Grenze International Journal of Engineering and Technology, Jan Issue

Hand Sign Recognition using YOLOV5

Alen Charuvila Saji [1], K.Ramalakshmi[2], Senbagavalli M[3], Hemalatha Gunasekaran[4] and Shamila Ebenezer[5]
[1]B. Tech, Computer Science and Engineering, K.I.T.S., India
Email: alencsaji@gmail.com
[2]Associate Professor, Computer Science and Engineering, Alliance University,Bangalore
Email: ramalakshmivenkatesan@gmail.com
[3]Associate Professor, Information Technology, Alliance University, Bangalore
Email: senba1983@gmail.com
[4]IT Department, University of Technology and Applied Sciences, Oman
Email: hemalatha.ibr@cas.edu.om
[5]Assistant Professor, Computer Science and Engineering, K.I.T.S., India
Email: shamila_cse@karunya.edu

***Abstract*—The deaf-mute community utilises sign language for interacting among themselves and others. The introduction of standard sign language has made their lives much easier. This paper proposes an effective hand-sign recognition method using a deep learning technique and is based on YOLOv5, which is a real-time object detection algorithm which detects a hand sign and outputs the corresponding text. The proposed model utilises various sub-models namely, Cross Stage Partial Network (CSPNet), Path Aggregation Network (PANet), Dense Prediction. This model can be conveniently deployed into an android application with a user-friendly interface.**

***Index Terms*— YOLOv5; classification; arrhythmia;deep learning; convolution deep learning; Webservices.**

I. INTRODUCTION

One of the pressing issues that the disabled community faces today is a lack of adequate technology to communicate. While today'stechnology addresses many everyday issues, itseldom caters to the needs of these people. Around 466 million people in the world suffer with hearing impairment which is over 5% of our world's population. There are 34 million people having impairment among this 50% are children who are the most affected. This group which mostly includes school-goers and adolescents experience social stigma affecting their academic performance and subsequently causing decline in their overallwell-being. The introduction of the sign language in the 18th century made their lives a little better but it is not enough. The only source of communication for deaf-mute children is sign language. There is a need for systematic and effective solution to bring down the barrier in communication. The Children often are marginalized in the society because of this. It is the need of the hour to introduce an image-based hand sign recognition technology to convert non-verbal language to intelligible text.

Grenze ID: 01.GIJET.8.1.32

II. OBJECTIVE

The primary objective of this model is to classify different hand-sign images to different classes using a YOLOv5 model to help the hearing-impaired individuals. We also aim to deploy this model as an android application with a user-friendly interface which can be easily accessed topredict the hand-signs and is a much faster and efficient way.

III. PROBLEM STATEMENT

Deaf-mute community face challenges while communicating in their day-to-day life. There should be a cost-effective and efficient solution to this based on technology.

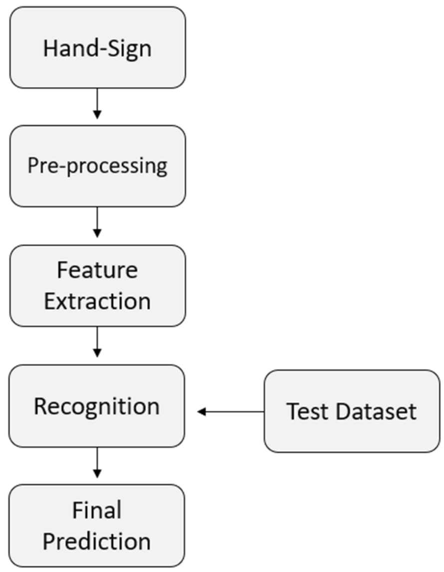

Fig. 1. Flow diagram of hand sign recognition system

A. Literature Survey

[1] Diksha Hatibaruah, Anjan Kumar Talukdar and Kandarpa Kumar Sarma, have created an SL interpreter that accepts a sign gesture as input and displays the result on a display device. To train the system with a particular database, they used Convolutional Neural Networks (CNNs). They employed the Indian SL database for our study, which has 26 alphabets and 10 digits, and used the Histogram BackProjection approach for picture segmentation. After that, the datasets are divided into classes, which are then fed into a CNN for training and testing. At 5 epoch, we discovered that the testing accuracy was 99.89 percent and the validation accuracy was 99.85% after training. The system is then put to the test with real-time input, and the results are displayed on the display device.

