Using YOLOv5 Algorithm to Detect and Recognize American Sign Language

Akhil Sharma
Department of Information Technology
Sardar Patel Institute of Technology
Mumbai, India
akhil.sharma@spit.ac.in

Shantanu Kadam
Department of Information Technology
Sardar Patel Institute of Technology
Mumbai, India
Shantanu.kadam@spit.ac.in

Dr. Surekha Dholay
Department of Computer Engineering
Sardar Patel Institute of Technology
Mumbai, India
surekha_dholay@spit.ac.in

***Abstract*—Sign language is a visual communication system utilized by individuals who face challenges with speech or hearing. Understanding and interpreting these gestures is crucial for inclusivity in verbal communication. However, people unfamiliar with sign language often struggle to comprehend its meaning. This research paper presents a solution for detecting and recognizing sign language gestures for alphabets and numbers. While previous studies have explored deep learning methods for sign language recognition, their effectiveness remains limited. To address this, we propose a lightweight, fast, and accurate solution based on YOLOv5, a state-of-the-art object detection model. By leveraging YOLOv5, we aim to enhance the recognition and understanding of sign language, enabling greater communication accessibility for individuals with speech or hearing difficulties.**

***Index Terms*— YOLOv5, Object Recognition, Computer Vision**

I. INTRODUCTION

Verbal language serves as the primary means of communication for most individuals, but those with hearing and speech difficulties often rely on hand and facial gestures to express their thoughts. However, there exists a significant communication barrier between sign language users and the verbal language community, as the latter is unfamiliar with interpreting sign language gestures. Sign languages possess their own unique grammar, syntax, and variations based on geography and context, such as American Sign Language, Bangla Sign Language, and Indian Sign Language. Additionally, the same gesture can have different interpretations within a single sign language variant, sometimes representing an entire word and other times denoting a single alphabet or number. Furthermore, combinations of hand, mouth, and facial expressions are employed to convey messages effectively. Sign language can be categorized into three variants: non-manual features encompassing tongue, facial expression, body poses, and hand gestures; word-level sign spelling where each gesture signifies a complete word; and finger vocabulary where individual gestures represent specific alphabets or numbers. Bridging the communication gap between sign language users and the verbal language community requires a deeper understanding of these linguistic nuances and variations.

In this paper,The dataset used in our project is our personalized data set. The given dataset has been compiled from various persons.

The dataset contains 1200 images of size 416 x 416 pixels. The dataset has 6 classes of images namely Hello(200 images), Yes(200 images), No(200 images), ILoveYou(200 images), Thanks(200 images) and Please(200 images).to recognize finger-spelled vocabulary.

Fig. 1. Sign Language gesture

In finger spelled vocabulary, there are a total of 6 unique gestures, comprising of “Hello”, “Please”, “Yes”, “No”, “Thank you”, “I Love You”. Existing literature presents two primary approaches for gesture recognition. The first involves employing specialized devices to capture and identify the gestures, while the second utilizes deep learning techniques to analyze hand movements. However, deep learning models can be computationally expensive, and specialized gesture-capturing devices are often costly and not readily accessible to the general population.

A promising solution for gesture recognition is YOLO (You Only Look Once), a Convolutional Neural Network (CNN)-based algorithm known for its fast and efficient object detection capabilities. YOLO has been successfully applied to real-time tasks like pedestrian detection, traffic sign recognition, and mask detection. Building upon this, the proposed solution employs YOLOv5, a lightweight and pretrained model specifically designed for gesture recognition. This model offers high accuracy and frames per second (fps) performance, making it suitable for real-world scenarios.The advantage of the proposed approach is that it does not require any specialized devices.

II. Literature Review

The paper ,"Sign Language Interpreter System: An alternative system for machine learning" by Salma A. Essam El-Din; Mohamed A. Abd El-Ghany et al. (2020):
Paper presents a novel deep learning architecture for real-time sign language recognition using a low-cost wearable glove that captures hand gestures. The architecture includes a temporal convolutional network and an attention-based encoder-decoder network that can recognize sign gestures performed by different users with a high accuracy of 97.57% and 95.44% for American Sign Language (ASL) and gesture recognition tasks, respectively. The proposed method has the potential to aid in improving communication between the deaf and hearing communities by recognizing different sign languages and achieving real-time performance with a latency of 30 ms.
Topics discussed:
Introduction to sign language and the challenges faced by the deaf community in communicating with the hearing community.
End-to-end deep learning architecture for real-time SLR
Overview of existing sign language recognition (SLR) systems and their limitations

