International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 05 | May 2020 www.irjet.net p-ISSN: 2395-0072

Gesture Controlled Media Player using TinyYoloV3

Abhilash Dayanandan[1], Akshay Chakkungal[2], Anooj Kommeri[3], Deepak Koppuliparambil[4],

Dr. Prashant Nitnaware[5]

[1,2,3,4]*Department of Computer Engineering, Pillai College of Engineering, Navi-Mumbai, Maharashtra, India-410206*
[5]*Professor, Department of Computer Engineering, Pillai College of Engineering, Navi-Mumbai, Maharashtra, India-410206*

Abstract - *Today we are in a generation where everyone is dependent to perform most of their tasks using computers, thus making it an integral part of our life. Though computers have been with us through several decades, still we follow the same, old, primitive methods such as a mouse, keyboard, etc. to interact with them. Also, there is a wide range of health problems that affect many people, caused by the constant and continuous work with the computer. This is where Human-Computer Interaction (HCI) provides us a solution that focuses on the interfaces between users and computers by researching the design and use of computer technology, which involves using human gestures which constitute a space of motion expressed by the body, face, and/or hands. Among a variety of gestures, hand gestures are the most expressive and the most frequently used. The focus of creating hand gestures is to create better communication between humans and computers for conveying information. The proposed project aims to create a gesture-controlled media player wherein we can use our hands and control the video played on the computer. Rather than using simple image processing and machine learning, with the help of deep learning and neural networks several combinations of hand gestures can be recognized with increased accuracy by using the YOLO object detection model.*

Key Words: **Human Computer Interaction (HCI), Human gestures, Hand gestures, Media player, Deep learning, Neural networks, YOLO, Object Detection**

1. INTRODUCTION

The development in natural interaction between humans and computers has had significant growth over the past few decades. Although hand gesture recognition has been one of the most attractive in the field in human-computer interaction (HCI), which can be found in virtual reality, games, robotics, and automated homes. Here, the gesture recognition tasks need to be carried out in real-time with limited computational resources (such as GPU, memory, etc) as most of the applications are usually built on the embedded platforms. Difficulties such as non-rigid and semantic similarity of the gestures, complexity of the background, and the variability of the illumination, etc. which needs to be overcome by the gesture recognition system by locating correct gesture regions and classifying them. To control a media player with gestures, the recognition time, as well as accuracy, is a crucial aspect that needs to be taken under consideration to evaluate the performance of the system. This can be achieved by using YOLO, an object detection model which unlike any deep learning algorithm is a single-stage detector resulting in fast and accurate detection, especially for embedded systems. YOLO is a unified model for object detection developed by Joseph Redmon and Ali Farhad. The model is simple to construct and can be trained directly on full images. Unlike classifier-based approaches, the detection performance of YOLO is based on a loss function and the entire model is trained together. Since YOLO integrates well into new domains, this makes it ideal for applications that rely on fast, robust object detection.[1]

2. RELATED WORK

2.1 Image Processing Techniques for Hand Gesture and Sign Recognition:

Divyashree B A and Manjushree K describe the use of image processing techniques to extract the features from the images. The extracted features are sent to a machine learning algorithm that recognizes the hand signs accurately. The datasets are in the form of static or dynamic format. The usage of hand gloves also results in calculating the corner tip of gestures wrongly.[12] The system gives better accuracy only when the images are static compared to the motion gestures.

2.2. YOLO-LITE: A Real-Time Object Detection Algorithm Optimized:

Real-time object detection model is developed to run on portable devices such as a laptop or cellphone lacking a Graphics Processing Unit (GPU). YOLOLITE was designed to create a smaller, faster, and more efficient model increasing the accessibility of real-time object detection to a variety of devices.[13] We will try to use yolo lite for accuracy and speed. YOLO-LITE can be more accurate in detecting the flow of people as some interference, such as people in the billboards and unrelated backs, can be ignored because of the boundary selection. YOLO-Lite has only 7 layers and still is faster than most other models using this speed we can process more frames and give a better fps output video which will greatly increase the accuracy.[13]

2.3 YOLOv3: An Incremental Improvement:

In YOLOv3 single neural network is applied to the full image. This neural network divides the image into regions. The divide regions are then used to predict bounding boxes according to their probabilities for each region. The weight of the bounding boxes are given by the probabilities. In mAP measured at 0.5 IOU YOLOv3 is on par with Focal Loss but about 4x faster. Moreover, you can easily tradeoff between speed and accuracy simply by changing the size of the model, no retraining required.[14] YOLOv3 is extremely fast and accurate. This algorithm will be used for gesture recognition and for training the model by bringing a balance between speed and accuracy.