[2] Hongxu Ma, Qiang Wang, Xiang Ma, and Mohamed E. M. Salem, developed a method for simplifying the definition of human hand movements, then implemented human hand movement detection, the fabrication of a human hand shape, and a mechanical structure capable of performing sign language gestures. The soft hand's tracking and interaction abilities with human hand movement were exhibited in a variety of tests, and data on human hand movement was collected and analysed. In terms of sign language interactive terminals and other associated features, the system has a lot of engineering potential.

[3] DardinaTasmere and Boshir Ahmed proposes a new hand gesture recognition framework - to overcome the large communication gap between deaf and non-sign language users, presents a new hand gesture detection framework for Bangla sign language. HSV and YC b C r colour spaces were practised by the hand. Deep convolution neural networks can recognise a total of 37 characters (8 vowels and 29 consonants). They used the Bangla sign language to create 37 classes for 37 alphabets, and their framework also helped the gesture recognition system by providing a new dataset for the Bangla sign language. A total of 3219 photos from six different persons make up the collection. This new dataset allows us to achieve a 99.22% accuracy rate.

[4] Sandrine Tornay, Marzieh Razavi, Mathew Magimai.-Doss Hand movement modelling is also done using target sign language independent data by derivation of hand movement subunits in a multilingual sign language method. Validating the proposed approach through research into Swiss German Sign Language, German Sign Language, and Turkish Sign Language, as well as demonstrating that sign language recognition systems can be constructed efficiently by using multilingual sign language resources.

[5] Xianzhi Chu, Jiang Liu, and Shigeru Shimamoto proposed a sensor-based data acquisition glove for Japanese Sign Language (JSL) hand gesture recognition. To detect the bending degree of fingers and hand movement information, five flex sensors, an Inertial Measurement Unit (IMU), and three Force Sensing Resistors (FSRs) are employed. An Arduino Micro transmits the detected data to the computer. For a single individual, the average accuracy of hand gesture identification utilising the Support Vector Machine (SVM) and Dynamic Time Wrapping (DTW) algorithms is 96.9% and 94.5 percent, respectively. For cross-identification among three subjects, the suggested approach obtains an average recognition accuracy of roughly 82.5 percent. The experimental findings show that our suggested system has a lot of potential for recognising JSL hand gestures.

[6] DardinaTasmere, Boshir Ahmed, and Md Mehedi Hasan, presented a novel real-time method for Bangla sign digits (0-9) that focuses on three sections: image acquisition, pre-processing, and lastly, recognition of Bangla sign digits. For cross-identification across three individuals, the suggested approach obtains an average recognition accuracy of around 82.5 percent. Our suggested system has a lot of potential for JSL hand gesture recognition, according to the results of the experiments.

IV. Proposed Work

YOLO is a real-time object detection algorithm and it has 5 versions currently. It is extremelyfast and accurate. In map measured at .5 IOUYOLOv3 is on par with Focal Loss but about 4xfaster. Moreover, it can easily trade-off betweenspeed and accuracy simply by changing the size ofthe model, no retraining required.We are using YOLOv5 which is also the latestversion.It uses PyTorch framework and has very fast inferences. YOLOv5 has noticeable performance improvement like training and increasedperformance, including multi-scale predictions, abetter backbone classifier, and more when comparedto the previous versions.YOLO uses a totally different approach. It applies asingle neural network to the full image. This network divides the image into regions and predicts bounding boxes and probabilities for each region.These bounding boxes are weighted by the predicted probabilities.

Backbone - Cross Stage Partial Network (CSPNet)Used to extract important features from images.CSPNet has significant improvement in processingtime and requires less computational power.

Model Neck- Path Aggregation Network (PANet)Used to generate feature pyramids, which helps toidentify the same object with different sizes andscales. Helps the model to perform well on unseendata.

Model Head - Dense Prediction (used in one-stage-detection algorithms)

The model Head is mainly used to perform the finaldetection part. It applied anchor boxes on featuresand generated final output vectors with classprobabilities, objectness scores, and boundingboxes.

Activation Function: Leaky ReLU - f(x)=max(0.01*x , x). This function returns x if it receives any positive input, but for any negative value of x, it returns a really small value which is 0.01 times x., Sigmoid - f (s) = 1 1 + e − s, where s is the input and f is the output. In YOLO v5 the Leaky ReLU activation function is used in middle/hidden layers and the sigmoidactivation function is used in the final detectionlayer.

