Volume: 10 Issue: IX Month of publication: September 2022

DOI: https://doi.org/10.22214/ijraset.2022.46928

Real-time Telugu Sign Language Translator with Computer Vision

Varshini Dusa[1], Satya Shodhaka R Prabhanjan[2], Sharon Carlina Chatragadda[3], Sravani Bandaru[4], Ajay Jakkanapally[5]
[1, 3, 4, 5]Student, Department of CSE, VNR VJIET, Hyderabad, Telangana, India
[2]Student, Department of IT, CBIT, Hyderabad, Telangana, India

Abstract: Sign language is the basic communication method among hearing disabled and speech disabled people. To express themselves, they require an interpreter or motion sensing devices who/which converts sign language in a few of the standard languages. However, there is no system for those who speak in the Telugu language and hence they are forced to speak in the national language over the regional language of their culture along with the same issues of cumbersome hardware or need for an interpreter. This paper proposes a system that detects hand gestures and signs from a real-time video stream that is processed with the help of computer vision and classified with object detection YOLOv3 algorithm. Additionally, the labels are mapped to corresponding Telugu text. The style of learning is transfer learning, unlike conventional CNNs, RNNs or traditional Machine Learning models. It involves applying a pre-trained model onto a completely new problem to solve the related problem statement and adapts to the new problem's requirements efficiently. This requires lesser training effort in terms of dataset size and greater accuracy. It is the first system developed as a sign language translator for Telugu script. It has given the best results as compared to the existing systems. The system is trained on 52 Telugu letters, 10 numbers and 8 frequently used Telugu words.

Keywords: Sign language translator, Computer Vision, Transfer Learning, YOLOv3, Object Detection, Darknet.

I. INTRODUCTION

When we are talking to others, we use our voice to express ourselves and use our hearing ability to listen to what others have to say. However, when one of these abilities is inhibited, we resort to gestures. We cannot accommodate the complexities of a language beyond a certain degree of conversation with a deaf or mute person using common hand gestures. To ensure communication isn't hindered by any obstacles, a sign-to-text system is proposed, that receives real time input and translates it to Telugu language, so that communication does not have to stop. This sign translator system mainly aims to help disabled people to be independent in terms of communication.

With the real-time processing system for translating the sign language, they do not need a translator with them wherever they go. It interprets the sign language to Telugu language text and can be translated into desired languages. This reaches out to the community of people who had barriers with the language and opens up a whole new possibility of a new trainable workforce and thus enriching their lives while we learn from their resilience and spirit in collaboration with them. This step implemented properly would eradicate any disparity in challenged people because technology has to power to bring lives together in a more meaningful way.

Previous research work carried out in this field, paved a way to develop various models related to image preprocessing from multiple input channels and classification of pre-processed segments to translate into text, based on the dataset used.

II. LITERATURE SURVEY

Shivashankara S, Srinath S [1] proposed a system that translates 24 static ASL alphabet gestures and 10 static ASL numbers gestures into English text. In this system, preprocessing image is done using HSV (Hue, Saturation, Value) to detect skin region mark pixels, YCbCr color model to optimize functioning of skin clustering and recognize gesture using centroid, area, perimeter, peak offset and number of peaks.

Victoria A. Adewale, et al. [2] proposed a system to convert hand gestures into text using unsupervised learning. In this system, image segmentation and object detection using Speeded Up Robust Features and Features from Accelerated Segment Test algorithms and classification using K-Nearest Neighbour showed that unsupervised learning classification could determine the best matched feature from the existing database.

Shaik Khadar Sharif, et al. [3] proposed a comparative study of feature extraction methods of thresholding and RGB converted datasets. It is done for preprocessing of image obtained from Region of Interest (ROI) and Recurrent Neural Network algorithm of Keras Model is used for classification.

