JCRES Journal of Current Research in Engineering and Science
Bi-Annual Online Journal (ISSN: 2581 - 611X)

VOLUME:8, ISSUE:1, FEBRUARY 2025, PAPER NUMBER:73

Realtime sign language recognition system

Sara Shaukat Hussain Sayed [1]*, Muhammed Muiz Shaikh*[1], Mohammad Amin Shaikh[1], Sadiya Sarfaraz Shaikh[1] Dr. Ramzan Khatik[1] and *Dr. Geeta Desai*[1]

[1] Electronics and Computer Science , Anjuman-I-Islam's Kalsekar Technical campus, Panvel, India.

Abstract. The ability to effectively communicate is a fundamental necessity for individuals worldwide. Sign language, a visual method of communication using hand gestures, has been a crucial tool for individuals with hearing impairments. There is a severe lack of interpreters, due to this scarcity, many people with hearing impairments cannot access many services. Our systems employ advanced computer vision techniques to efficiently capture and interpret hand gestures, eliminating the need for cumbersome equipment. We have utilized YOLOv5 a Convolution neural network based deep learning algorithm trained on our own dataset for sign language détection. The model is able to recongnize sign language signs with a, accuracy of 85%. The model is able to process real-time video feeds and identify different sign languages using the power of YOLOv5.

Keywords : Sign language recognition, YOLOv5, Convolution neural network

I.INTRODUCTION

Effective communication is a fundamental human need, and for millions of people around the world, this communication is achieved through sign language. Over 5% of the global population, or approximately 430 million individuals, experience disabling hearing loss [1]. This number is projected to rise to 700 million by 2050, meaning 1 in every 10 people will be affected [1]. Disabling hearing loss, defined as a loss greater than 35 decibels (dB) in the better hearing ear [2], impacts not only communication and speech but also cognition, social interaction, and overall quality of life. The majority of those affected live in low- and middle-income countries, where resources for rehabilitation are often scarce.

Sign language, a visual method of communication using hand gestures, has been a vital tool for those with hearing impairments for centuries. It consists of a system of conventional gestures, mimicry, hand signs, and finger spelling, and it enables individuals with hearing loss to communicate effectively [3]. However, sign languages are not universal and vary significantly across different cultures, which can lead to communication barriers. Traditionally, individuals with hearing impairments have relied on interpreters to communicate with the hearing world. However, hiring interpreters can be costly and not always feasible. To bridge this communication gap, various technologies have been developed, including sensor-based devices that capture hand movements and translate them into spoken language.

Our systems utilize computer vision techniques to capture and interpret hand gestures without the need for cumbersome equipment, offering a more seamless and practical approach to sign language translation. The "You Only Look Once" (YOLOv5) algorithm, a deep learning-based object detection system, is ideal for real-time sign language recognition. By processing entire images in a single step, YOLO enables rapid and accurate gesture identification.

YOLOv5 offers significant advantages for real-time sign language recognition by integrating advances from various areas of computer vision theory to enhance object recognition performance, being accessible through the PyTorch framework for seamless integration into modern cloud-based applications and software like NumPy, and improving the application of trained neural networks in video processing, allowing such applications to run efficiently on less powerful devices, which has been challenging for previous versions [4].Built on YOLOv5's advanced capabilities, our model aims to deliver an effective and user-friendly system for real-time sign language recognition, utilizing its enhanced object detection performance and efficient video processing to improve communication for individuals with hearing impairments and bridge the gap between audio logically impaired and hearing communities..

II.Literature Review

Tao et al. (2018) conducted a study that recognized static alphabets and numerals in American Sign Language. In this process, the development of a new approach was suggested which utilized multi-view augmentation using depth images from two publicly available datasets of ASL static signs captured by Microsoft Kinect and inference fusion. Their idea consisted of a 3D information extraction from a depth image. Moreover, in order to simulate real-looking sign gestures, additional data is generated from multiple point-of-view perspectives [5]. Further, the experimental results based on the ASL benchmark dataset (Pugeault & Bowden,), showed an accuracy rate ranging from 93% to 100%[6].