Optimizer: SGD (default), Adam Optimizers shape and mold a model into its most
accurate possible form by learning from the weights.The loss function is the guide to the terrain, tellingthe optimizer when it's moving in the right or wrongdirection.

Loss function: Binary Cross-Entropy with LogitsLoss (default), Focal loss

A loss function is used to optimize the parameter values in a neural network model. Loss functions map a set of parameter values for the network onto a scalar value that indicates how good a prediction model does in terms of being able to predict the expected outcome (or value).YOLOv5 have 4 models: yolov5-s, yolov5-m,yolov5-l, yolov5-x.

From the above comparison it is clear that YOLOv5x has better performance but will take more time than other models. We are training withthe YOLOv5x model.

Pre-Processing

For training YOLOv5 requires the dataset to be inYOLO(Darknet) format. That is training andvalidation images have to be in train and validationfolders respectively and these two folders should beplaced in images folder. Similarly labels have to beplaced in the labels folder. Both labels and imagesshould be placed in a single folder.

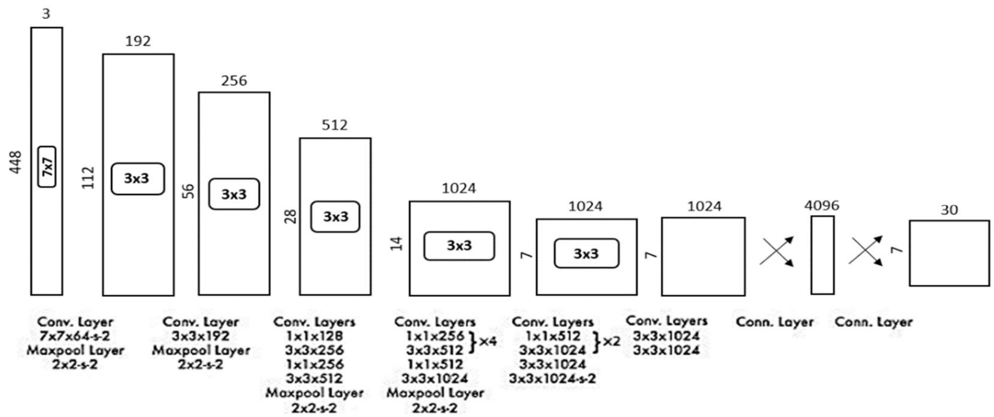

Fig. 2. Layered Architecture

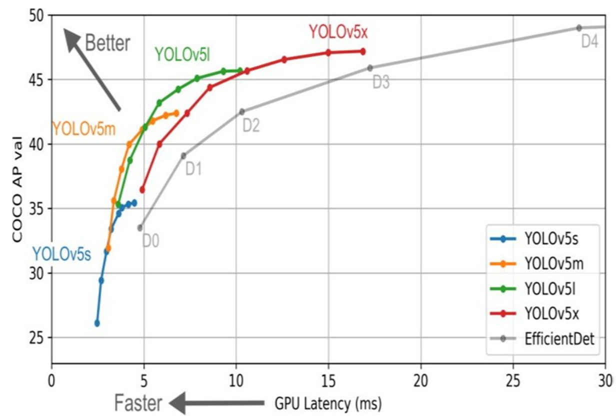

Fig. 3. Performance of proposed technique: YOLOv5x

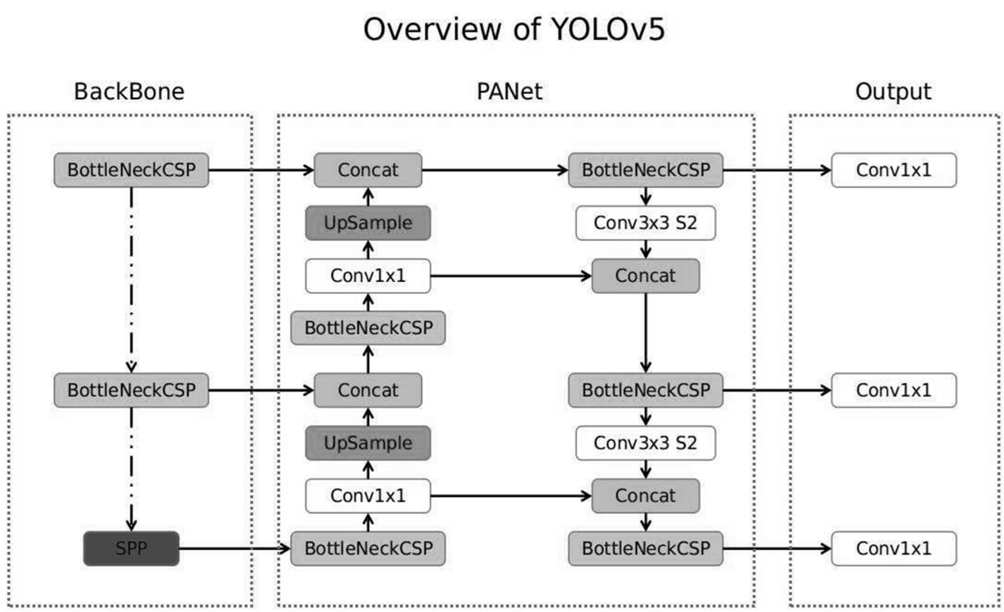

Fig. 4. Overview of YOLOv5

This format willhave a text annotation file for each image in thetraining set. Best practice would be to keep 70% datain the training set, 20% in the validation set, and 10% in the testing set.
A label for an image is an annotation file in textformat which contains id for the class and thenumerical representation of the binding boxes(labelled part). These four values comes in range of[0,1].

Training