2.4 Understanding of Object Detection Based on CNN Family and YOLO:

You Only Look Once (YOLO), which breaks through the CNN family's tradition and innovates a complete new way of solving the object detection with the most simple and highly efficient way. YOLOv2 has achieved 76.8 mAP at 67 FPS and 78.6 mAP at 40 FPS. It imposes strong spatial constraints on bounding box predictions such as recognizing small objects in groups. It still struggles to generalize to objects in new or unusual aspect ratios or configurations.[15] It's also not perfect that YOLO's loss function treats errors the same in small bounding boxes vs large boxes.

2.5 A New Robust Approach for Real-Time Hand Detection and Gesture Recognition:

A new robust approach in gesture recognition based on skin color detection is used. Use of three different color correction algorithms is done before skin detection and then they are classified into gestures. It tracks the hand(s) location in real-time and recognizes several gestures. Only a webcam is required. It can differentiate if the user is performing one or two hand gestures. Cannot detect grab or swipe gestures. Problems when there is motion blur or hand trembling.[16] These issues are solved by making a model and training it upto a certain accuracy.

3. PROPOSED WORK

In our proposed system, we are taking into consideration the need for improvement of hand gesture recognition accuracy as well as keeping the processing time as low as possible while implementing Machine Learning and Deep learning algorithms

3.1 System Architecture

The system architecture is given in Figure 1. Each block present is described in this section. In our proposed system, we are taking into consideration the need for improvement of hand gesture recognition accuracy as well as keeping the processing time as low as possible while implementing Machine Learning and Deep learning algorithms.

A. Webcam: When the media player opens, the webcam automatically starts to function and captures live video which then is read by OpenCV frame by frame and fed into the YOLO neural network for object detection.

B. Image preprocessing: The proposed architecture begins with the user performing some hand gestures facing the camera. The camera captures live video which is then divided into multiple image frames and are then pre-processed using the following operations:

- Down Sampling: It is the process of reducing the spatial resolution of the image while maintaining its 2D representation. It is used in the reduction of storage and transmission requirements of images.[3]
- Mean Subtraction: Mean subtraction is used to prevent illumination changes in the input images in our dataset. It is used to aid our Convolutional Neural Network.[4]
- Scaling: After we perform mean subtraction we scale our images by some factor. This value is set to 1 by default(i.e. no scaling) but we can give another value as well. The scale factor should be $1/\sigma$ since we are multiplying the input channels by a scale factor.[4]
- Channel Swapping : OpenCV assumes images are in BGR channel order; however, the mean value assumes we are using RGB order. To resolve this discrepancy we can swap the R and B channels in image by setting swapRB value to True.[4]

