Machine Learning Techniques for Sign Language Recognition

Víctor Osejo[1]; Mateo Ballagán[1]; Estefanía Oñate[1]; Jeffrey Guerrero[1]; Viviana Moya[1]; Andrea Pilco[1]; Juan Vásconez[2]

[1]Faculty of Technical Sciences, International University of Ecuador UIDE, Quito, Ecuador, viosejoal@uide.edu.ec, maballaganfu@uide.edu.ec, esonatemo@uide.edu.ec, jeguerrerope@uide.edu.ec, vimoyago@uide.edu.ec, anpilcoat@uide.edu.ec.

[2]Energy Transformation Centre, Faculty of Engineering, Universidad Andres Bello, Santiago, Chile, juan.vasconez@unab.cl.

Abstract*– In this paper, a sign language recognition system for the Ecuadorian Sign Language vowels (A, E, I, O, U) using Random Forest (RF) and YOLOv8 models is proposed. For this purpose, a new dataset with a total of 500 RGB images in natural light for single-hand gestures was created. RF model used the normalized hand landmark coordinates obtained by using Mediapipe while for real-time gesture detection, YOLOv8 took images with higher resolutions. Hypothesis testing results also showed that the RF model had better accuracy, precision, recall, and computational complexity with the accuracy, precision and Recall scores all 100 % and were preferred for real-time applications. YOLOv8 performance was high with a precision of 100% revealing the model as suitable for tasks related to images. Final real-time inference tests validated our claims of scalability and efficiency of RF as it was able to classify gestures within an average of 0.0055 seconds of inference time. This paper underscores the importance of machine learning models in enhancing inclusion as well as closing communication barriers for the hearing-impaired population.

Keywords—Sign Language, Random Forest, YOLOv8, hand recognition, image classification

I. Introduction

For impaired and deaf communities, visual gestural language systems facilitate the expression of concepts, ideas, feelings, and emotions through manual signs, finger movements, facial expressions, and body movements. Instead of relying on oral communication and sound patterns, those with hearing impairments rely on visual signs to interact and communicate. The communication gap experienced by people with speech impairments has driven research into Sign Language Recognition (SLR) systems. These efforts combine computer vision [1] , artificial intelligence, machine learning [2], natural language processing, and linguistics disciplines to develop methods and algorithms capable of identifying sign language gestures and accurately interpreting their meanings.

Sign Language (SL) has become relevant within societies worldwide. However, there is still a communication barrier between hearing people and deaf people, which limits inclusion and equitable access to education, employment, and information and interaction with other individuals within the community. Approximately 80% of individuals with hearing disabilities reside in low- and middle-income countries [3]. Hearing loss can significantly impede spoken language development in children, posing challenges to their communication and education. Historically, formal education for deaf children has sought to address these barriers by developing tools that support and enhance learning [4].

This research aims to empower users by enabling seamless recognition of vowels in sign language. By leveraging computer vision and artificial intelligence techniques, the proposed interactive system is designed to accurately interpret hand movements, offering a tool for vowel recognition. This system is a resource for skill development, particularly for children and beginners learning sign language.

No universal sign language is used by all hearing- and speech-impaired individuals [5]. Each country typically has a unique SL, reflecting the region's culture and linguistic nuances [6]. Research efforts focus on each country's distinct sign languages and dialects, aiming to address their specific characteristics. Moreover, with the rapid advancement of artificial intelligence, progress has been made in the detection and classification of SL. For instance, the study [7] utilized the ROBITA Indian Sign Language Gesture database along with a Convolutional Neural Network (CNN) model for sign-to-text conversion, achieving an accuracy of 70%. In [8] utilized the Bangla Sign Language (BdSL) dataset and the American Sign Language (ASL) dataset to evaluate the performance of CNN, Visual Geometry Group 16 (VGG16), and Residual Network (ResNet) models in recognizing sign language. The performance of these models was compared using metrics such as accuracy, precision, recall, and F1-score.