Sawant Pramada, et al. [4] proposed a system that use image processing, machine learning and artificial intelligence to identify hand gestures developed used Binary Sign Language. In this system, the input is processed by RGB color recognition, thresholding, coordinate mapping, color calibration and matched with the Pattern matching algorithm where comparisons are made for each image. Template matching of templates from coordinates from pre-processed image and alphabet recognition using binary finger tapping tool are done.

CasamNjagi NYAGA, Ruth Diko WARIO [5] conducted usability test on an existing gesture recognition system. The system discussed uses OpenCV module to record input, blur and perform thresholding, draw contours and display results. There is a good response to sign in results obtained.

YuanchengYe, et al. [6] proposed 3DRCNN model that incorporates 3D convolutional neural network (3DCNN) and enhances fully connected recurrent neural network (FC-RNN), where 3DCNN acquires multi-modality features from RGB, motion, and depth channels, and FCRNN captures the temporal information among short video clips split from the original video. Consecutive clips with the same semantic meaning are singled out by applying the sliding window approach to segment the clips on the entire video sequence.

Paulo Trigueiros, et al. [7] proposed a vision-based hand gesture recognition system that interprets Portuguese Sign Language. In this system, data acquisition is done by video input. OpenCV is used for vision-based extraction operations and OpenNI for depth image acquisition. Classification is done using Dlib library, that uses SVM to work with high dimensional data.

Mahesh Kumar [8] proposed system that recognizes 26 hand gestures in Indian Sign Language using MATLAB software. In this system, feature extraction is done using Eigen values and Eigen vectors. The Linear Discriminant Analysis (LDA) algorithm was utilized for gesture recognition which is converted into text format. The proposed system helps in dimensionality reduction.

Alhussain Akoum, Nour Al Mawla [9] discussed steps to take input, recognize and analyze hand gestures, then translating into text. In this approach, a digital camera provides input and background is eliminated using thresholding and filtering. Then, comparison is done with all existing images in the database via correlation and edge detection to get a result.

Josep Querl, et al. [10] discusses socio-linguistic aspects and methodological problems of the signing community. General vision-based techniques are discussed outlining the challenges posed by vision in terms of tracking and speed of signing.

J.Rekha, et al.[11] proposed two approaches of hand gesture recognition in real time environment. A hybrid of SURF and Hu Moment Invariant methods is one approach while other one is adding derived features to previous one. Both use KNN and SVM algorithms for classification. The latter proves to be more robust than the former. Classification done using KNN algorithm attempts to estimate the underlying distribution of the data in a non-parametric fashion.

Uttam Prabhu Dessai, et al. [12] proposed sign language translation system for American Sign Language. In this approach, Singular Value Decomposition is used for feature extraction and Support Vector Machine uses a technique called the kernel trick that transforms data and then based on these transformations, it finds an optimal boundary between the possible outputs.

Shailesh bachani, et al. [13] presented a real time system for hand gesture recognition. In this approach, subsampling is a layer where all induvidual units within the layer has a receptive field of a fixed size that is imposed on the input, used for feature extraction. CNN algorithm is used for classification, where each specific neuron receives multiple inputs and then takes a weighted sum over them, where it passes it through an activation function and responds back with an output based on weights and biases.

Omkar Vedak, et al.[14] implemented ISL translator. In this system, histograms of Oriented Gradients are used for Feature extraction, where feature descriptor is used for computer vision and image processing for the purpose of object detection. SVM is used for Classification and Recognition.

Eleni Efthimiou, [15] proposed Dicta-Sign project that offers unified framework for sign language recognition, animation and language modeling. A Hidden Markov Model is built, which is a statistical Markov model where the system is presumed to be a Markov process with unobservable states. Signing Gesture Markup Language (SIGML) is developed for sign language gestures. ANVIL and ELAN annotation tools are used for labeling signs. This project aims to advance enabling technologies for communication in multi-disciplinary approach.

Hanine El Hayek, et al. [16] proposed sign to letter translator system using hand glove that gives output on LCD screen. In this approach, glove with flex sensors gives input to microcontroller that uses flex values to predict signs.