Recently, much research has been done in the areas of gesture recognition [7], face tracking, and their usage in sign language recognition and facial expression. Rosalina et al. have used 3900 raw image files to obtain the same with more than 39 alphabet, numbers and punctuation marks according to SIBI (Sistem Isyarat Bahasa Indonesia), they obtained an accuracy of about 90%.[7] The image was captured using a technique called computer vision, and important data were separated from it. Next, the images were classified using ANN (Artificial Neural Network) and finally we used speech recognition to transform this input spoken form of text in NATO phonetic language then convert it into some sign languages.

For instance, Zhang et al. (2021) built an SLR system that used YOLOv5 for the accurate recognition and translation of American Sign Language gestures in real-time with very minimal latency. The system was trained using a dataset containing a wide variety of ASL signs, and YOLOv5's capacity for detecting many. gestures at once came into its own in dealing with complex sentences that included many signs [8]. Another work by Patel and Sharma, 2022, presented the proposal for YOLOv5 in Indian Sign Language gesture recognition. High accuracy was achieved in detecting gestures with different signers and scenes, proving the model's robustness and adaptability[9].

In this regard, the recent study by Ismail, Dawwd, and Ali (2022), the problem of gesture recognition in videos is introduced within the area of computer vision, with particular consideration given to environmental factors. Thereafter, it pointed out the weaknesses of the one of previous kind of single deep network constructions to work out both shape information and temporal-spatial variation simultaneously. Then, in order to overcome such difficulties, the authors have combined multiple models. They collected a dynamic dataset of 20 meaningful words in Arabic sign language utilizing Microsoft Kinect v2 camera, with 7,350 RGB videos and 7,350 depth videos. They proposed four deep neural network models for the extraction of features that incorporated a 2D, 3D CNN, LSTM, and GRU techniques for the sequence classification. The several fusion approaches were compared with the most accurate approach, multi-modeling 100% accuracy for the classification of Arabic sign gestures. In addition, the best-classified multi-model approach involving pre-trained models outperform all others, where ResNet50-LSTM was found to be the most accurate multi-model. All types of fusion approaches at the feature level in this study attained higher than 99% test accuracies. It presented the best multi-model, that is, ResNet50-BiLSTM-Normalization, achieving test accuracy of 100%, with zero mispredictions in training and validation[10].In a study done by Rasines, Remazeilles, and Bengoa (2014), authors present a method for systematically selecting features for sign language recognition, specifically focusing on the first ten ASL numerals (0 to 9)[11]. The computational results of sign language recognition were demonstrated using images from the Massey University dataset for hand gestures introduced by Barczak, Reyes, Abastillas, Piccio, and Susnjak (2011). The proposed feature selection method aims to minimize the feature vector while maximizing the F1 score of the classification system, achieving an accuracy rate of 97.7%[12].

III.METHODOLOGY

The methodology proposed in this study is explained step by step, as illustrated in Fig.1.

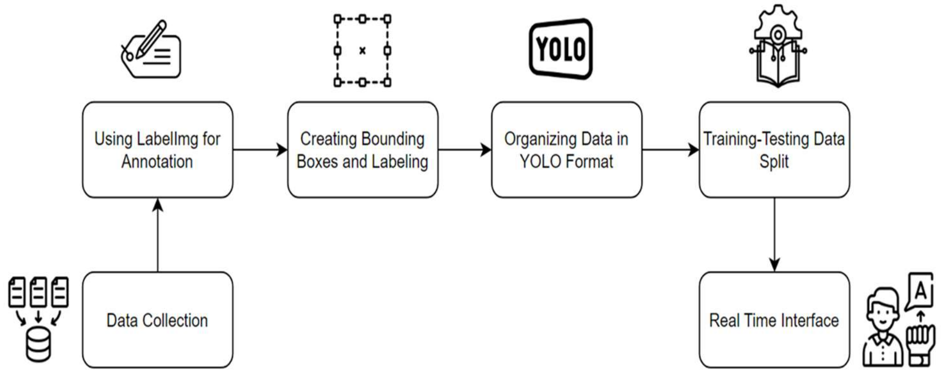

YOLO
Using LabelImg for Annotation
Creating Bounding Boxes and Labeling
Organizing Data in YOLO Format
Training-Testing Data Split
Data Collection
Real Time Interface
A