The paper,"Indian Sign Language Character Recognition " by Dr.Sapna B Kulkarni et al. (2020):
The paper describes a real-time sign language detection system that uses the Tensorflow object detection model zoo. The system uses a single camera for video acquisition and processing and can detect and recognize various sign language gestures in real-time with high accuracy. The proposed system can be used as a communication tool for hearing-impaired individuals and integrated with various applications to improve their quality of life. The system achieves an accuracy of 89.68% on a dataset of sign language gestures and can be further improved by using more advanced object detection models and collecting more diverse and larger datasets.
Topics discussed:
Sign language recognition system for Indian Sign Language (ISL).
The system achieves an accuracy of 98.45% on the test dataset.
The system uses a deep convolutional neural network (CNN) and transfer learning.
custom dataset of 26 different ISL characters.

The paper "Real-Time Sign Language Detection using TensorFlow, OpenCV and Python" by Prashant Verma, Khushboo Badli et al. (2013):

The system utilizes a convolutional neural network (CNN) model to detect hand gestures and translate them into corresponding text.They collected a dataset of hand gesture images using a webcam and preprocessed the data by resizing, normalization, and augmentation.

Topics discussed:

System designed usingTenserFlow,OpenCV, and python.

Trained Cnn model with Tensorflow with 97.8% accuracy.,OpenCV for real-time video capture and processing.

Model classify hand gestures and translates into text using lookup table,Achieved Frame rate of 20fps.,detected 26 American Sign language(ASL)

The paper,"Sign Language Detection using Tensorflow Object Detection Model Zoo et al. (2021):
The paper describes a real-time sign language detection system that uses the Tensorflow object detection model zoo. The system uses a single camera for video acquisition and processing and can detect and recognize various sign language gestures in real-time with high accuracy. The proposed system can be used as a communication tool for hearing-impaired individuals and integrated with various applications to improve their quality of life. The system achieves an accuracy of 89.68% on a dataset of sign language gestures and can be further improved by using more advanced object detection models and collecting more diverse and larger datasets
Topics discussed:
Real-time sign language detection system using the Tensorflow object detection model zoo.
The proposed system achieves an accuracy of 89.68%.
Proposed system is a cost effective solution that uses a single camera for video acquisition and processing.
Detected 26 American Sign language(ASL)

III. Methodology

A. Data Collection and Pre-Processing

1) Dataset: We have created a unique dataset consisting of 1200 close-up, color images. These images feature hand postures captured from four different individuals, under various lighting conditions. To enhance the quality of the dataset, we have employed various image processing techniques. The hand postures are displayed against a diverse range of backgrounds. Our dataset comprises a total of six distinct gestures, as illustrated in Figure 1.

2) Data Pre-Processing:

a) Data Labeling: The dataset consists of gesture images that require labels and annotated bounding boxes to be used for training in YOLOv5. It is necessary for the bounding box coordinates to be normalized between 0 and 1. To facilitate this process, we utilize an online platform called Roboflow (www.roboflow.com). This website simplifies the task of annotating and formatting the data labels according to our requirements.

b) Data Augmentation: Since the size of our training dataset is relatively small, we employ a widely recognized technique called data augmentation to enhance model generalization and prevent overfitting. This technique involves applying various transformations to the training images. To facilitate this process, we utilize the data augmentation capabilities provided by the Roboflow website. This platform not only enables us to augment the training data but also takes care of updating the corresponding labels for the augmented images. The table below outlines the specific augmentation techniques used, along with the corresponding augmentation values. Each original image is augmented to produce three additional versions, effectively increasing the total number of training images by three times.

c) Dataset Split: We divided the dataset into three parts: 70% for training, 10% for validation, and 20% for testing. The original dataset consisted of 1200 images. After splitting the dataset according to the 70-20-10 ratio, the validation set and test set contained 120 and 240 images respectively, while the training set had 840 images. In order to augment the test set, we applied augmentation techniques, resulting in three additional versions for each image. Consequently, the training set expanded to a total of 3600 images, including the augmented versions.

