYSIS

This section evaluates the experiments conducted and the chosen model's results. Our experiments were mainly on ResNet+BILSTM, VideoMAE, I3D, and the proposed model, MediaPipe+SignBart. All the models were evaluated with Accuracy, Precision, Recall, and the Macro F1-score. We choose these specific metrics to quantify the overall classification correctness and to verify that the performance is balanced across all classes. These metrics were chosen because accuracy alone is not good enough for this complex task, and we wanted to ensure that the performance is not biased toward certain signs.

Fig. 10 illustrates the comparison of all models with the chosen evaluation metrics. To ensure a fair evaluation, all the models were implemented on the same data split and tested on the test set.

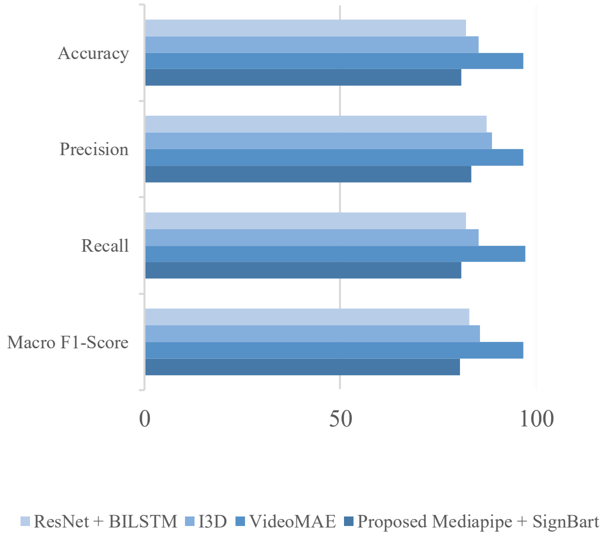

Fig. 10. Performance comparison of evaluated models using accuracy, precision, recall, and macro F1-score.

The VideoMAE model achieved around 96–97% on all metrics, which is the highest score across all experiments. It showed outstanding performance, showing how a transformer-based model successfully works with video frames. Also, I3D came in the second place with nearly 85-88% across all metrics. This is because it utilizes all video data. Additionally, the ResNet+BILSTM model also reaches high results, especially in precision and recall. This suggests that combining the spatial and temporal features is a good option for working with a sign language recognition task. These encouraging results show that the models trained on RGB frames can have very strong performance with this setup, also, with a limited number of signers.

However, these results must be interpreted carefully because the dataset is much smaller than other benchmarks in the field of sign language, especially that it is limited to only three signers, which helps video-based models to produce great results by learning visual data including the background and the signer's appearance, that are not actually related to the sign itself.

Despite the fact that the proposed MediaPipe+SignBart model achieves less than the other models, it consistently achieves around 80% across all evaluation metrics. It is worth noting that the proposed model was implemented under harder conditions. For example, instead of learning directly from video frames, it really depends on only the skeleton-based keypoints taken using MediaPipe. This technique forces the model to learn the motion and the structure of the sign itself and also ignoring meaningless details, which makes it better for generalization. Also, due to hardware limitations, the proposed model was trained and tested on an even smaller subset of the chosen dataset compared to the other models. Regardless of these extra restrictions, our proposed model still preserves its stable and very promising performance across all of the evaluation metrics, which shows and proves that it can learn the actual gesture patterns.

LIMITATIONS

Limited training data size: although it enables faster training and practical deployment within the project deadline, the training samples were reduced which reduces the overall performance as well.

Class balancing constraints: efforts were made to balance the dataset across the selected emergency-related signs. As a result, this balancing process may have caused some signs to be underrepresented, which might limit the overall model's ability.

Restricted input modality: the selected skeleton-based model does not utilize RGB video input. Consequently, certain features, for example hand shape details, may be lost.

Training from scratch: our selected model was trained entirely from scratch without the use of privileges such as pretrained weights.

Controlled recording conditions: experimental results were obtained using data collected under controlled conditions, including fixed lighting, consistent backgrounds and professional signers. It might affect the performance of real-world deployment scenarios.

