INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Impact Factor: 7.542

American Sign Language Object Detection with Image and Voice

Pooja Pawar[1], Bharati Dhone[1], Priyanka Mane [1], Pooja More[1], Aashish Tulshi[2]

B.Tech Student, Department of Computer Science Engineering, JSPM's Bhagwant Institute of Technology,

Barshi India[1]

Professor Department of Computer Science Engineering, JSPM's Bhagwant Institute of Technology, Barshi, India[2]

ABSTRACT:Sign language has always been a major tool for communication among people with disabilities. In this paper, a sign language fingerspelling alphabet identification system would be developed by using image processing technique, supervised machine learning and deep learning. People use this technology to interact with computers or electronic devices without touching them.In particular, 24 alphabetical symbols are presented by several combinations of static gestures (excluding 2 motion gestures J and Z). The proposed system is implemented using the YOLOv5l model. It is a robust object detection framework. This system requires only a camera and a computer for detection.Computer vision algorithms or cameras are used to convert these sign languages. This is one of the primitive connectors between man and machine. This makes it one of the important applications of Graphic User Interface (GUI). The process involves detection of Hand portion and then finding out the gestures through sign language made by hand. Major application of this technology is in the automotive sector.

KEY WORDS: YOLO v5 model ,Computervision,sign language recognition, alphabet recognition, Graphic User Interface, automotive, detection of voice.

Python Version: 3.8
Packages: pandas, numpy, matplotlib, sklearn, opencv, ast, albumentations, tqdm, torch, IPython, PIL, shutil

I. INTRODUCTION

Communication is essential in building a nation.Good communication leads to better understanding, and it encompasses all the members of the community, including the deaf. In the Philippines, 1.23% of the entire population is either deaf, mute or hearing impaired [1]. Sign language is a complex combination of hand movements, body postures and facial expression; which is used to convey meaning among hearing and speech impaired people, as well as between people with disabilities and normal people. It is then realized that the design of a fast and highly accurate interpreting system plays a vital role, as it can facilitate the communication process for those in need, and further integrate them into the common social world. Through the capturing and recognition of hand gestures, it is possible to achieve deep analysis of human behavior, and further responds accordingly. Only by the implementation of a concrete sign language identification system this strenuous problem can be resolved.Sign language
bridges the gap of communication with other people. However, most hearing people do not understand sign language and learning it is not an easy process. As a result, there is still an undeniable barrier between the hearing impaired and hearing majority. Over the past few decades, many efforts have been made in creating a sign language recognition (SLR) system

Generally, hand gesture recognition is based on the following 2 approaches:

1.vision based approach

2.non-vision based approach.

In recent years, researches were conducted to develop hand gesture recognition systems have been more concentrated on vision based methods as they produce much less, to no restraint on the users, as compared to the previous non-vision based procedures which employ special gloves or sensors to directly collect data from hands. The first step in hand gesture recognition with vision-based approach is preprocessing, in which skin segmentation

is fundamental. The authors in .combined the skin-color model (RGB to YCbCr) with eight neighborhood method to segment out the hand region and erase noise. The skin color range can be adaptively changed based on different users, if motion detection and Gaussian Mixture Model were involved .