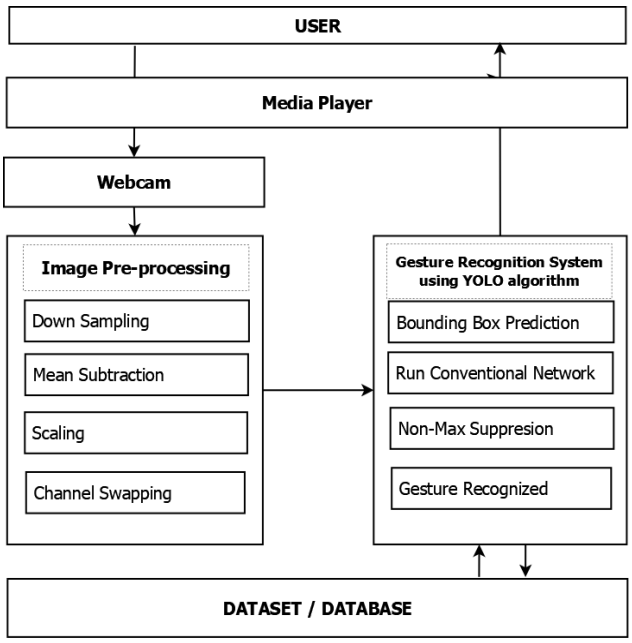

Fig-1: Proposed system architecture

C. Gesture Recognition System using yolo algorithm:

After image pre-processing, the output is fed into a gesture recognizing system that uses the YOLO algorithm. The YOLO algorithm recognizes the gesture in the following phases:

- Bounding Box Prediction: YOLO divides the image into a grid of 13 by 13 cells: Each cell approximately estimates 5 bounding boxes. A bounding box describes the rectangle that encloses an object. YOLO also gives a confidence score as output that tells us about the certainty of the predicted bounding box of actually enclosing some object.[5]
- Non-Max Suppression: In this process, we're discarding bounding boxes with low object probability and with the highest IOU(Intersection over union).
- Gesture Recognition: It gives class labels to each gesture by comparing the corresponding gesture in the dataset/database.

D. Database/Dataset: The dataset has been created using the following steps:

- Collect the raw data: Before training the model we need data/images. This dataset contains gestures performed by 4 different people, each performing 5 different gestures, for a total of 1500 images. The 5 different gestures used are thumbs up and down for increasing and decreasing the volume, right and left point to seek forward and seek backward and palm to pause and play the media player

Fig-2:. Raw input images present in the dataset

- Label the data: Once we gather the data, the next step is to label/annotate them. We used the labelImg tool for labeling the images since it lets us save the annotations directly into YOLO format. This is illustrated in fig.3.
- Split the data : 90% of the images in Dataset are used for training the model and 10% of images are used for testing the model.

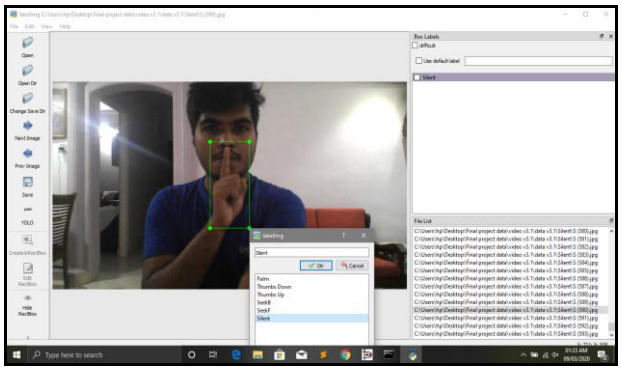

Fig-3: Labelling of image using LabelImg

4. DATA ACQUISITION

In our project we have captured 6 different gestures by the webcam of our laptops. The dataset contains gestures performed by 4 different people, each performing 6 different gestures, for a total of 1500 images. Six video samples were gathered from each member of our group for each gesture. 90% of the images in the dataset are used for training the model and 10% of images are used for testing the model.

5. EXPERIMENTAL RESULTS

The system was tested in real time and the results we have achieved are promising. Our system was tested several times for different gestures with different persons. Table 1 represents the accuracy achieved in real time for each gesture. Figure 3 displays the accuracy mentioned in the Table 1 for each gesture along with the bounding box. The graph for average loss function over the iterations is being shown in Chart 1. Also Figure 5 shows the user interface for the system which we named as Eidolon lens.

Table -1: Hand Gesture Recognition Results

Gestures	Recognition Rate(%)
Play / Pause	99.9741
Volume Down	93.7529
Volume Up	91.1508
Seek Forward	72.4973
Seek Backward	98.7789
Mute	99.2391