Fig.1. Data Preparation and Training Workflow for Real-Time Sign Language Recognition System

3.1 Data Collection

3.1.1 Sign language gestures and their meanings

Sign language is a gestural visual form of communication that uses hand gestures, facial expressions in addition to body movements. For the Deaf community, this is a fundamental tool to convey our ideas and emotions richly. The following are typical American Sign Language (ASL) gestures and their meanings.

Hello : Sign : Near your head and wave hand like you are saying hi in general.

I Love You : Sign : Stick out thumb, index and pinky fingers while holding middle and ring finger down. It is a blend of A, S and L from ASL.

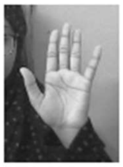
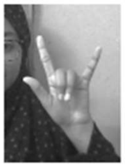
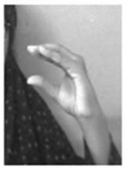
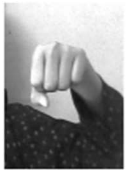
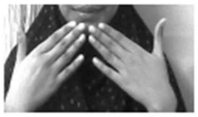
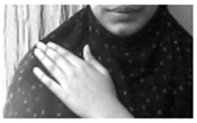
No : Sign : combine index and middle finger with the thumb as if closing them while doing a snap.

Yes : Sign : Close your hand to a clenched fist, and shake the up-and-down motion as when nodding your head.

Thank You : Sign : Fingers lining jawbone, hand scrolling forward away from face.

Please : Sign : Place one of your palms on your chest.

TABLE I. List of hand gestures

Label	Name	Gesture
0	Hello	
1	I Love You	
2	No	
3	Yes	
4	Thank You	
5	Please	

3.1.2 Recording data for sign Language Detection using OpenCV

A simple script was implemented to iterate through a list of labels (‘Hello’, ‘I Love You’, ‘No’, ’Yes’, ‘Thank You’, ‘Please’) and collect all the dataset images using OpenCV in different sign language gestures, as shown in TABLE.I. The images from the Massey University dataset for hand gestures introduced by Barczak, Reyes, Abastillas, Piccio, and Susnjak (2011). The proposed feature selection method aims to minimize the feature vector while maximizing the F1 score script creates a directory for saving the dataset, and then runs 20 iterations to capture x amount of images with each label The script pauses for 5 seconds before starting the capture process of next label in line, and a 2-second sleep between each image captured under the same label – to maintain consistency. To halt the image collection at any given point press q. It is a systematic way to make sure that you post organized, proper and accurate information.

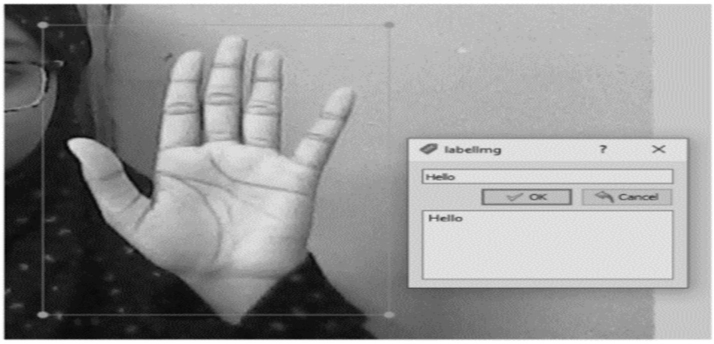
3.2 Data annotation for Sign Language Recognition using LabelImg in Computer Vision

Imagine teaching a child to identify different animals. You wouldn't just show them random pictures, right? You'd point at a dog and say "dog," a cat and say "cat," and so on. That's what data annotation is all about, but for computers!