II. PREPROCCESSING

Labeling the images: Manual bounding box labels were created on the original images using the labelImgsoftware.Each of the pictures and bounding box coordinates were then passed through an albumentations pipeline that resized the images to 1024 x 1024 pixel squares and added probabilities of different transformations.These transformations included specified degrees of rotations, shifts in the image locations, blurs, horizontal flips, random erase, and a variety of other color transformations.
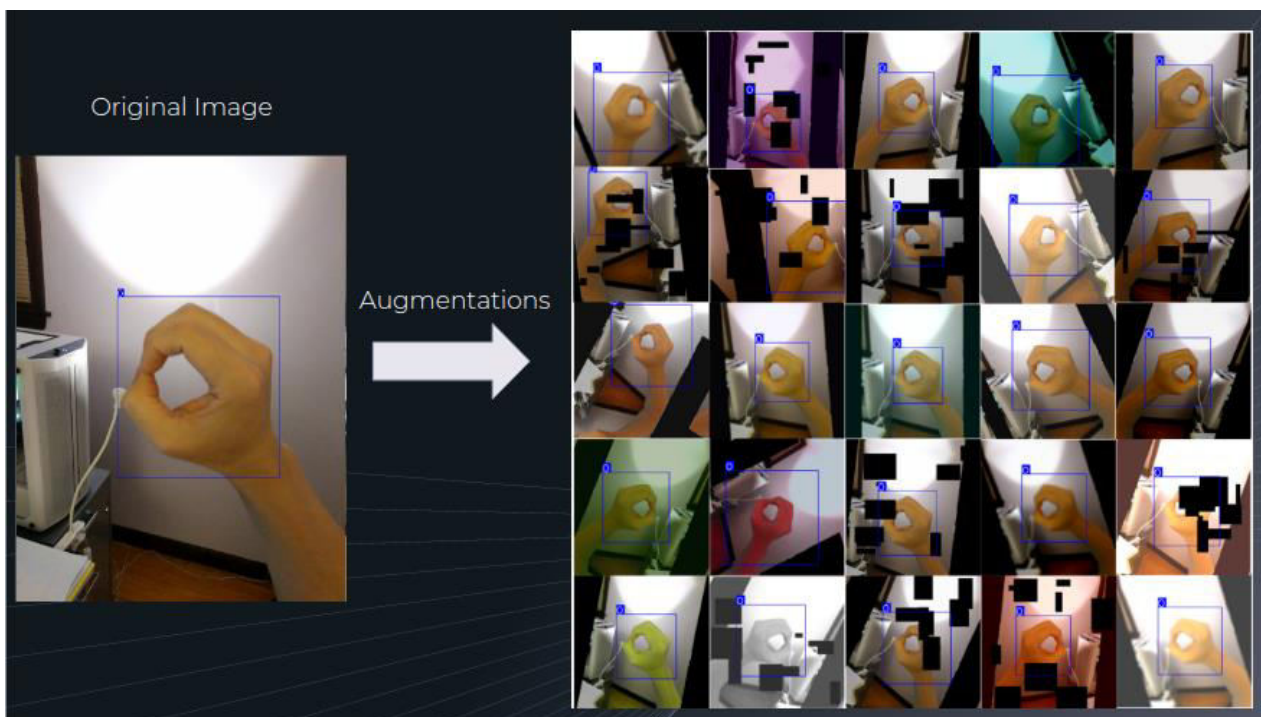

Fig.Labeling the images

III. MODELING

Backbone

Dense Block &DenseNet : To improve accuracy, we can design a deeper network to extend the receptive field and to increase model complexity. And to ease the training difficulty, skip-connections can be applied. We can expand this concept further with highly interconnected layers.A**Dense Block** contains multiple convolution layers with each layer *Hi* composed of batch normalization, ReLU, and followed by convolution. Instead of using the output of the last layer only, *Hi* takes the output of all previous layers as well as the original as its input. i.e. x_0, x_1, …, and x_{i-1}. Each *Hi* below outputs four feature maps. Therefore, at each layer, the number of feature maps is increased by four — the growth rate.
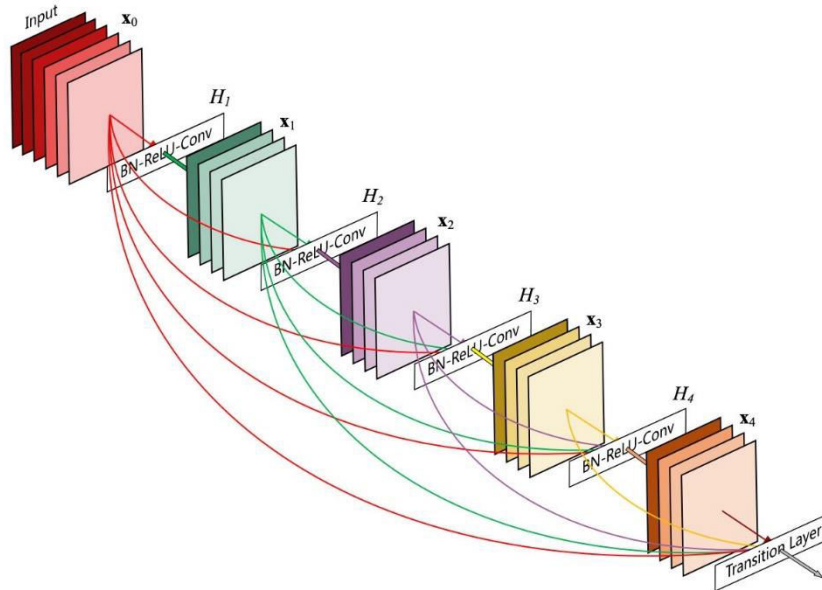
Fig.Dense Block

Then a DenseNet can be formed by composing multiple Dense Block with a transition layer in between that composed of convolution and pooling.
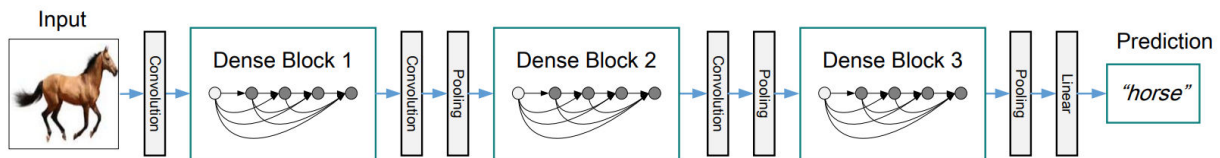