CONCLUSION AND FUTURE WORK

In this project we experimented with different deep learning models for recognizing Arabic Sign Language, with a focus on medical emergency situations. A subset of 131 emergency-related signs was selected from the KArSL-502 dataset to better represent real and critical communication scenarios. The models were tested to see how well they could learn both visual and motion information, especially when evaluated on signers that were not seen during training. The results showed that frame-based models like ResNet do not perform well on their own, particularly in signer-independent experiments. When temporal information was added, the performance improved, but the best results came from models designed specifically for video data. Also I3D showed great potential in learning spatial and temporal features with an overall accuracy of 85%.VideoMAE achieved the highest accuracy at around 96%, showing that transformer-based models are very effective at learning complex sign movements over time. A skeleton-based approach using MediaPipe and SignBart was also tested and we found that it is the best model due to many factors such as its strong pipeline, and that it can generalise extremely well compared to the other models, because it neglect all useless information and focus solely on the sign gesture, this is mainly because of avoiding using raw video data.

Overall, the experiments confirm that modelling both spatial and temporal information is essential for Arabic Sign Language recognition, especially in emergency contexts. For future work, the system can be improved by increasing the number of emergency-related signs, including more signers to enhance generalization, and performing more detailed real-time performance analysis.

ACKNOWLEDGMENT

We would like to express our sincere thanks to Dr. Noor Felemban for her continues support, guidance and encouragement throughout this project. And we would also like to thank the researchers whose valuable work and contributions laid the foundation for this project.

Appendix I

Additional Materials

The following additional material is provided:

- An Excel file (.xlsx) containing Data_Extraction_Table used during the literature review process.

Appendix II: Confusion Matrices

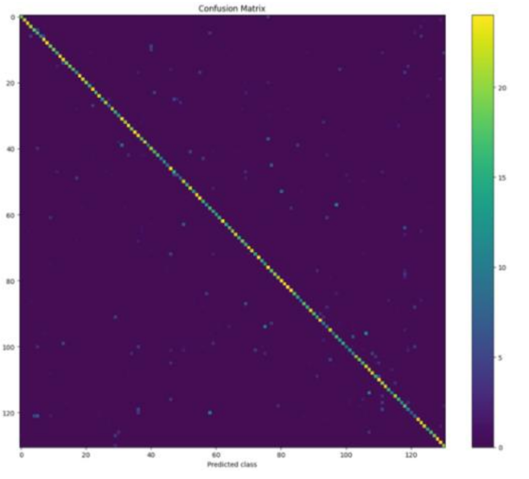

Figure II.1 Confusion matrix of the proposed Mediapipe+SignBart model on the test set.

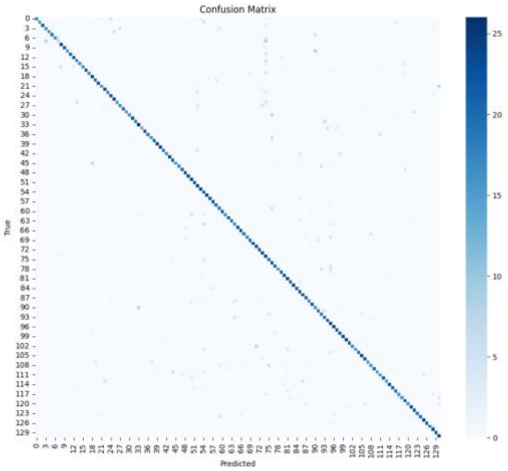

Figure II.2 Confusion matrix of the I3D model on the test set.

Name (Team Leader): Shaden Alsuhayan

Student ID: 2220007144

Group No.: 6

Name (2nd member): Haneen Alhabib

Student ID: 2220001259

Group No.: 6

Name (3rd member): Mozoon Alkhalis

Student ID: 2220002873

Group No.: 6

Name (4th member): Haneen Alomri

Student ID: 2210003424sa455

Group No.: 6

Name (5th member): Nora Aljomuh

Student ID: 2220004452

Group No.: 6

Name (6th member): Ritaj Alhamli

Student ID: 2210002809

Group No.: 6

Mention in one Paragraph the new skills you have learned while working on this project:

While working on this project, we gained practical experience in developing deep learning model and deeper understanding of sign recognition. We learned how to process videos and skeletal data, implementing and train different models using PytTorch, and evaluate their performance under different settings. This project also improved our writing abilities, critical thinking and our teamwork skills.

References

[1] M. Alsulaiman et al., “Facilitating the communication with deaf people: Building a largest Saudi sign language dataset,” Journal of King Saud University - Computer and Information Sciences, vol. 35, no. 8, p. 101642, Sep. 2023, doi: https://doi.org/10.1016/j.jksuci.2023.101642.