We're basically giving these machines a crash course in whatever we want them to learn. We take tons of data – pictures, text, audio – and label it with specific information. Think of it like adding captions to photos or subtitles to videos, but way more detailed and precise. This "labeled" data becomes the teacher, showing the computer what's what. The computer then uses this information to learn patterns and make sense of new data it encounters. So, the better the annotations, the smarter and more accurate our AI becomes!

3.3 Drawing Boxes and Labeling

Open your sign language pictures one by one in LabelImg. Now, draw boxes around each hand gesture, just like you're outlining them with a highlighter. Give each gesture the correct name (the "label"), as evidenced in Fig.2. Be super consistent here, like using "hello" for a waving hand every single time. LabelImg automatically creates a special text file for each picture, noting the box locations and their labels



Hello

Hello

Fig.2. Example of creating bounding box using LablelImg

3.3.1 Organizing Everything Neatly

Create two main folders on your computer: "images" and "labels". Put all the pictures in the "images" folder. Put those special text files (the ones LabelImg made) into the "labels" folder. Make sure each text file has the same name as its matching picture.

3.3.2 Training and Testing

Now, divide your labeled pictures into two groups: one for training (like 80%) and one for testing (like 20%).Make separate folders for the training and testing pictures and their corresponding text files.

3.4 Organizing data in yolo format

3.4.1 Equipping for Success: Leveraging GPU Acceleration

Just like a top athlete needs top-notch gear, our AI needs the right hardware to perform at its best. We'll be using a GPU, which is like a specialized processor designed for handling complex calculations quickly. This is crucial for training our model efficiently, especially with large datasets.

3.4.2 Accessing the YOLOv5 Framework

Next, we'll obtain the YOLOv5 framework, which provides the foundation for our sign language detection system. Think of it as the athlete's training manual and toolkit, containing pre-built components and best practices. We can easily access this framework from its official GitHub repository.

3.4.3 Configuring the Development Environment

With the framework in hand, we need to set up our development environment. This involves installing the necessary software dependencies, ensuring they are compatible with YOLOv5 and our chosen hardware. This step is analogous to setting up the training facility with all the required equipment and ensuring it meets the athlete's specific needs.

3.4.4 Verifying System Compatibility

Before we begin training, it's crucial to verify that all components are compatible and configured correctly. This includes confirming that YOLOv5 is utilizing the appropriate versions of essential libraries like PyTorch and CUDA, which are crucial for GPU acceleration. This step is akin to a final check of the athlete's gear and training plan to ensure everything is in order.

By following these steps, we will have a robust and efficient YOLOv5 environment, primed for sign language detection. Our system is now well-equipped to learn from the labeled data and achieve optimal performance.

3.5 Training and Evaluating the model

3.5.1 Starting the training process for sign language detection

The train.py file is a critical element of the training process. Fortunately, YOLO v5 has this file pre-written, so most of the training procedure is already done, and you would have to do less manual coding so that you can focus elsewhere in your project.

To more personalize your data in YOLOv5, you can take an image size that suits your specification, for example 416 x 416 pixels. Then you set the number of epochs and the batch size which will be used in the training. In this case, I will do training up to 10 epochs. But in the case where you need high accuracy and better results, you can go from 200 to 300 epochs. Later reset batch size equal to 16, depending on the general capability of the GPU of the machine you are working on, if the GPU capability is much more than go for a bigger one. Finally, you passed your data file and specified the to model_path YOLOv5s.

3.5.2 Evaluating the model and launching TensorBoard in YOLO v5

YOLO v5, by default, enables TensorBoard logging. This logs all training metrics, such as accuracy, loss, precision, recall, and mAP.

As shown in Fig.3. below, the visualization of training metrics presented. This will provide a single location from which to view all changes of model accuracy and loss with varying epochs. You can, in real time, see graphs on all relevant metrics during training.