Fig.DenseNet

Neck

Object detectors composed of a backbone in feature extraction and a head (the rightmost block below) for object detection. And to detect objects at different scales, a hierarchy structure is produced with the head probing feature maps at different spatial resolutions. To enrich the information that feeds into the head, neighboring feature maps coming from the bottom-up stream and the top-down stream are added together element-wise or concatenated before feeding into the head. Therefore, the head's input will contain spatial rich information from the bottom-up stream and the semantic rich information from the top-down stream. This part of the system is called a neck. Let's get more details in its design.
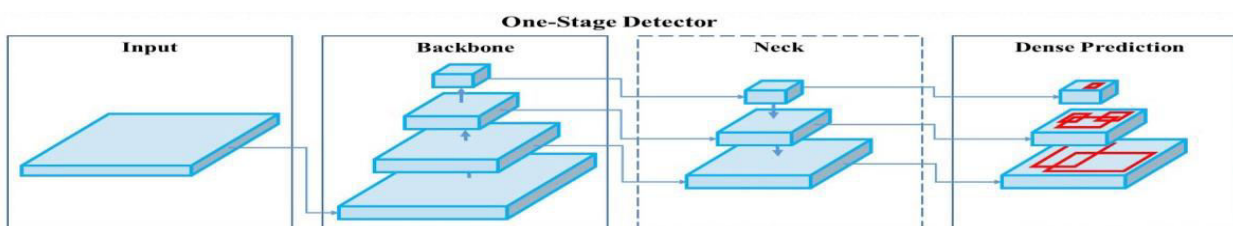

Fig.One-Stage Director

In model We can capture image from camera and image like hand gesture and sign by hand. Then this captured image is matched with database where we are already stored in memory of that model while training that model. While creating model we have do Train and Test process on that images and according that model can understand pattern in images and able to produce correct output .After checking correct pattern with captured image that alphabet in form of image will be displayed on output with voice .And the whole process is displayed as below:
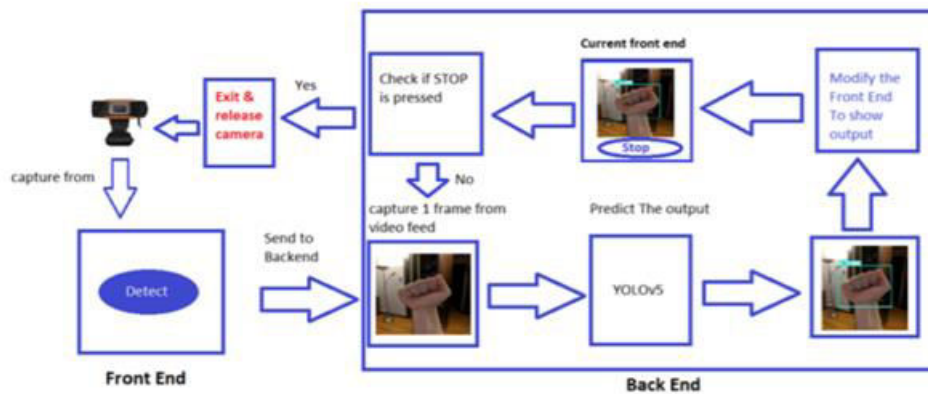

Fig. Front End and Back End

IV. MODULES

OpenCV

OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products. Being a BSD-licensed product, OpenCV makes it easy for businesses to utilize and modify the code.The library has more than 2500 optimized algorithms, which includes a comprehensive set of both classic and state-of-the-art computer vision and machine learning algorithms. These algorithms can be used to detect and recognize faces, identify objects, classify human actions in videos, track camera movements, track moving objects, extract 3D models of objects, produce 3D point clouds from stereo cameras, stitch images together to produce a high resolution image of an entire scene, find similar images from an image database, remove red eyes from images taken using flash, follow eye movements, recognize scenery and establish markers to overlay it with augmented reality, etc.

Numpy

NumPy is an open source project aiming to enable numerical computing with Python. It was created in 2005, building on the early work of the Numerical and Numarray libraries. NumPy will always be 100% open source software, free for all to use

Pyttsx3

It is a text-to-speech conversion library in Python. Unlike alternative libraries, it works offline, and is compatible with both Python 2 and 3.

Advantages

1) Functioning:

1.1) The output of the sign language will be displayed in the text form in real time.

1.2) The communication with hearing and speech impaired people become easier.

1.3) The images captured from a continuous video stream of a webcam are passed in a yolo model and the output is text. Thus, this feature of the system makes communication very simple and delay free.

2) No external hardware required:

2.1) As traditional approach is using sensors or gloves, the implemented system suggests the direct use of images from a web camera for recognition which reduces the cost

2.2) The only requirement here is a webcam, a computer (for dedicated application) or a server (for a web application which is to be deployed on a server).