Pooja Sonawane, Anitha Nikalje [17] proposed text to ISL translation. This system uses techniques like gray scale convergence, edge detection, array of image generation for pre-processing. Classification is done by pattern matching.

Yellapu Madhuri, et al. [18] proposed a mobile vision based interactive application program that uses LabVIEW systems engineering software.

It acquires image from in-built mobile camera, converts to feature vectors, processes and extracts features using color thresholding and pattern matching for classification.

Steve Daniels, et al. [19] developed a sign language recognition system that translates real time video input using YOLOv3 pre-trained model, that is trained on required configuration to identify hand signs. Image preprocessing is done on images collected from input. The model is run on both image and video data to compare the accuracy of the results obtained.

III. PROPOSED SYSTEM

To overcome the obstacles of higher degree communication in the people with disabilities of hearing and speech, we propose a real-time sign translator system that can identify hand gestures, process them, provides labels, and converts them to text as output to the hand gesture the end user has given. Object detection model, YOLOv3 is used to build this system. YOLOv3 uses Darknet-53 for feature extraction, which has 53 convolutional layers resulting in high accuracy.

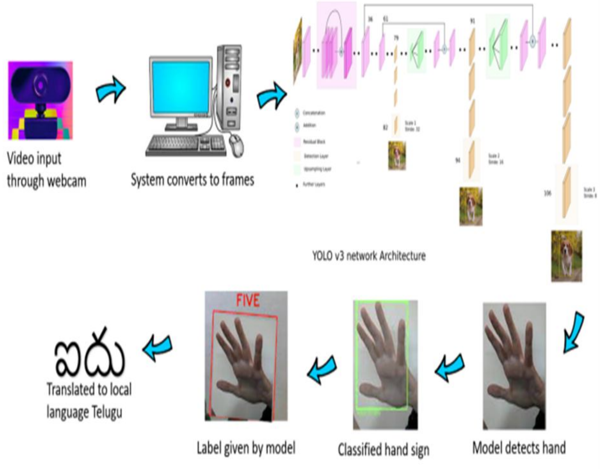

Video input through webcam
System converts to frames
YOLO v3 network Architecture
Model detects hand
Classified hand sign
FIVE
Label given by model
ఐదు
Translated to local language Telugu

Fig.1. Process Flow of sign language translator

As the flow diagram (see Fig. 1.) describes, hand gestures of the user will be taken as a real time video input captured using webcam. Then, it is converted into frames by OpenCV. YOLOv3 identifies the specific hand gesture information from the frames by converting image into grid. Using predefined classes, YOLOv3 gives scores to the objects and grid cells. It predicts number of boundary boxes around objects which scores high. Thereafter, the model classifies the hand sign and assigns a label to the hand sign. The labels are mapped to Telugu language letter, word or number conforming to the dataset. This system enables communication without extra hardware.

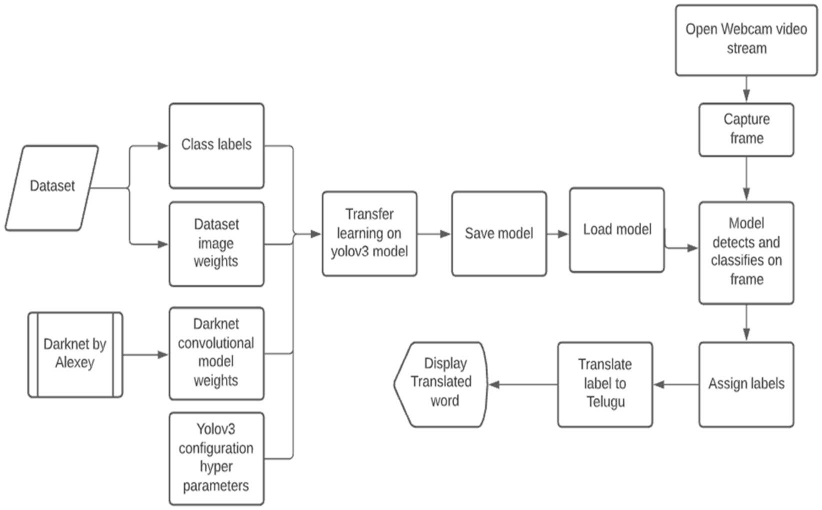

Open Webcam video stream
Capture frame
Dataset
Class labels
Dataset image weights
Darknet by Alexey
Darknet convolutional model weights
Yolov3 configuration hyper parameters
Transfer learning on yolov3 model
Save model
Load model
Model detects and classifies on frame
Assign labels
Translate label to Telugu
Display Translated word