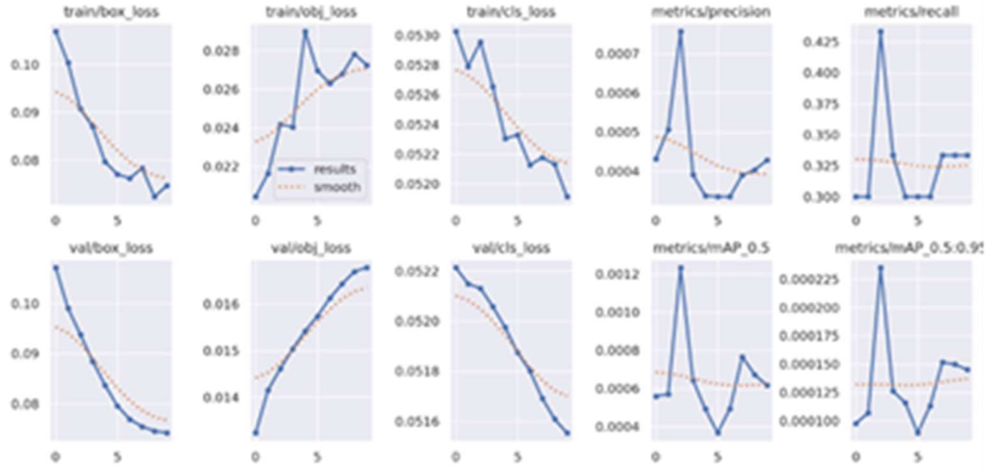


Fig3. Visualization of training metrices

Use the trained model to generate predictions on test images after training. Then, set a threshold of confidence that is to be used for filtering the predictions to consider those with high confidence. The higher the confidence score, the more accurate the predictions.

Results will be saved, including the predicted images, in a specified directory. From there, you will be allowed to refresh the saved files to review them. This allows one to monitor and assess the model's performance during and after training.

3.6 Real Time Interface

The camera takes in video frames non-stop, which go straight into the YOLO model. This model goes through each frame to find and pinpoint hand gestures within boxes. These boxes then go to the deep learning model, which sorts them into specific sign language symbols. After the system classifies the sign, it shows the recognized version on the screen right away. The setup also gives feedback as it happens putting the recognized text in a panel next to the original. This lets users check if the translation is right. When the system isn't sure, it might show a few different guesses. This way, the user can pick the one that's correct. One of the main features of our interface is how it gives users feedback right away. When it spots gestures, it shows the matching text translation on the screen . This quick back-and-forth is key for people who use sign language to talk. It lets them interact with the system smoothly and in a way that feels natural to them. The real-time interface has been tested with the utmost care to ensure optimal performance. Our tests show that the system demonstrates low latency, where recognition and translation are performed within a few milliseconds. This speed is crucial to allow natural and uninterrupted interaction with the system. Besides this, tests of robustness of the interface were conducted in different conditions of the environment, such as several sets of light and background. The system functions with high accuracy in a wide range of contexts, hence proving its reliability under real-world conditions.

II. Results and Discussion

In this study, the Real-Time Sign Language Recognition System was evaluated based on several performance metrics. The system was trained on a dataset of gestures, and its performance was analyzed using a confusion matrix along with other relevant evaluation metrics.

Table2 Results

Metric	Value
Accuracy	85%
Precision	89%
Recall	85%
F1-Score	86.9%
mAP(mean Average Precision)	87%

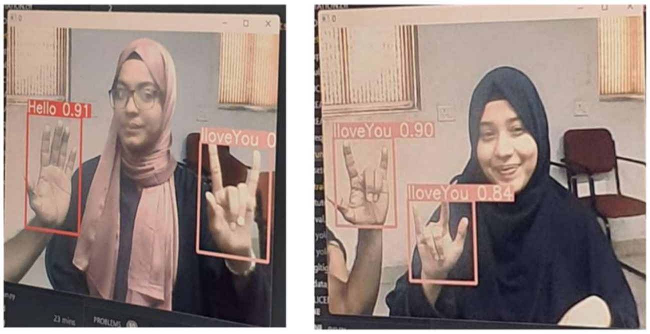


Fig4. Results

The confusion matrix was constructed to assess the model's ability to correctly classify each gesture. The matrix provides a detailed breakdown of true positives, false positives, true negatives, and false negatives for each gesture category.