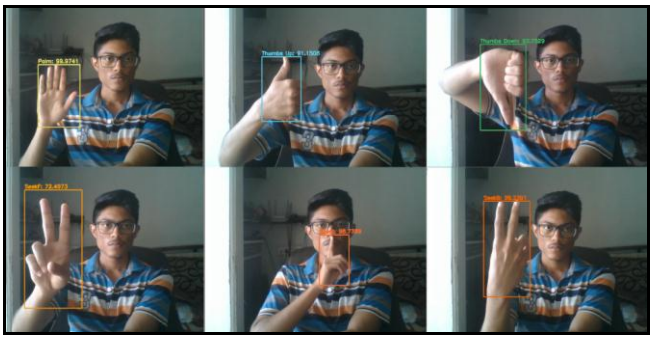

Fig-4: Accuracy achieved of the trained model for each gesture

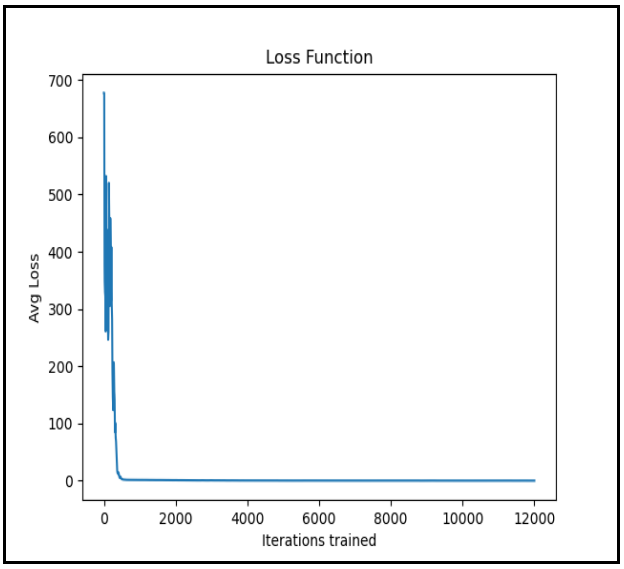

Chart -1: Average loss function graph

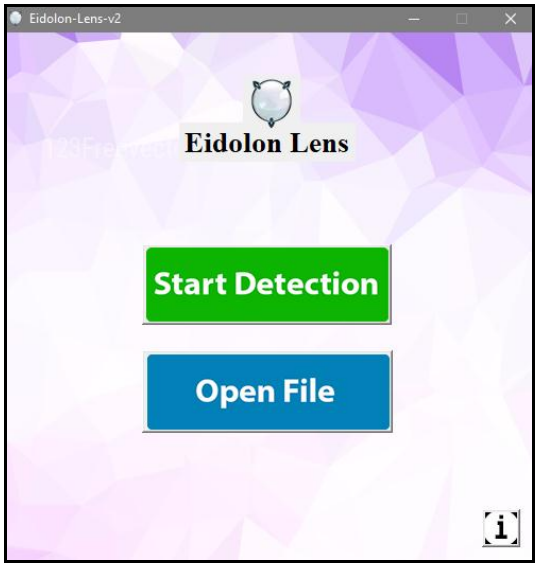

Fig-5: Interface of Eidolon lens

Following images show the results obtained after using different gestures on VLC media player.