Fig.2. Framework of Sign Language Translator

There are various components in the framework (see Fig. 2.) of Sign Language Translator system. Initially, the dataset with class labels and image weights is generated. Along with dataset, Darknet pre-trained model weights and manually fine-tuned YOLOv3 hyper parameters are given to the YOLO model to carry out transfer learning. The trained model is saved as a weights file. On the other hand, the video input is streamed using web camera. The model is run on a captured frame of the video stream. It detects and classifies the sign in the frame. Thereafter, it assigns labels and translating to Telugu. Finally, the text is displayed.

The algorithm comprises of deep learning framework Darknet, which uses neural networks that simulate how the brain perceives.

A. Darknet

Darknet is a collection of open-source neural networks written in C and CUDA. It is very fast, easy to install and provides support for GPU and CPU. It also uses different deep learning models, which makes object detection fast and accurate. The central model used in this system is YOLOv3 trained by transfer learning from darknet.conv.53.

B. YOLO

YOLO is a real-time object detection model that identifies specific objects in given input video or images. Unlike many object detection models that need to go through the object for several times to detect the image, YOLO takes single look and predicts the type of object, thus named YOLO, "You Only Look Once". It takes entire image in a single instance and tries to predict it. It can also take generalized representation of objects. It is incredibly fast and accurate as it can process up to 45 frames per second. YOLO mainly uses Convolutional Neural Network; specifically, Darknet is the best option among Neural Networks it can work with. With time, YOLO has introduced multiple versions like YOLOv1, YOLOv2, and YOLOv3.

The first version of YOLO can detect only one object which falls into the grid. It also has many issues like close object detection, that is, it cannot recognize multiple images in a single image and has low confidence level of object prediction. It uses a single convolution network that gives up to 88% accuracy rate and has a major drawback of recognizing only 49 objects. The next version introduced is YOLOv2.

Itis an extended version of YOLOv1 with additional features like batch normalization, high resolution classification, convolution with anchor boxes and Darknet-19 Network Architecture ,backbone for YOLOv2, with 19 convolution layers and 5 max pooling layers with an accuracy of 91.2% which is better than YOLOv1.

The algorithm used in our system is YOLOv3.It is pre-trained model which is more efficient and faster than earlier versions because it is the hybrid approach between YOLOv2 and Darknet-19.

It is one of the fastest detection models with GPU of M40 or TitanX and uses 53 convolution layers, also called as Darknet-53 with 3*3 and 1*1 successive size.

C. Working of YOLOv3

The working of YOLOv3 starts by dividing the image into cells/grids which are responsible to predict bounding boxes, confidence level and class probability. Bounding boxes uses a specific dimension clusters as anchor boxes. It mainly consists of 4 coordinates for each bounding box on 4 sides. It tries to predict an object score by using Logistic Regression, width and height by using cluster centroids. In next step of prediction of classes, each bounding box is used to predict the class of the objects. Multilabel classification and independent logistic classifiers are used for classification of those classes. During training, it uses binary- cross entropy for class prediction as this helps to work more conveniently with complex data like Open Image Datasets, that has many overlapping labels (like man, persons) within an image, where multi label classification plays a key role in better prediction. Another most important function in YOLOv3 is prediction across scales. One of the best features of YOLOv3 is high prediction rate. It predicts objects at 3 different scales by using several convolution layers. Those are