[2] L. Al Khuzayem, S. Shafi, S. Aljahdali, R. Alkhamesie, and O. Alzamzami, “Efhamni: A Deep Learning-Based Saudi Sign Language Recognition Application,” Sensors, vol. 24, no. 10, p. 3112, Jan. 2024, doi: https://doi.org/10.3390/s24103112.

[3] S. Elhassen, K. L. Al, A. Alhothali, O. Alzamzami, and N. Alowaidi, “Continuous Saudi Sign Language Recognition: A Vision Transformer Approach,” arXiv.org, 2025. https://arxiv.org/abs/2509.03467 (accessed Nov. 13, 2025).

[4] N. Algethami, R. Farhud, M. Alghamdi, H. Almutairi, M. Sorani, and N. Aleisa, “Continuous Arabic Sign Language Recognition Models,” *Sensors*, vol. 25, no. 9, art. 2916, May 2025, doi:10.3390/s25092916.

[5] S. Aly and W. Aly, “DeepARSLR: a novel Signer-Independent deep learning framework for isolated Arabic sign language gestures recognition,” IEEE Access, vol. 8, pp. 83199–83212, Jan. 2020, doi: 10.1109/access.2020.2990699.

[6] M. A. Bencherif et al., “Arabic sign language recognition system using 2D hands and body skeleton data,” IEEE Access, vol. 9, pp. 59612–59627, Jan. 2021, doi: 10.1109/access.2021.3069714.

[7] N. Alasmari and S. Asiri, “ASLDetect: Arabic sign language detection using ResNet and U-Net like component,” *Scientific Reports*, vol. 15, Article 18012, 2025, doi: 10.1038/s41598-025-01588-w.

[8] A. Al-Obodi *et al.*, “A Saudi Sign Language Recognition System based on Convolutional Neural Networks,” in *Proc. 2020 Int. Conf. Advances in Science, Engineering and Technology (ICASET)*, 2020, pp. 1–6, doi: 10.1109/ICASET.2020.9350868.

[9] Mazen Balat, Rewaa Awaad, H. Adel, A. B. Zaky, and S. A. Aly, “Advanced Arabic Alphabet Sign Language Recognition Using Transfer Learning and Transformer Models,” pp. 1–6, Dec. 2024, doi: https://doi.org/10.1109/icca62237.2024.10927914.

[10] K. Meanhor and M. Thu, “Emergency Care System using Deep Learning for People with Disabilities in Cambodia,” 2025 International Conference on Software, Knowledge, Information Management & Applications (SKIMA), Paisley, United Kingdom, 2025, pp. 1–8, doi: 10.1109/SKIMA66621.2025.11155686.

[11] . H. Noor et al., “Real-Time Arabic Sign Language Recognition Using a Hybrid Deep Learning Model,” *Sensors*, vol. 24, no. 11, art. 3683, June 2024, doi:10.3390/s24113683.

[12] V. Adithya and R. Rajesh, “A video dataset of the hand gestures of Indian Sign Language words used in emergency situations,” *Data in Brief*, vol. 31, 106016, pp. 1–8, 2020. doi: 10.1016/j.dib.2020.106016

[13] M. Vázquez-Enríquez, J. L. Alba-Castro, L. Docío-Fernández, and E. Rodríguez-Banga, “SWL-LSE: A Dataset of Health-Related Signs in Spanish Sign Language with an ISLR Baseline Method,” *Technologies*, vol. 12, no. 10, art. 205, 2024. doi: 10.3390/technologies12100205

[14] K. Kim, H. Lee, and S. Park, “Korean Disaster Safety Information Sign Language Translation Benchmark Dataset,” in *Proc. LREC-COLING 2024*, pp. 10051–10061, Torino, Italy, 2024. Available: https://aclanthology.org/2024.lrec-main.869

[15] H. Luqman, “ArabSign: A Multi-modality Dataset and Benchmark for Continuous Arabic Sign Language Recognition,” in *Proc. IEEE Conf. Automatic Face and Gesture Recognition (FG)*, 2023. Available: https://hamzah-luqman.github.io/ArabSign

[16] “KArSL,” *Github.io*, 2021. https://hamzah-luqman.github.io/KArSL/

[17] T. Nguyen and T. Minh, “SignBart -- New approach with the skeleton sequence for Isolated Sign language Recognition,” *arXiv.org*, 2025. https://arxiv.org/abs/2506.21592 (accessed Dec. 14, 2025).

[18] T. Nguyen et al., “SignBart: Official Implementation,” GitHub repository, 2025. [Online]. Available: https://github.com/tinh2044/SignBart