B. Deep Learning Architecture

[28] For this experiment, we employed YOLOv5 as the deep learning architecture. YOLOv5 is known for its lightweight and fast nature, requiring less computational power compared to other state-of-the-art models, while maintaining accuracy levels comparable to them. It outperforms previous versions of YOLO in terms of speed. YOLOv5 utilizes CSPNET as its backbone for extracting feature maps from the input image. Additionally, it incorporates Path Aggregation Network (PANet) to enhance information flow. The architecture of YOLOv5 is depicted in Figure [30]. We chose to use YOLOv5 for this experiment based on the following factors:

1) Has useful components such as state-of-the-art activation function, hyperparameter, data augmentation technique and a convenient manual.

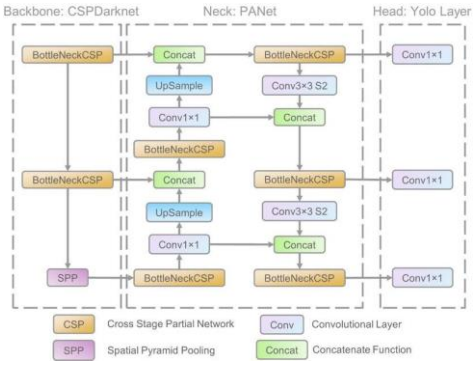

Fig. 2. YOLOv5 network architecture

2) Its lightweight architecture makes it computationally easy to train with small resources.
3) The size of the model is quite small and lightweight, thus can be used with mobile devices.

C. Model Training Process

In this approach, we utilized transfer learning by fine-tuning the YOLOv5 model, which was originally pretrained on the COCO dataset, for our relatively smaller dataset. To augment the input images, we applied various augmentation techniques such as HSV, color spacing, mosaic, and image scaling. The hyperparameters that were fine-tuned on the COCO dataset, including the SGD optimizer, a learning rate of 0.01, a weight decay of 0.0005, and a batch size of 16, were also used in our training process. We trained the model for 300 epochs and observed that it achieved stable and satisfactory accuracy after 183 epochs, with negligible improvements thereafter.

During the experimental evaluation, a confidence threshold of 0.4 was set to determine the detection results. The training process was carried out using Python 3.8 with PyTorch 1.8.1, leveraging a 12GB NVIDIA Tesla K80 GPU and the Colab Notebook environment. The training process took approximately 1 hour to complete all 300 epochs.

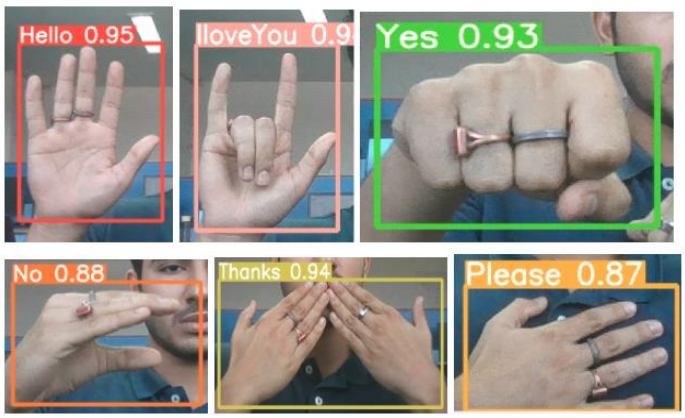

Fig. 3. Detected labels on the training set

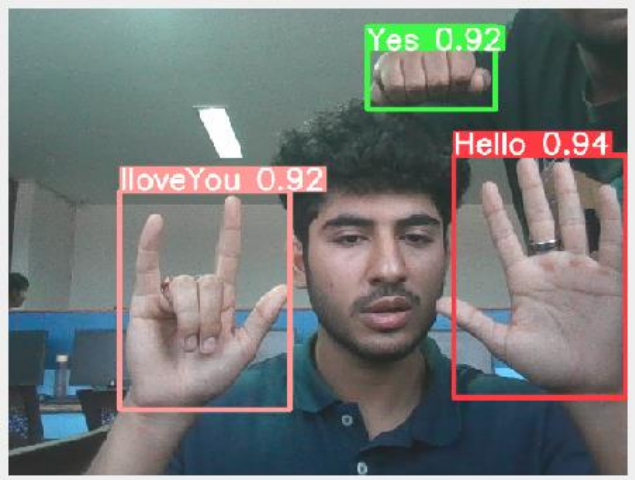

Fig. 4. detected labels on test images

IV. Results

The results are performed with a confidence threshold of 0.4, Initially, even when performed on limited data, we achieved on average 0.927 map@0.5 and 0.945 map@0.5:0.95.