First, clone the YOLOv5 repo from GitHub to the environment to train. In my case it is Google Colab. Install the dependencies using the pip command. Weneed two configuration files to train the model,amute.yaml. A configuration yaml file that contains path to the stored dataset, number of classes and classnames respectively. yolov5x.yaml containsnumber of classes to train. Add these configurationfiles to the cloned repo. Start training the model withthe above configuration files, yolov5x.yaml and dataset file (amute.yaml file). Save the output weights toa path. Run the detect.py with the output weight asparameter with a confidence value and the folderpath for testing.
• Number of images to train - 2320
• Number of images for validation - 580
• Best precision obtained - 0.844 (84%)
• Time taken - 6.416 hours
• Batch Size - 4
• Epoch – 100
• Output weights - best_amute.pt, last_amute.pt

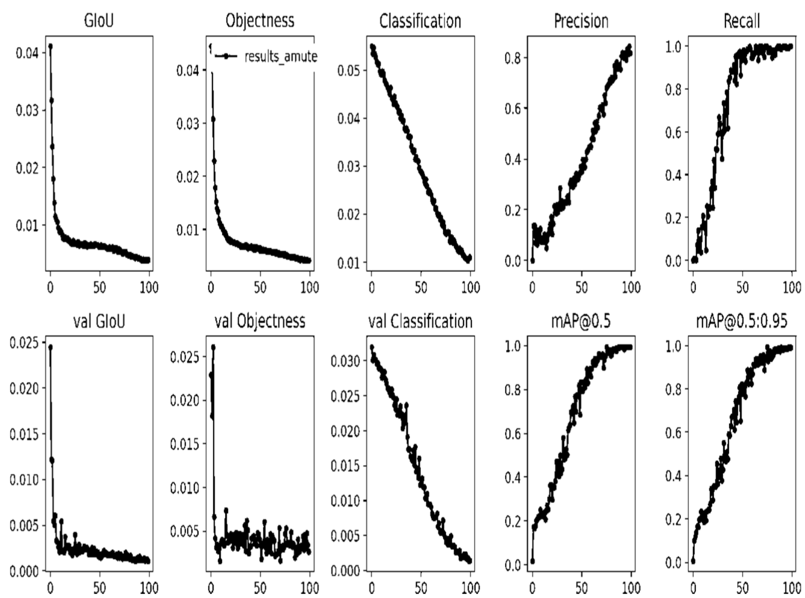

Fig. 5. Evaluation of Proposed System using different metrics

V. ANDROID APPLICATION – AMUTE

Functional requirements
Graphical User interface with the User.

Software requirements
IDE : Android Studio 4.1.2
Libraries : OpenCV – 3.2.0
Coding languages : Java, XML
Operating System : Windows 10
The Graphical User Interface (GUI) consists of acamera frame as the first element in thesystemfollowed by a text box below it. The signs shown bythe user on the phone’s camera can be seen in thecamera view frame and the text output for each signis displayed in the text box.On the backend, we can process each frames fromthe camera view. This enables us to captureparticular frames in use by converting the frame intoan image file. For

example, a hand-sign shown in theframe is converted into an image and this image canbe used as the input for sign prediction. The resultanttext will be displayed the text box.