Sign language is categorized as either static or dynamic [9]. The study [10] utilized a dataset comprising 2,369 training images and 342 validation images, representing 49 distinct American Sign Language (ASL) signs for static SL. The study employed YOLOv5 and CNN to translate static signs into text. For dynamic sign language, [11] developed a recognition system using the YOLOv5 target detection algorithm combined with a Long Short-Term Memory (LSTM) network and OpenPose technology. This system relied on an open-source Chinese Sign Language dataset from CUHK. Similarly, study [12] focused on recognizing, interpreting, and translating dynamic sign language into text or speech, specifically for Indonesian Sign Language, using LSTM technique.

According to Spanish Sign Language, [13] developed a system for recognizing Mexican Sign Language. The system used a motion sensor to capture depth images, followed by classification with pattern recognition algorithms such as Random Forests, Decision Trees, and Artificial Neural Networks to identify the signs. Similarly, [14] focused on developing a system for Peruvian Sign Language, employing an artificial neural network. The research in [15] utilized a dataset containing static signs of the 30 letters composing the Spanish alphabet and tested two types of neural networks: Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs). The study concluded that CNNs achieved superior accuracy, with a maximum value of 96.42%.

The study [16] emphasized the need for intelligent solutions in sign language systems, noting that developing a perfect intelligent system for sign language recognition remains a significant challenge. This has motivated research into sign language recognition systems, exploring and testing multiple models to identify the most effective approaches, as demonstrated in [17]. Additionally, [18] introduced a dataset of vowel signs captured from a single individual and analyzed the performance of several deep learning models. Building on these findings, this research has focused on developing an Ecuadorian vowel sign recognition system designed to provide a solution for supporting impaired and deaf communities, particularly children and beginners, in learning sign language.

The key contributions of this research include:

1) Creation of an RGB image dataset for each Ecuadorian Sign Language vowel (a, e, i, o, u).
2) Evaluate and compare the performance of two implemented models: Random Forest and YOLOv8.
3) Implementation of a real-time system for interpreting the corresponding vowels in sign language.

II. Sign Language Recognition System

The overall methodology pipeline to train algorithm is illustrated in Fig. 1, which visually represents the steps and techniques employed throughout this section.

A. Dataset Acquisition

For the generation of the dataset, a Python code was created with images of the gestures of the five vowels of the Ecuadorian Sign Language (LSE) acquired and taken with only one hand (Right) in a natural environment and lighting conditions. Fig. 2 shows the code interface to take the images for the dataset.

The database contains the images corresponding to the gestures of the five vowels. It comprises 500 images representing the five vowels (A, E, I, O, U).

Table I shows the distribution of the images for each class. For the dataset collection, images were taken of a single person performing vowels in sign language using a conventional webcam. The dataset contains RGB images in JPG format, 640 x 480 in size, and a resolution of 96 dpi (dots per inch). A sample of the dataset is shown in Fig.3.