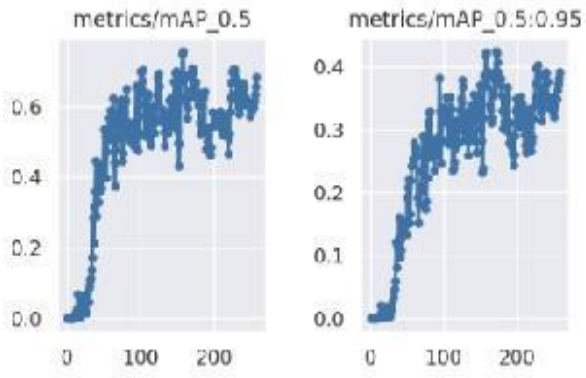

Fig. 5. evaluation graph

A. Correctly Labeled Data

The analysis of the confusion matrix (figure 6) indicates that the models are able to accurately label a significant portion of the data. Additionally, upon observing the images, it is evident that the models successfully identify the hand locations and accurately assign labels to the gestures. Moreover, the confidence values associated with the recognized gestures are consistently high.

B. Incorrectly Labeled Data

The model, despite its overall accuracy, does make some errors in word detection. It occasionally fails to provide predictions for certain letters and numbers, leaving them unclassified.. These misclassifications indicate areas where the model's performance can be enhanced for more accurate word recognition..

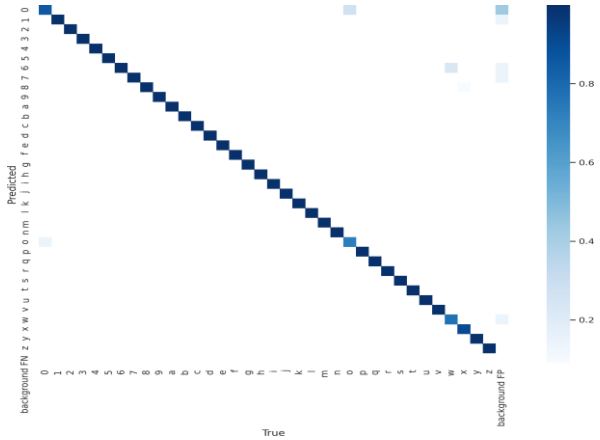

Fig. 6. confusion matrix on validation set

V. Discussion

Based on our experiment, we can draw the conclusion that even with a medium-sized dataset, the gesture detection and classification process can be performed quickly and accurately. The results obtained from the image analysis (figure 5) exhibit promising potential for utilizing the YOLOv5 algorithm in real-time gesture detection. Additionally, the model is lightweight and efficient, making it suitable for deployment on edge-based platform. Nonetheless, it is crucial to consider that the model was trained with a relatively small dataset. Given a sufficient amount of data and longer training time, it is highly probable to achieve improved results in this task.

VI. Conclusion

With a relatively small dataset, the Sign Language identification using YOLOv5 achieves an average F1 score of 0.945 for distinct classes. This performance demonstrates promising potential for utilizing YOLOv5 in recognizing Sign Language. The YOLOv5x solution comes with a lightweight pre-trained weight of only 167MB. Moreover, YOLOv5x offers high frames per second (fps), ensuring real-time gesture detection with optimal accuracy and speed. Its efficient pre-trained weight allows for deployment on various AI computing platforms, making it an excellent choice for real-time Sign Language recognition.

References

[1] A. L. C. Barczak, N. H. Reyes, M. Abastillas, A. Piccio, and T. Susnjak, "A new 2d static hand gesture color image dataset for asl gestures," *Research Letters in the Information and Mathematical Sciences*, vol. 15, pp. 12–20, 2011.

[2] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[3] H. Qu, T. Yuan, Z. Sheng, and Y. Zhang, "A pedestrian detection method based on yolov3 model and image enhanced by retinex," in *2018 11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*. IEEE, 2018, pp. 1–5.

[4] P. S. Zaki, M. M. William, B. K. Soliman, K. G. Alexsan, K. Khalil, and M. El-Moursy, "Traffic signs detection and recognition system using deep learning," *arXiv preprint arXiv:2003.03256*, 2020.

[5] V. Sharma, "Face mask detection using yolov5 for covid-19," 2020.