VI. INTEGRATION

In order to integrate my saved weight to my android application, the pytorch weight(best_amute.pt)should be converted to a tensorflow graph or model(.pb) and then this converted model is again optimized to tensorflow Lite model (.tflite). This new optimized model can be integrated and further used in for prediction.

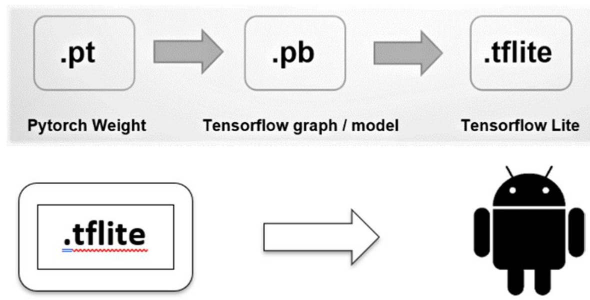

Fig. 6. Convertion from Pytorch weight to Tensorflow lite

VII. CONCLUSION

This report highlights the use of machine learningand mobile technology to enhance the lives of thedeaf-mute. The Amute application uses sign-to-texttechnique to break down any barriers enablingcommunication with ease. The application mostlyaims to assist theadolescents and teenagers as theyare prone to become depressed and ideate suicidalthoughts. Youngsters should rise beyond thereimpairments and not limit themselves. Thisapplicationis the ultimate solution for the deaf-mutes.

REFERENCES

[1] Diksha Hatibaruah; Anjan Kumar Talukdar; Kandarpa Kumar Sarma,"A Static Hand Gesture Based Sign Language Recognition System using Convolutional Neural Networks", IEEE India Conference (INDICON),2020.
[2] Hongxu Ma; Qiang Wang; Xiang Ma; Mohamed E. M. Salem," A Sign Language Interaction System Based on Pneumatic Soft Hand", IEEE Conference on Industrial Electronics and Applications (ICIEA),2020.
[3] Dardina Tasmere; Boshir Ahmed ,"Hand Gesture Recognition for Bangla Sign Language Using Deep Convolution Neural Network", International Conference on Sustainable Technologies for Industry 4.0 (STI),2020.
[4] Sandrine Tornay; Marzieh Razavi; Mathew Magimai.-Doss," Towards Multilingual Sign Language Recognition", International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 2020.
[5] Xianzhi Chu; Jiang Liu; Shigeru Shimamoto," A Sensor-Based Hand Gesture Recognition System for Japanese Sign Language", IEEE Global Conference on Life Sciences and Technologies (LifeTech),2021.
[6] Dardina Tasmere; Boshir Ahmed; Md Mehedi Hasan, "Bangla Sign Digits: A Dataset For Real Time Hand Gesture Recognition", International Conference on Electrical and Computer Engineering (ICECE),2020.
[7] Kshitij Bantupalli; Ying Xie " American Sign Language Recognition using Deep Learning and Computer Vision"
[8] Rajesh George Rajan;M. Judith Leo " American Sign Language Alphabets Recognition using Hand Crafted and Deep Learning Features"
[9] Hand gesture recognition using machine learning algorithms by Abhishek B1, Kanya Krishi, Meghana M, Mohammed Daaniyaal, Anupama H, B.E, Computer Science and Engineering, BMS Institute of Technology, Bangalore, India, Associate Professor, BMS Institute of Technology, Bangalore, India
[10] G. Pala, J. B. Jethwani, S. S. Kumbhar and S. D. Patil, "Machine Learning-based Hand Sign Recognition," 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS), 2021, pp. 356-363, doi: 10.1109/ICAIS50930.2021.9396030.
[11] P. N. Huu and H. L. The, "Proposing Recognition Algorithms For Hand Gestures Based On Machine Learning Model," 2019 19th International Symposium on Communications and Information Technologies (ISCIT), 2019, pp. 496-501, doi: 10.1109/ISCIT.2019.8905194.
[12] K. K. Dutta and S. A. S. Bellary, "Machine Learning Techniques for Indian Sign Language Recognition," 2017 International Conference on Current Trends in Computer, Electrical, Electronics and Communication (CTCEEC), 2017, pp. 333-336, doi: 10.1109/CTCEEC.2017.8454988.