International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

REAL TIME OBJECT DETECTION USING YOLO ALIGORITHM

Dr.K.Bhargavi, Professor, Department of Computer Science and Engineering, Teegala Krishna Reddy Engineering College, Hyd.
N.SaiSuthiksh, B.Tech., Department of Computer Science and Engineering, Teegala Krishna Reddy Engineering College, Hyd.
bhargavi.mtech@gmail.com,sasisuthiksh.n@gmail.com

ABSTRACT:

Object detection using deep learning has achieved very good performance but there are many problems with images in real-world shooting such as noise, blurring or rotating jitter, etc. These problems have a great impact on object detection. The main objective is to detect objects using You Only Look Once (YOLO) approach. The YOLO method has several advantages as compared to other object detection algorithms. In other algorithms like Convolutional Neural Network (CNN), Fast-Convolutional Neural Network the algorithm will not look at the image completely, but in YOLO ,the algorithm looks the image completely by predicting the bounding boxes using convolutional network and finds class probabilities for these boxes and also detects the image faster as compared to other algorithms. We have used this algorithm for detecting different types of objects and have created an android application which would return voice feedback to the user.

Key Words: YOLO , image processing, object detection, CNN, Deep Learning, Bounding boxes, Neural Network

INTRODUCTION

Object detection is one of the most important research directions for computer vision. object detection is a technique that detects the semantic objects of a particular class in digital images and videos. One of its real-time applications is self-driving cars or even an application for visually impaired that detects and notify the disabled person that some object is in front of them. Object detection algorithms can be divided into the traditional methods which used the technique of sliding window where the window of specific size moves through the entire image and

the deep learning methods that includes YOLO algorithm. In this, our aim is to detect multiple objects from an image. The most common object to detect in this application are the bus, bottle, and mobile. For locating the objects in the image, we use concepts of object localization to locate more than one object in real-time systems. There are various techniques for object detection, they can be divided into two categories, first one is the algorithms based on Classifications. CNN and RNN come under this category. In this category, we have to select the interested regions from the image and then have to classify them using Convolutional Neural Network. This method is very slow as we have to run a prediction for every selected region. The second category is the algorithms based on Regressions. YOLO method comes under this category. In this, we won't have to select the interested regions from the image. Instead here, we predict the classes and bounding boxes of the whole image at a single run of the algorithm and then detect multiple objects using a single neural network. YOLO algorithm is faster as compared to other classification algorithms. YOLO algorithm makes localization errors but it predicts less false positives in the background. These algorithms are not tested with degraded images, i.e. they are trained with academic data sets, including ImageNet, COCO and VOC, etc. but they are not well tested with randomly captured data sets. The main issues of images captured in the real scene are: 1) Due to the instability of the camera, the captured images may be blurred. 2) The images can also not be clear enough because the object can be obstructed. 3) The images may have poor quality as a result of bad weather, overexposure or low resolution. (see Fig. 1)

Fig. 1- Image problems: under exposure, blur, noise

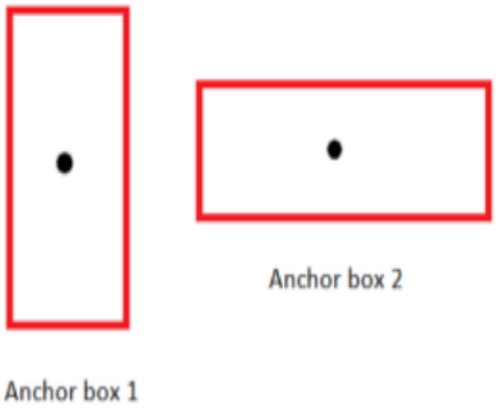


Fig. 3- Anchor boxes

The above figure shows the anchor box of the image we considered. The vertical anchor box is for the human and the

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

horizontal one is the anchor box of the car.

EXISTINGSYSTEM:

The existing system has the CNN and image processing to the real time object detection ,btu the accuracy of the object detection is less.