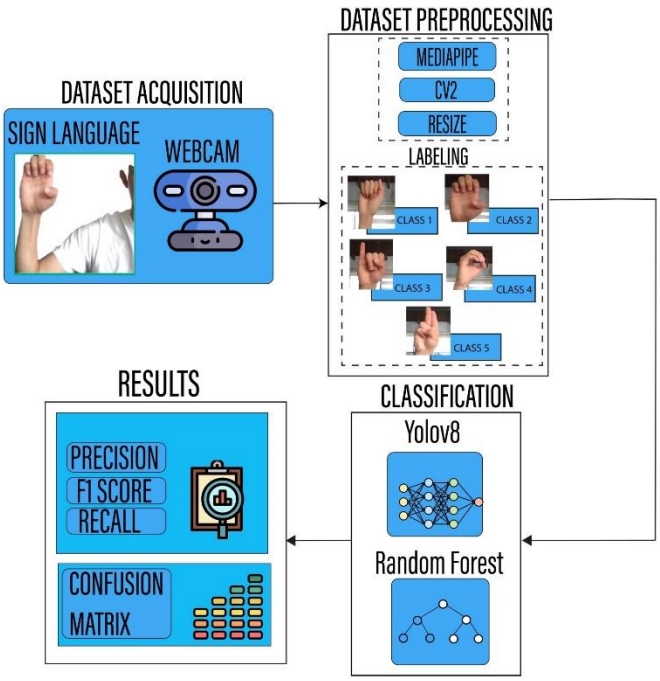

Fig. 1. The methodology pipeline to train algorithm.

Fig. 2. Interface of the code for image collection.

TABLE I
DATASET INFORMATION

Vowel	Number of Images
A	100
E	100
I	100
O	100
U	100
Total	500

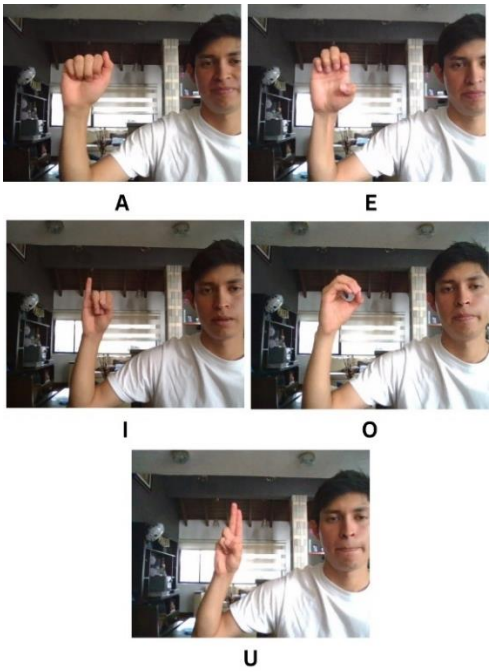

Fig. 3. Sample of the vowels for the dataset.

B. Dataset *Preprocessing*

The vowel detection system was made using the Mediapipe library, a tool that facilitates the accurate capture of the key points of the hands through its "Hand Landmarks" function. This library identifies 21 points on each hand, distributed between the palm and fingers, which are essential for interpreting gestures.

Of these 21 points, the first 6 correspond to the palm and wrist, while the other 15 are distributed along the phalanges of the fingers. The palm points (0 to 5) define the base of the hand, and the points from 6 to 21 correspond to the phalanges of the fingers (the thumb has 3 points, and the other fingers have 4). Each of these points is represented by normalized (x, y) coordinates within the range [0, 1], which makes the positions proportional to the image size, regardless of the camera resolution. Fig. 4 shows an example of the preprocessing where the "Landmarks" are drawn for each image.

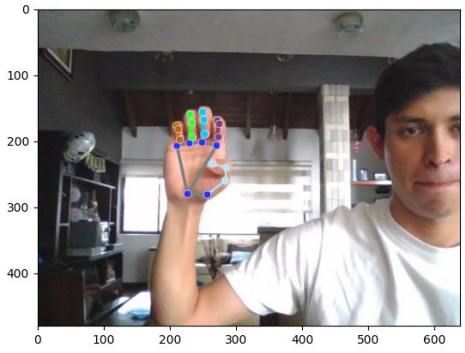

Fig. 4. Sample of the Landmarks.

To ensure consistency in the data obtained, these coordinates (x, y) are stored in an "Array", which will be used to train our model and are normalized about the minimum values of each axis in the image. In this way, the coordinates are adapted to facilitate further analysis. These key points provide a complete representation of the hand, capturing its static and dynamic position, which is essential to identify the finger configurations corresponding to the LSE vowels.

Once the data for each image has been processed and normalized, it is stored with the corresponding label in a list that constitutes the system's training set. All this information is stored in a file called "data.pickle" to ensure persistence and facilitate its use in future sessions. This file contains the processed data and associated labels, allowing the system to load and use it to train machine learning models. These models can accurately recognize LSE vowel gestures, facilitating real-time communication between deaf or hard-of-hearing people and their environment.