1) 3D encoding bounding box
2) Objectness
3) Class prediction

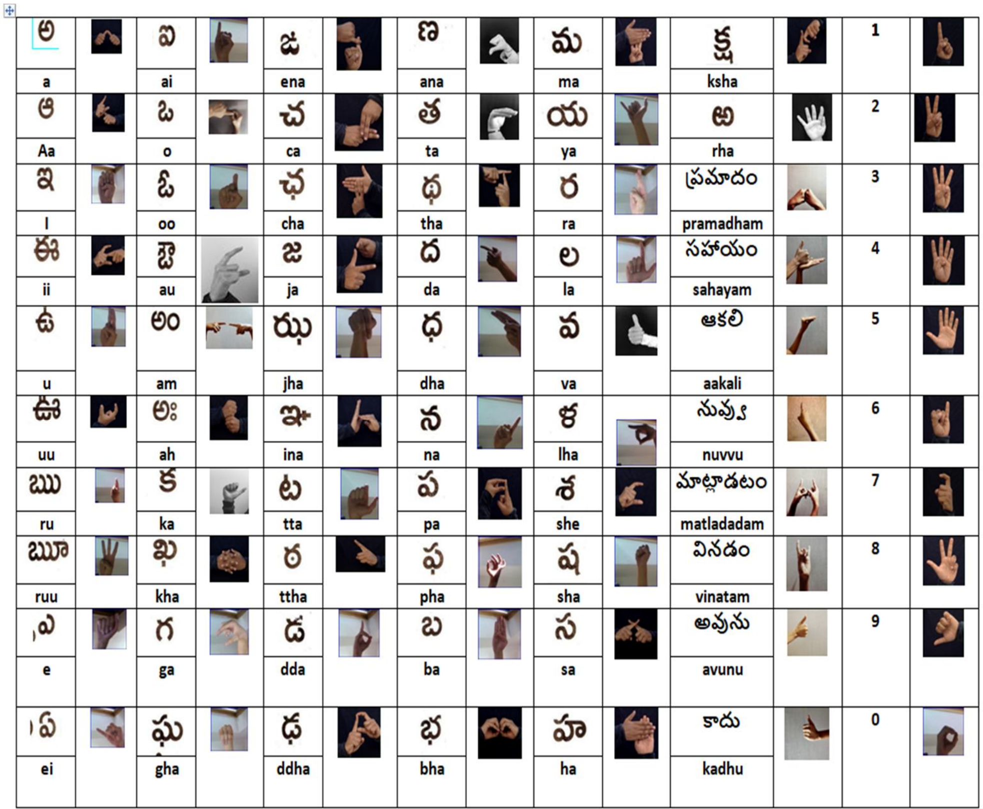
As the convolution layers increases, it becomes more efficient and accurate in prediction and K- means clustering is used to determine bounding box priors. Here, it sorts 9 clusters and 3 scales and divide them evenly across those scales.

Overall, YOLOv3 uses multi-scale and end-to-end training with many data augmentation, batch normalization and Darknet neural network frameworks for training and testing of datasets.

D. Dataset

A set of signs for each Telugu letters and few crucial words are developed, by consolidating gestures from American Sign Language, Indian Sign Language, Arabic Sign Language, British Sign Language and many more sign languages widely used and the signs for numbers are taken from American Sign Language, to create general hand gestures easily understood.

Fig.3. depicts the dataset we have employed.



అ	a	ఐ	ai	ఙ	ena	ణ	ana	మ	ma	క్ష	ksha	1
ఆ	Aa	ఒ	o	చ	ca	త	ta	య	ya	ఱ	rha	2
ఇ	I	ఓ	oo	ఛ	cha	థ	tha	ర	ra	ప్రమాదం	pramadham	3
ఈ	ii	ఔ	au	జ	ja	ద	da	ల	la	సహాయం	sahayam	4
ఉ	u	అం	am	ఝ	jha	ధ	dha	వ	va	ఆకలి	aakali	5
ఊ	uu	అః	ah	ఞ	ina	న	na	ళ	lha	నువ్వు	nuvvu	6
ఋ	ru	క	ka	ట	tta	ప	pa	శ	she	మాట్లాడటం	matladadam	7
ౠ	ruu	ఖ	kha	ఠ	ttha	ఫ	pha	ష	sha	వినడం	vinatam	8
,ఎ	e	గ	ga	డ	dda	బ	ba	స	sa	అవును	avunu	9
ఏ	ei	ఘ	gha	ఢ	ddha	భ	bha	హ	ha	కాదు	kadhu	0