[6] G. Jocher, A. Stoken, J. Borovec, NanoCode012, ChristopherSTAN, L. Changyu, Laughing, tkianai, yxNONG, A. Hogan, lorenzomammana, AlexWang1900, A. Chaurasia, L. Diaconu, Marc, wanghaoyang0106, ml5ah, Doug, Durgesh, F. Ingham, Frederik, Guilhen, A. Colmagro, H. Ye, Jacobsolawetz, J. Poznanski, J. Fang, J. Kim, K. Doan, and L. Y. , "ultralytics/yolov5: v4.0 - nn.SiLU() activations, Weights & Biases logging, PyTorch Hub integration," Jan. 2021. [Online]. Available: https://doi.org/10.5281/zenodo.4418161

[7] H.-D. Yang, "Sign language recognition with the kinect sensor based on conditional random fields," *Sensors*, vol. 15, no. 1, pp. 135–147, 2015.

[8] S. A. Mehdi and Y. N. Khan, "Sign language recognition using sensor gloves," in *Proceedings of the 9th International Conference on Neural Information Processing, 2002. ICONIP'02.*, vol. 5. IEEE, 2002, pp. 2204–2206.

[9] L. T. Phi, H. D. Nguyen, T. Q. Bui, and T. T. Vu, "A glove-based gesture recognition system for vietnamese sign language," in *2015 15th International Conference on Control, Automation and Systems (ICCAS)*. IEEE, 2015, pp. 1555–1559.

[10] L. Pigou, S. Dieleman, P.-J. Kindermans, and B. Schrauwen, "Sign language recognition using convolutional neural networks," in *European Conference on Computer Vision*. Springer, 2014, pp. 572–578.

[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, pp. 1097–1105, 2012.

[12] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[13] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning. thirty-first aaai conf," *Artif. Intell*, 2017.

[14] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[15] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.

[16] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real- time object detection with region proposal networks," *arXiv preprint arXiv:1506.01497*, 2015.

[17] R. Sharma, Y. Nemani, S. Kumar, L. Kane, and P. Khanna, "Recognition of single handed sign language gestures using contour tracing descriptor," in *Proceedings of the world congress on engineering*, vol. 2, 2013, pp. 3–5.

[18] B. Garcia and S. A. Viesca, "Real-time american sign language recognition with convolutional neural networks," *Convolutional Neural Networks for Visual Recognition*, vol. 2, pp. 225–232, 2016.

[19] P. Rathi, R. Kuwar Gupta, S. Agarwal, and A. Shukla, "Sign language recognition using resnet50 deep neural network architecture," *Available at SSRN 3545064*, 2020.

[20] L. Torrey and J. Shavlik, "Transfer learning," in *Handbook of research on machine learning applications and trends: algorithms, methods, and techniques*. IGI global, 2010, pp. 242–264.

[21] S.-K. Ko, J. G. Son, and H. Jung, "Sign language recognition with recurrent neural network using human keypoint detection," in *Proceedings of the 2018 Conference on Research in Adaptive and Convergent Systems*, 2018, pp. 326–328.

[22] P. T. Krishnan and P. Balasubramanian, "Detection of alphabets for machine translation of sign language using deep neural net," in *2019 International Conference on Data Science and Communication (IconDSC)*. IEEE, 2019, pp. 1–3.

[23] P. Liu, X. Li, H. Cui, S. Li, and Y. Yuan, "Hand gesture recognition based on single-shot multibox detector deep learning," *Mobile Information Systems*, vol. 2019, 2019.

[24] S. Kim, Y. Ji, and K.-B. Lee, "An effective sign language learning with object detection based roi segmentation," in *2018 Second IEEE International Conference on Robotic Computing (IRC)*. IEEE, 2018, pp. 330–333.

[25] J. Redmon and A. Farhadi, "Yolo9000: better, faster, stronger," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7263–7271.

[26] ——, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[27] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: Op- timal speed and accuracy of object detection," *arXiv preprint arXiv:2004.10934*, 2020.

[28] C.-Y. Wang, H.-Y. M. Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, and I.-H. Yeh, "Cspnet: A new backbone that can enhance learning capability of cnn," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 2020, pp. 390–391.

[29] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8759–8768.

[30] R. Xu, H. Lin, K. Lu, L. Cao, and Y. Liu, "A forest fire detection system based on ensemble learning," *Forests*, vol. 12, no. 2, p. 217, 2021.