C. Classification

For image classification, we trained and tested two algorithms of deep learning and supervised learning models. To be more specific, the YOLOv8 and Random Forest models were chosen for the purpose of this research.

The images of the dataset were further divided into 80/20 split in order to train the "Random Forest" model. However, in the case of the "YOLOv8" model, the sets of images distributed 70%/20%/10% for training, validating and testing respectively.

The following subsections will further explain the basics of both models and how they were implemented in this work.

1) Random Forest:

For this work, Random Forest (RF) was chosen to accomplish sign language recognition due to its robustness and simplicity despite noisy data. This machine learning method constructs multiple decision trees using random feature subsets, improving generalization and reducing overfitting [19]. The RF model was trained on a normalized feature matrix extracted from processed sign language data, paired with class labels representing gestures.

Once it has been trained, RF infers predictions from all trees via majority voting, ensuring robust and accurate classification even under noisy or ambiguous inputs. Its inherent ability to handle complex datasets makes it suitable for diverse applications, including gesture recognition. The model's randomness in tree construction enhances prediction stability and reduces bias. RF's reliability and scalability make it a powerful tool for advancing machine learning in sign language recognition tasks. The selected training parameters, detailed in Table II, were chosen to balance performance and computational efficiency and get a high accuracy.

TABLE II
TRAINING HYPERPARAMETER FOR RANDOMFOREST

Hyperparameter	Description
Number of trees (`n_estimators`)	100 estimators used for constructing the ensemble model.
Test data proportion	20% of the total dataset, stratified to maintain the class distribution.
Evaluation metric	Accuracy calculated using the `accuracy_score` metric.

2) YOLOv8:

The other model consider for this paper was YOLOv8. It was chosen for its high-accuracy object detection capabilities, optimized to identify sign language vowels from real-time images. As the latest iteration of the YOLO models, YOLOv8 features advanced architectural improvements, including anchor-free detection, hierarchical feature extraction, and enhanced computational efficiency, making it well-suited for gesture recognition tasks [20], [21].

Input pictures are resized to 800 pixels (imgsz=800) to maintain consistent high-resolution input, ensuring the model captures fine details of hand gestures. YOLOv8 employs convolutional layers with Leaky ReLU activations, which are particularly effective at preserving small-scale features and ensuring relevant information is retained. Unlike traditional anchor-based methods, YOLOv8 uses an anchor-free mechanism to directly predict bounding box coordinates and class labels, significantly improving detection accuracy for small objects like hands while reducing computational complexity [21], [22].

The model was trained for 25 epochs using a custom annotated dataset downloaded from Roboflow, with hyperparameters detailed in Table III. Real-time performance metrics, including confusion matrices and validation curves, were monitored to evaluate accuracy and detect potential overfitting. This combination of robust architecture, optimized hyperparameters, and precise training ensures that YOLOv8 efficiently classifies vowels (A, E, I, O, U) from manual gestures in real-time, demonstrating high accuracy and generalization during testing.

TABLE III
TRAINING HYPERPARAMETER FOR YOLOV8

Hyperparameter	Description
Number of epochs	25 epochs for iterative model optimization.
Image size (`imgsz`)	800 pixels, ensuring consistent high-resolution inputs.
Base model	`yolov8s.pt` (pretrained small version).
Loss function	Weighted sum of coordinate loss, bounding box size loss, and class classification loss.
Optimizer	SGD with a learning rate of 0.001.
Metrics visualization	Enabled (`plots=True`) to analyze results during training.