Fig.3.Sign Language for numbers, Telugu alphabets and words

E. Algorithm Of The Sign Language Translator

Algorithm to do Transfer Learning for sign language translator model and saving the model:

1) Connection of Google Colab to Google Drive
2) Edit Darknet makefile to enable GPU and OpenCV usage
3) Verify CUDA
4) Run make command to generate make file for the makefile
5) Create classes names
6) Create labelled data files
7) Create train files
8) Create test files
9) Check for Darknet function
10) Run darknet detector train command on labelled data, configuration file and pre-trained Darknet model weights
11) Test the model that is generated using darknet detector test command

F. Algorithm To Load The Saved Model And Get Output To The Video Input

1) 1.Connection of Google Colab to Google Drive
2) Edit Darknet makefile to enable GPU and OpenCV usage
3) Verify CUDA

4) Run make command to generate make file for the makefile
5) Import required dependencies like NumPy, PIL, JavaScript
6) Import Darknet
7) Load configuration file, labelled data and saved model weights file
8) Declare Darknet helper functions to perform object detections
9) Declare function to convert JavaScript object to OpenCV image
10) Declare function to convert OpenCV rectangle bounding box on byte string of video stream
11) Declare JavaScript code to create live video stream using webcam as input
12) Start video streaming from webcam
13) Initialise bounding box to empty
14) Convert JS response to OpenCV image
15) Create transparent overlay for bounding box
16) Call Darknet helper to perform pre-processing
17) Loop through detections and drawing on transparent overlay image
18) Get label of sign
19) Run translator on label to print Telugu text
20) Convert overlay of bounding box into bytes
21) Update bounding box for next frame to get new overlay

IV. RESULTS

To see how the model perfomed with less computational resources feasible for anyone to work with, the training was carried out in Google Colab.The learning rate set at 0. 001.The height and width of images in dataset provided to it are modified to take maximum of 416 units. It is trained on 3 channels, that is detection layers. The average loss recorded is 1.918.

The accuracy achieved on image data provided to the system is 91% in 32.7ms. The accuracy achieved on video stream for real-time prediction is ~90% without any time lag or delay in identification and prediction. The rate of prediction is faster than all the previous models discussed above. The accuracy for image and video data is comparably more than models that use pattern matching algorithm, KNN algorithm, SVM algorithm that are previously discussed.

Table I. Signs and their accuracies taken from the trained model

Label	Accuracy	Label	Accuracy	Label	Accuracy
అ	96	ఙ	87	మ	92
ఆ	89	చ	89	య	87
ఇ	80	ఛ	84	ర	85
ఈ	90	జ	79	ల	94
ఉ	88	ఝ	87	వ	87
ఊ	89	ఞ	99	ళ	84
ఋ	98	ట	98	శ	90
ౠ	81	ఠ	90	ష	93

ఎ	86	డ	95	స	97
ఏ	97	ఢ	89	హ	99
ఐ	90	ణ	88	ళ	85
ఒ	89	త	92	క్ష	89
ఓ	96	థ	99	ఆకలి	99
ఔ	90	ద	90	కాదు	99
అం	75	ధ	82	వినటం	97
అః	89	న	91	సహాయం	86
క	90	ప	89	అవును	90
ఖ	96	ఫ	90	మాట్లాడడం	96
గ	92	బ	87	నువ్వు	91
ఘ	90	భ	88	ప్రమాదం	98

V. CONCLUSION

Interaction of human beings in sign language with the devices or other people was complicated or slow needing additional human help by translation or static devices with hardware equipment.However, in recent times, smart phones are capable of capturing the pictures when the user shows their palm or any other gestures and it responds.This system integrated with other video communication applications like Zoom, can help capture captions for the signs made by deaf and dumb, to communicate with ease in the meetings and conferences.This way an all inclusive community bridged by technology comes into existence preserving cultural heritage of languages which can be extended to langugaes across the world.Based on the good accuracy of the results it can be applied in real life applications easily and reliably.