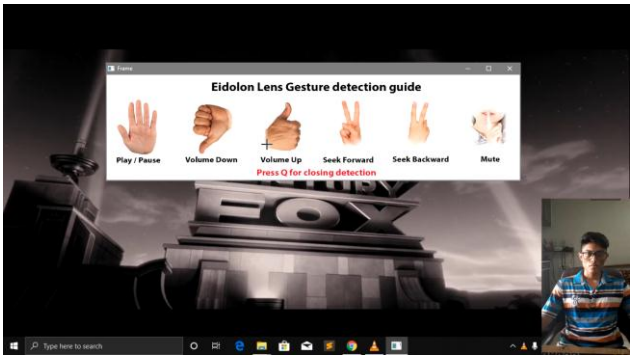

Fig-6: Starting interaction with Eidolon Lens

Fig-7: Pause video when palm gesture detected

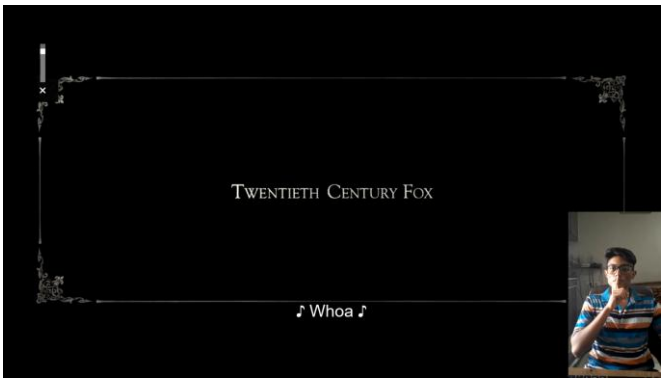

Fig-8: Mute video when silent gesture detected

Fig-9: Increase in volume of video for thumbs up gesture

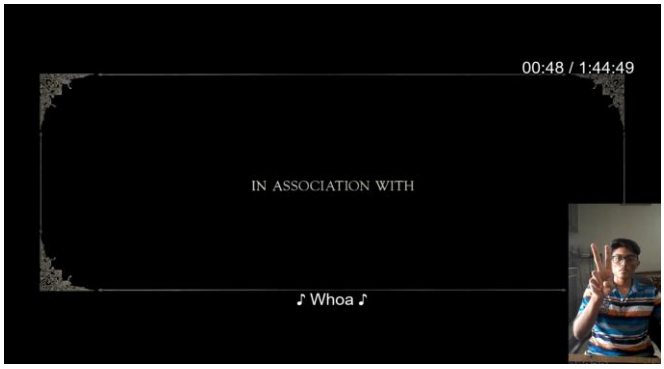

Fig-10: Seek Forward in video for Forward-V gesture

5. METRICS

The metric used to evaluate object detection and classification models is called Mean Average Precision. To calculate the mAP for a set of detections, the interpolated average precision is calculated for each class, and the mean is calculated over it. The PR curve is created by clubbing each detection to its most overlapping object instance. Detections whose IOU with the truth value above the threshold are considered as True Positives while the others are said to be False Positives Next, two metrics called Precision and Recall are calculated as follows.

$$Precision = \frac{TP}{TP + FP} \quad (5.1)$$

$$Recall = \frac{TP}{\text{No. of Ground Truth Boxes}} \quad (5.2)$$

The IOU or Intersection over Union of predicted bounding box Bp and ground truth bounding box B_{gt} is defined as follows.[7]

$$IOU = \frac{\text{Area of overlap of } B_p \text{ and } B_{gt}}{\text{Area of union of } B_p \text{ and } B_{gt}} \quad (5.3)$$

The average precision(AP) is computed by averaging the precision values on the Precision Recall curve where recall is in the range [0, 0.1 ... 1].[7]

$$AP = \frac{1}{11} \sum_{r \epsilon \{0, 0.1, .., 1\}} p_{\text{interp}}(r) \quad (5.4)$$

The precision at each recall level r is interpolated by determining the maximum precision measured for a method for which the corresponding recall value exceeds r.[7]

$$p_{inter}(r) = max_{p:p \geq r}\ \mathrm{p}(\tilde{r}) \quad (5.5)$$

Now the mAP is calculated as follows.[7]

$$mAP = \frac{\sum_{\text{i } \epsilon \text{ classes}} AP_i}{\text{Total no. of classes}} \quad (5.6)$$

6. CONCLUSION

The proposed hand-gesture recognition system provides better results in complex backgrounds compared to the methods present in the literature. The selection of simple and easily distinguishable hand-gestures using the YOLO network has helped in making the proposed system run in real-time with high recognition accuracy which is a crucial requirement to control a media player. Our system only uses a webcam that would lead to a new era of HCI where no physical contact with the device is required.

ACKNOWLEDGMENT

We would also like to expand our deepest gratitude to our Project Guide, Dr. Prashant Nitnaware who guided us by providing us with his valuable suggestions in numerous consultations on this project which gave us the inspiration to improve our assignment. We would also like to express our heartfelt thanks to our Head Of Department, Dr. Sharvari Govilkar and our Principal, Dr. Sandeep Joshi for providing us with a platform where we can try to work on developing projects and demonstrate the practical applications of our academic curriculum.