III. EXPERIMENTAL ANALYSIS

This section evaluates the performance of the Random Forest (RF) and YOLOv8 models for recognizing Ecuadorian Sign Language vowels. The experiments were conducted using a Dell G15 5515 Ryzen Edition laptop, equipped with an AMD Ryzen 7 5800H processor, 8 GB RAM, 512 GB SSD, and a 4 GB NVIDIA GeForce RTX 3050 Ti GPU. The dataset was divided into training and testing subsets as 80%/20% for RF and 70%/20%/10% for YOLOv8, for training, validation, and testing, respectively.

A. Model Accuracy comparison

Table IV summarizes the evaluation metrics for both models on the test dataset. The RF model was able to achieve a classification performance of 100% precision, recall, and F1-score, making it highly reliable for this task. The YOLOv8 model also delivered high performance results, achieving 99.7% precision. However, its slightly lower performance compared to RF reflects the complexity of image-based deep learning approaches in scenarios with limited training data.

TABLE IV
TESTING PROCEDURE FOR EACH CLASS

Model	Precision%	Recall %	F1-score %
Random forest	100	100	100
YOLOv8	99.7	100	100

B. Real-Time Gesture Inference

Real-time inference tests were conducted using the RF model to classify each vowel gesture. Table V displays the results, including the number of iterations, accuracy, average inference time per gesture, and total test time. RF consistently achieved 100% accuracy across all tests, with an average inference time of approximately 0.0055 seconds per gesture. These results demonstrate its efficiency and reliability for real-time applications.

TABLE V
METRICS RESUME OF REAL-TIME INFERENCE

Test	Vocal	Iterations	Accuracy (%)	Average Time (s)	Test Time (s)
1	A	185	100	0.005403	17.19983
2	A	175	100	0.005478	16.26526
3	E	163	100	0.005602	16.46459
4	E	168	100	0.005506	15.65461
5	I	142	100	0.005483	13.88560
6	I	158	100	0.005670	15.44933
7	O	187	100	0.005447	17.14277
8	O	177	100	0.005562	16.50498
9	U	162	100	0.005704	15.83840
10	U	169	100	0.005515	16.00188

Finally, the gestures are inferred in real time and labelled with the corresponding vowel. Fig. 5 shows an example of the inference in real time, in this case, the vowel A.

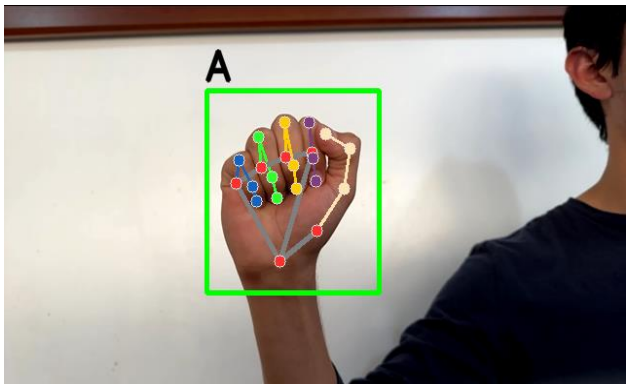

Fig. 5. Example of the inference in real time

C. *Test Results*

Once the model has been evaluated using the test portion of the dataset, it was confirmed that the model can effectively identify the five vowels, as shown in Fig. 6. Then, a confusion matrix was generated as shown in Fig. 7, emphasizing that this matrix is the same in both models. The testing dataset was generated with 20 pictures per class, corresponding to each vowel to be detected. Runing the trained model with all 100 pictures and comparing them with the real labels corresponding to each one, a 100% accuracy was obtained.