3) Portable:

3.1) If the Entire project is implemented on a Raspberry Pi computer, the entire system becomes portable and can be taken anywhere.

3.2) Deploying the project on a web application will increase its access without having to setup manually.

3.3) Deploying this project as a dedicated Desktop application will enable the model to use a dedicated GPU on it which in turn will increase its performance.

4) Does not get damaged through use:

4.1) As no specific hardware is required here, and the project is just a software, there is no degrade in performance with respect to time.

4.2) The only variable factors for the proposed system are that it depends on how powerful the computer it is deployed on is.

Disadvantages

1. Room must have proper light as our project uses hardware like camera for providing input .If proper light will not be there then output may be not possible as expected.

2. For correct result palm must be faced towards camera so that we can get correct output.

3. The performance of the system also depends on the type of hardware used. Higher end systems can yield higher accuracy as well as high frames per second.

V. CONCLUSION

We are successfully implemented this model using YOLO v5 for deaf people so that they can easily communicate with other people without hesitation. Also whatever letter is getting recognized will be translated into voice also. That is why people can easily understand by voice .The traditional approach seems to be quite expensive and a complex

hardware system is required. This also demands high maintenance. On the other hand, the proposed system focuses on recognizing gestures through the camera, thus reducing hardware requirement, complexity and is cost effective. Both techniques mentioned above, however, have their advantages and disadvantages and may perform well in some challenges while being inferior in others.

REFERENCES

1. Ira Cohen NicuSebeAshutoshGarg Lawrence S Chen and Thomas S. Huang "Facial expression recognition from video sequences: temporal and static modeling" Computer Vision and Image Undertaking pp. 91 February 2003.
2. Bernard Boulay Francois Bremond and Monique Thonat "Human Posture Recognition in Video Sequence" IEEE International Workshop on VSPETS Visual Surveillance and Performance Evaluation of Tracking and Surveillance 2003.
3. AnupNandy Jay Shankar Prasad SoumikMondalPavanChakraborty and G. C. Nandi "Recognition of Isolated Indian Sign Language Gesture in Real Time" in Communications in Computer and Information Science book series CCIS vol. 70.
4. NehaBaranwal and G. C. Nandi "Continuous dynamic Indian Sign Language gesture recognition with invariant backgrounds by KumudTripathi" 2015 Conference on Advances in Computing Communications and Informatics (ICACCI).
5. Carol NeidleAshwinThangali and Stan Sclaroff "Challenges in Development of the American Sign Language Lexicon Video Dataset (ASLLVD) Corpus" 5th Workshop on the Representation and Processing of Sign Languages: Interactions between Corpus and Lexicon LREC 2012 2012.
6. Szegedy et al. "Going Deeper with Convolutions" CVPR 2015.
7. SeppHochreiter et al. "Long Short-Term Memory" Neural Computation vol. 9 no. 8 pp. 1735-1780 1997.
8. Diederik P. Kingma and Jimmy Ba "Adam: A Method for Stochastic Optimization" Published as a conference paper at the 3rd International Conference for Learning Representations 2015.
9. Geoffrey Hindon et al. "Dynamic Routing Between Capsules" Nov 2017.
10. Shuai Li Wanqing Li Chris Cook Ce Zhu and YanboGao "Independently Recurrent Neural Network (IndRNN): Building A Longer and Deeper RNN" CVPR 2018.
11. C. Chuan E. Regina and C. Guardino "American Sign Language Recognition Using Leap Motion Sensor" Proc. 13th International Conference on Machine Learning and Applications (ICMLA) pp. 541-544 2014.
12. A. Joshi H. Sierra and E. Arzuaga "American sign language translation using edge detection and cross correlation" Proc. IEEE Colombian Conference on Communications and Computing (COLCOM) pp. 1-6 2017.

Resources

1. Cudnn install guide: https://docs.nvidia.com/deeplearning/cudnn/install-guide/index.html
2. Install Opencv: https://www.codegrepper.com/codeexamples/python/how+to+install+opencv+in+python+3.8
3. Roboflow augmentation process: https://docs.roboflow.com/image-transformations/image-augmentation
4. Heavily utilized research paper on image augmentations: https://journalofbigdata.springeropen.com/articles/10.1186/s40537-019-0197-0#Sec3
5. Pillow library: https://pillow.readthedocs.io/en/latest/handbook/index.html
6. Labeling Software labelImg: https://github.com/tzutalin/labelImg
7. Albumentations library https://github.com/albumentations-team/albumentations

INNO SPACE
SJIF Scientific Journal Impact Factor
Impact Factor: 7.542

doi® crossref

ISSN
INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA

NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

📱 9940 572 462   6381 907 438   ✉ ijircce@gmail.com

Scan to save the contact details