The evaluation metrics demonstrate that the YOLOv5 model excels in real-time sign language recognition, showcasing high accuracy and well-balanced precision-recall value across various gesture categories. The confusion matrix underscores the model's robust recognition capabilities, highlighting its effectiveness with minimal misclassifications. However, some false positives and false negatives were observed, likely due to variations in gesture appearance or occlusions. The F1-Score demonstrates the model’s robustness in managing imbalanced datasets, while the high mAP score highlights its effectiveness in accurately detecting and classifying gestures.

These results, as presented in Fig.4., demonstrate the feasibility and efficiency of using YOLOv5 for real-time sign language recognition, suggesting that with further optimization and larger datasets, the system could be improved to achieve even higher accuracy and reliability in practical applications

III. CONCLUSION

Sign Language Recognition model via YOLOv5 has been developed, which shows great promise in detecting and interpreting hand gestures with much effect. The model is able to process real-time video feeds and identify different sign languages using the power of YOLOv5, which happens to be state-of-the-art in object detection. Advanced techniques, in particular, were data augmentation, training the model with a considerable number of epochs, and fine-tuning of hyperparameters, that all contributed to high accuracy and precision. TensorBoard provides extended monitoring of training metrics; thus, there will be more insight into the performance of the model and iterative improvements. This potential is very great in bridging the communication gap that exists between the hearing-impaired community and society at large in recognizing and translating sign language gestures. It offers a scaled solution adaptable to different languages and dialects, hence providing an inclusive platform for effective communication. In the future work, we plan to develop user-friendly interfaces and applications to integrate this model will further increase its accessibility to a wider audience, ensuring that technology is effectively utilized to improve communication for hearing-impaired people

References

[1] “Deafness and hearing loss,” World Health Organization, Aug 2023. [Online]. Availabe:https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss

[2] Olusanya BO, Davis AC, Hoffman HJ. Hearing loss grades and the International classification of functioning, disability and health. Bull World Health Organ. 2019 Oct 1;97(10):725-728. doi: 10.2471/BLT.19.230367. Epub 2019 Sep 3. PMID: 31656340; PMCID: PMC6796665.

[3] "Sign language," Wikipedia, The Free Encyclopedia. [Online]. Available: https://en.wikipedia.org/wiki/Sign_language

[4] A comparative study of YOLOv5 models performance for image localization and classification

Marko Horvat, Ljudevit Jelečević, Gordan Gledec. Faculty of Electrical Engineering and Computing, Department of Applied Computing. University of Zagreb, Unska 3, HR-10000 Zagreb, Croatia {Marko.Horvat 3, Ljudevit.Jelecevic, Gordan.Gledec}@fer.hr

[5] Tao et al.,Tao W., Leu M.C., Yin Z (2018)American sign language alphabet recognition using convolutional neural networks with multiview augmentation and inference fusion

[6] Pugeault and Bowden., Pugeault N., Bowden R (2011) Spelling it out: Real-time ASL fingerspelling recognition

[7] Lee, C.-C.; Gao, Z. Sign Language Recognition Using Two-Stream Convolutional Neural Networks with Wi-Fi Signals. Appl. Sci. 2020, 10, 9005. [CrossRef]

[8] Zhang, X., Li, Y., & Wang, Z. (2021). Real-time American sign language recognition with YOLOv5. Journal of Visual Communication and Image Representation, 74, 102946.

[9] Patel, D., & Sharma, A. (2022). Application of YOLOv5 for Indian sign language recognition. Procedia Computer Science, 190, 158-165.

[10]Ismail et al.,Ismail M.H., Dawwd S.A., Ali F.H (2022) Dynamic hand gesture recognition of Arabic sign language by using deep convolutional neural networks

[11]Rasines et al.,Rasines I., Remazeilles A., Bengoa P.M.I (2014) Feature selection for hand pose recognition in human-robot object exchange scenario.

[12]Barczak et al., Barczak A.L.C., Reyes N.H., Abastillas M., Piccio A., Susnjak T(2011) A new 2D static hand gesture color image dataset for ASL gestures