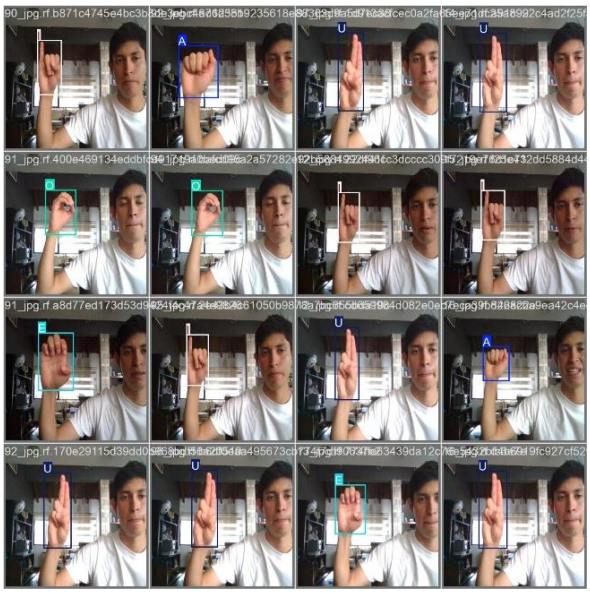

Fig. 6. Testing Results.

These results may be due to the nature of the pictures, where all the data was collected under the same lighting conditions, in the same scenario and with the same test subject. Although the dataset was divided and the testing section wasn't used during the training process, the model doesn't fail in detecting the vowel.

The confusion matrix shows a perfect detection of each class, without false positives nor false negatives. Further testing could include different subjects, environments and lighting conditions, but sue to limitations of time and resources it wasn't possible in this iteration of the model. Furthermore, the model is currently used in the environment tested, so the performance isn't compromise.

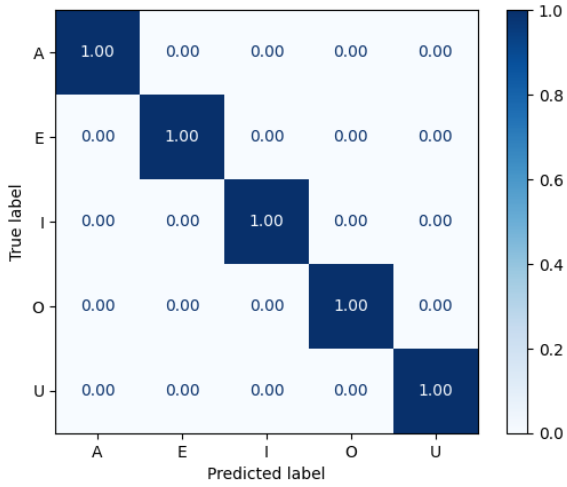

Fig. 7. First confusion matrix.

IV. Conclusions

This paper presents a machine learning system for recognizing the sign language vowels: A, E, I, O, U of Ecuadorian Sign Language using the Random Forest Model and the YOLOv8 Model. A new custom database was chosen with 500 images marked in natural environments to achieve adequate model training and testing.

The RF model showed high schematic ability and high prediction accuracy with scores of 100 at all the tested performance indicators. This strategy of using hand landmark coordinates is computationally not very demanding and a perfect fit for real-time application. However, for YOLOv8, which has image-based detection capabilities, the precision set at 99.7% and mAP of 99.5% were attained. Thus, YOLOv8 is quite efficient, but its training presupposes the use of large datasets and necessitates the application of means which require higher computational potency.

Future work will focus on expanding the dataset in EcuSign to cover the entire American Sign Language (ASL) alphabet, improving its effectiveness for deaf people. The goal will be to allow recognition of both vowels and consonants, increasing practical application. By advancing the creation and validation of these systems, the study promotes communication equity and strengthens inclusion, enabling diverse communities to be better educated, understood, and connected through accessible technologies.

Furthermore, future developments will focus on several areas require attention to enhance robustness. The dataset, while structured, is limited to a single subject and controlled conditions, which may result in overfitting or lower than expected accuracy on variant conditions. Expanding the dataset with diverse samples would improve model generalization and reliability in practical environments, making the system more adaptable. To further validate the data user-centered evaluations, including tests with hearing-impaired individuals, would be incorporated. Addressing these aspects would significantly strengthen the study's impact and help move toward a more complete and reliable solution for sign language recognition.

Finally, the comparison between RF and YOLOv8 models provides a strong base and will be enriched by exploring additional architectures such as CNNs, transformers, or hybrid methods.