REFERENCES

[1] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi "You Only Look Once: Unified" University of Washington, Allen Institute for AI , Facebook AI Research [2016] [Online] Available: https://www.cvfoundation.org/openaccess/content_cvpr_2016/papers/Redmon_You_Only_Look_CVPR_2016_paper.pdf

[2] Manuj Paliwal, Gaurav Sharma "A Dynamic hand gesture recognition system for controlling VLC media player" [January2013] [Online] Available: https://www.researchgate.net/publication/261489166_A_dynamic_hand_gesture_recognition_system_for_controlling_VLC_media_player

[3] Abdou Youssef "Image Downsampling and Upsampling Methods" The George Washington University [Online] Available: https://www2.seas.gwu.edu/~ayoussef/papers/ImageDownUpSampling-CISST99.pdf

[4] Adrian Rosebrock "Deep learning: How OpenCV's blobFromImage works"[November 6,2017] [Online] Available:https://www.pyimagesearch.com/2017/11/06/deep-learning-opencvs-blobfromimage-works/

[5] Nishan Pantha "Understanding Object Detection Using YOLO" [April 25,2019] [Online] Available: https://dzone.com/articles/un derstanding-object-detection-using-yolo
[6] Michal Maj "What is object detection ? Introduction to YOLO algorithm" [August,22,2018] [Online] Available: https://appsilon.com/object-detection-yolo-algorithm/?nabe=4634331497365504:0&utm_referrer=https://www.google.com/
[7] M M Vikram "A YOLO Based Approach for Traffic Sign Detection" National Institute of Technology Karnataka [April 2, 2018] [Online] Available: https://Vikram-mm.github.io/yolo_report.pdf
[8] "Gesture Recognition - an Overview by Science Direct" [Online]. Available: https://www.sciencedirect.com/topics/computer-science/gesture-recognition
[9] Kamakshi S "5 Useful Applications Of Gesture Technology"[July 31, 2013] [Online]Available: http://www.techtree.com/content/features/4254/5-useful-applications-gesture-technology.html
[10] M.K. Bhuyan, D. Ghoah, P.K. Bora "A Framework for Hand Gesture Recognition with Applications to Sign Language" [September 15 2006] [Online] Available: https://ieeexplore.iee e.org/document/4086294
[11] Hongyi Liu,Lihui Wang "Gesture recognition for human-robot collaboration: A review" [November 2018] [Online] Available: https://www.sciencedirect.com/science/article/pii/S0169814117300690#!
[12] Divyashree B A, Manjushree K "Image Processing Techniques for Hand Gesture and Sign Recognition"[January 2019] [Online] Available: https:www.irjet.net/archives/V6/i1/IRJET-V6I1298.pdf
[13] Rachel Huang, Jonathan Pedoeem, Cuixian Chen "YOLO-LITE: A Real-Time Object Detection Algorithm Optimized for Non-GPU Computers" [November 14 2018] [Online] Available: https://arxiv.org/pdf/1811.05588.pdf
[14] Joseph Redmon, Ali Farhadi "YOLOv3: An Incremental Improvement"[April 8 2018] [Online] Available: https://arxiv.org/abs/1804.02767
[15] Juan Du "Understanding of Object Detection Based on CNN Family and YOLO" [February, 2018][Online] Available: https://iopscience.iop.org/article/10.1088/1742-6596/1004/1/012029
[16] Rayane EL Sibai, Chady Abou Jaoude, Jacques Demerjian "A New Robust Approach for Real-Time Hand Detection and Gesture Recognition" [March 2017] [Online] Available: https://ieeexplore.ieee.org/document/8079780/

BIOGRAPHIES

Mr. Abhilash Dayanandan
B.E Computer Engineering
Pillai College of Engineering
Navi-Mumbai, India

Mr. Akshay Chakkungal
B.E Computer Engineering
Pillai College of Engineering
Navi-Mumbai, India

Mr. Anooj Kommeri
B.E Computer Engineering
Pillai College of Engineering
Navi-Mumbai, India

Mr. Deepak Koppuliparambil
B.E Computer Engineering
Pillai College of Engineering
Navi-Mumbai, India

Dr. Prashant Nitnaware
Dept. of Computer Engineering
Pillai College of Engineering
Navi-Mumbai, India