REFERENCES

[1] Shivashankara S, Srinath S, American Sign Language Recognition System: An Optimal Approach, Article in International Journal of Image, Graphics and Signal Processing, August 2018.

[2] Victoria A. Adewale, et al. Conversion of Sign Language to Text And Speech Using Machine Learning Techniques, Research Article Journal of Research and Review in Science, 58-65 Volume 5, December 2018.

[3] Shaik Khadar Sharif, et al. Sign Language Recognition, International Journal of Engineering Research & Technology (IJERT), Vol. 9 Issue 05, May-2020.

[4] Sawant Pramada, et al. Intelligent Sign Language Recognition Using Image Processing, Article in IOSR Journal of Engineering, February 2013.

[5] Casam Njagi NYAGA, Ruth Diko WARIO, Sign Language Gesture Recognition through Computer Vision, Article in IST-Africa Conference Proceedings, 2018.

[6] Yuancheng Ye, et al. Recognizing American Sign Language Gestures from within Continuous Videos, CVF Computer Vision and Pattern Recognition Workshops, 2018.

[7] Paulo Trigueiros, et al. Vision-based Portuguese Sign Language Recognition System, Conference Paper in World Conference on Information Systems and Technologies (WorldCIST'14), Funchal, Madeira, April 2014.

[8] Mahesh Kumar N, Conversion of Sign Language into Text, International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 9, 2018.

[9] Alhussain Akoum, Nour Al Mawla, Hand Gesture Recognition Approach for ASL Language Using Hand Extraction Algorithm, Article in Journal of Software Engineering and Applications · January 2015.

[10] Josep Quer1, et al. Handling Sign Language Data: The Impact of Modality, Pyschology jounela frontier, 2019.

[11] J.Rekha, et al. Hand Gesture Recognition for Sign Language: A New Hybrid Approach, International Conference on Image Processing, Computer Vision and Pattern Recognition, IPCV'11, 2011.

[12] Uttam Prabhu Dessai, et al. Sign Language Gesture Recognition System for Hearing Impaired People, International Journal of Computer Applications (0975 – 8887), Volume 179 – No.45, May 2018.

[13] Shailesh bachani, et al. Sign Language Recognition Using Neural Network, International Research Journal of Engineering and Technology (IRJET), Volume: 07 Issue: 04, Apr 2020.

[14] Omkar Vedak, et al. Sign Language Interpreter using Image Processing and Machine Learning, International Research Journal of Engineering and Technology (IRJET), Volume: 06 Issue: 04, Apr 2019.

[15] Eleni Efthimiou, et al. Sign Language Recognition, Generation, and Modelling: A Research Effort with Applications in Deaf Communication, Addressing Diversity, 5th International Conference, UAHCI 2009, Held as Part of HCI International 2009, San Diego, 2009.

[16] Hanine El Hayek, et al. Sign to Letter Translator System using a Hand Glove, Conference Paper in Third International Conference on e-Technologies and Networks for Development (ICeND), April 2014.

[17] Pooja Sonawane, Anitha Nikalje, Text to Sign Language Conversion by Using Python and Database of Images and Videos, IJERECE, 2018.

[18] Yellapu Madhuri, et al. Vision-Based Sign Language Translation Device, International Conference on Information Communication and Embedded Systems (ICICES), 2013.

[19] Steve Daniels, et al. Indonesian Sign Language Recognition using YOLO Method, International Conference on Information Technology and Digital Applications (ICITDA), 2020.