REFERENCES

[1] A. Pilco, V. Moya, A. Quito, J. P. Vásconez, and M. Limaico, "Image Processing-Based System for Apple Sorting," *Journal of Image and Graphics*, vol. 12, no. 4, pp. 362–371, 2024, doi: 10.18178/joig.12.4.362-371

[2] J. P. Vásconez, I. N. Vásconez, V. Moya, M. J. Calderón-Díaz, M. Valenzuela, X. Besoain, M. Seeger, and F. Auat Cheein, "Deep learning-based classification of visual symptoms of bacterial wilt disease caused by Ralstonia solanacearum in tomato plants," *Computers and Electronics in Agriculture*, vol. 227, p. 109617, 2024, doi: 10.1016/j.compag.2024.109617.

[3] J. S. Izquierdo-Condoy, L. E. Sánchez Abadiano, W. Sánchez, I. Rodríguez, K. De La Cruz Matías, C. Paz, and E. Ortiz-Prado, "Exploring healthcare barriers and satisfaction levels among deaf individuals in Ecuador: A video-based survey approach," *Disability and Health Journal*, vol. 17, no. 3, p. 101622, 2024, doi: 10.1016/j.dhjo.2024.101622.

[4] A. Lederberg, B. Schick, and P. Spencer, "Language and Literacy Development of Deaf and Hard-of-Hearing Children: Successes and Challenges," *Developmental Psychology*, vol. 49, Jul. 2012, doi: 10.1037/a0029558.

[5] Y. Grover, R. Aggarwal, D. Sharma, and P. K. Gupta, "Sign language translation systems for hearing/speech impaired people: A review," in *2021 International Conference on Innovative Practices in Technology and Management (ICIPTM)*, Feb. 2021, pp. 10–14, doi: 10.1109/ICIPTM52218.2021.9388434.

[6] Y. Grover, R. Aggarwal, D. Sharma, and P. K. Gupta, "Sign language translation systems for hearing/speech impaired people: A review," in *2021 International Conference on Innovative Practices in Technology and Management (ICIPTM)*, Feb. 2021, pp. 10–14, doi: 10.1109/ICIPTM52218.2021.9388434.

[7] K. S. Sindhu, Mehnaaz, B. Nikitha, P. L. Varma, and C. Uddagiri, "Sign Language Recognition and Translation Systems for Enhanced Communication for the Hearing Impaired," in *2024 1st International Conference on Cognitive, Green and Ubiquitous Computing (IC-CGU)*, Bhubaneswar, India, 2024, pp. 1–6, doi: 10.1109/IC-CGU58078.2024.10530832.

[8] S. Shaba, S. Sharmin, M. J. Hossain, and M. F. Monir, "NeuralGesture Communication: Translating one Sign Language to Another Sign Language Using Deep Learning Model and gTTs," in *2024 IEEE Vehicular Technology Conference (VTC2024-Spring)*, Jun. 2024, pp. 1–5, doi: 10.1109/VTC2024-Spring62846.2024.10683444.

[9] A. Wadhawan and P. Kumar, "Sign language recognition systems: A decade systematic literature review," *Archives of Computational Methods in Engineering*, vol. 28, pp. 785–813, 2021, doi: 10.1007/s11831-020-09443-3.

[10] M. M. Mohamed, E. Elnamla, N. H. H. Khamis, and N. A. B. N. Hisham, "Sign Language Recognition System for Service-Oriented Environment," in *2024 IEEE International Conference on Advanced Telecommunication and Networking Technologies (ATNT)*, Johor Bahru, Malaysia, 2024, pp. 1–4, doi: 10.1109/ATNT61688.2024.10719250.

[11] X. He, Y. Lin, Z. Hu, X. Xu, R. Xu, and W. Xiang, "AI Chinese sign language recognition interactive system based on audio-visual integration," in *2023 IEEE International Conference on Electrical, Automation and Computer Engineering (ICEACE)*, Changchun, China, 2023, pp. 962–968, doi: 10.1109/ICEACE60673.2023.10442295.

[12] R. Yunita, E. B. Nababan, and M. S. Lydia, "Indonesian Dynamic Sign Language Recognition for Individuals with Sensory Disabilities using LSTM," in *2024 4th International Conference of Science and Information Technology in Smart Administration (ICSINTESA)*, Balikpapan, Indonesia, 2024, pp. 417–420, doi: 10.1109/ICSINTESA62455.2024.10748114.

[13] R. Galicia, O. Carranza, E. D. Jiménez, and G. E. Rivera, "Mexican sign language recognition using movement sensor," in *2015 IEEE 24th International Symposium on Industrial Electronics (ISIE)*, Buzios, Brazil, 2015, pp. 573–578, doi: 10.1109/ISIE.2015.7281531.

[14] J. E. Mejía Gamarra, M. A. Salazar Cubas, J. D. Sosa Silupú, and C. E. Córdova Chirinos, "Prototype for Peruvian Sign Language translation based on an artificial neural network approach," in *2020 IEEE XXVII International Conference on Electronics, Electrical Engineering and Computing (INTERCON)*, Lima, Peru, 2020, pp. 1–4, doi: 10.1109/INTERCON50315.2020.9220257.

[15] E. Martinez-Martin and F. Morillas-Espejo, "Deep Learning Techniques for Spanish Sign Language Interpretation," *Computational Intelligence and Neuroscience*, vol. 2021, p. 5532580, Jun. 2021, doi: 10.1155/2021/5532580. PMID: 34220998; PMCID: PMC8219431.

[16] I. A. Adeyanju, O. O. Bello, and M. A. Adegboye, "Machine learning methods for sign language recognition: A critical review and analysis," *Intelligent Systems with Applications*, vol. 12, p. 200056, 2021, doi: 10.1016/j.iswa.2021.200056.

[17] K. Amrutha and P. Prabu, "ML Based Sign Language Recognition System," in *2021 International Conference on Innovative Trends in Information Technology (ICITIIT)*, Kottayam, India, 2021, pp. 1–6, doi: 10.1109/ICITIIT51526.2021.9399594.

[18] V. Alvear, C. Domínguez, and G. Mata, "Evaluation of Different Models for Spanish Sign Language Recognition," in *2024 10th International Conference on Control, Decision and Information Technologies (CoDIT)*, Vallette, Malta, 2024, pp. 888–892, doi: 10.1109/CoDIT62066.2024.10708369.

[19] A. Parmar, R. Katariya, and V. Patel, "A Review on Random Forest: An Ensemble Classifier," in *International Conference on Intelligent Data Communication Technologies and Internet of Things (ICICI) 2018*, J. Hemanth, X. Fernando, P. Lafata, and Z. Baig, Eds. Cham: Springer International Publishing, 2019, pp. 758–763, doi: 10.1007/978-3-030-03146-6_86.

[20] M. Sohan, T. Sai Ram, and Ch. Venkata Rami Reddy, "A Review on YOLOv8 and Its Advancements," in *Data Intelligence and Cognitive Informatics*, I. J. Jacob, S. Piramuthu, and P. Falkowski-Gilski, Eds. Singapore: Springer Nature Singapore, 2024, pp. 529–545, doi: 10.1007/978-981-99-7962-2_39.

[21] M. Hussain, "YOLOv5, YOLOv8 and YOLOv10: The Go-To Detectors for Real-time Vision," *arXiv*, 2024. [Online]. Available: https://arxiv.org/abs/2407.02988.

[22] J. Terven, D.-M. Córdova-Esparza, and J.-A. Romero-González, "A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS," *Machine Learning and Knowledge Extraction*, vol. 5, no. 4, pp. 1680–1716, 2023, doi: 10.3390/make5040083.