PROPOSEDSYSTEM

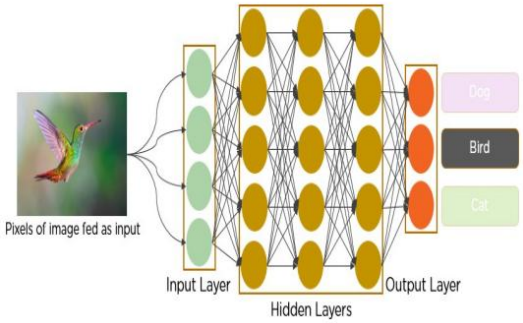
In this project using python, yolov3 (You Only Look Once v3) and OpenCV algorithms we are detecting objects from video and images. Yolov3 is a famous object detection algorithm developed by Washington university, this algorithm generate yolov3 weight model using python Deep Learning algorithm called CNN (Convolution Neural Networks). This algorithm is pre-trained with all images and assign unique class name to each unique images and then generate a model, this algorithm convert each images into layers and then for each layer extract features and add weight to the model, due to all possible features from single image another image with some related features can also be predicted. Whenever we are giving new image then that image will be applied on pre-trained weight model to get best accuracy matching image label.

ADVANTAGES:

- Accuracy is more
- Very fast

METHODLOGY

CNN: In deep learning, a **convolutional neural network** (**CNN/ConvNet**) is a class of deep neural networks, most commonly applied to analyze visual imagery. Now when we think of a neural network we think about matrix multiplications but that is not the case with ConvNet. It uses a special technique called Convolution. Now in mathematics **convolution** is a mathematical operation on two functions that produces a third function that expresses how the shape of one is modified by the other



CNN Architecture

3.1 Model Details:

The model details are as follows: The input is batch of images with shape (m,

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

608, 608, 3) •The output is a list of bounding boxes with the recognized classes. Each bounding box is denoted by 6 numbers (p_c, b_x, b_y, b_h, b_w, c). If you expand c i.e. classes we get an 80-dimensionalvector,each boundingboxisthenrepresented by85 numbers,

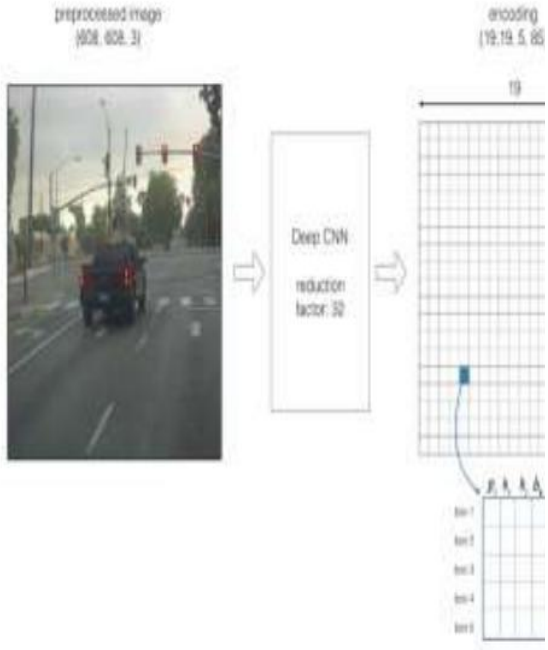


Fig. 4 -Architecture of YOLO

So, the architecture can be summarized as: IMAGE (m, 608, 608, 3) -> DEEP CNN -> ENCODING (m, 19, 19, 5, 85). If the center or the midpoint of an object falls into a grid cell, then that grid cell is responsible for detecting that object. Since in the model we are using 5 anchor boxes and each of the 19 x19 cells thus encodes information about 5 boxes. Anchor boxes are defined by their width and height. For simplicity, the image is first flattened that is the last two last dimensions of the shape (19, 19, 5, 85) encoding. so the output of the Deep CNN is in form: (19, 19, 425). Fig 3 shows the flattening.

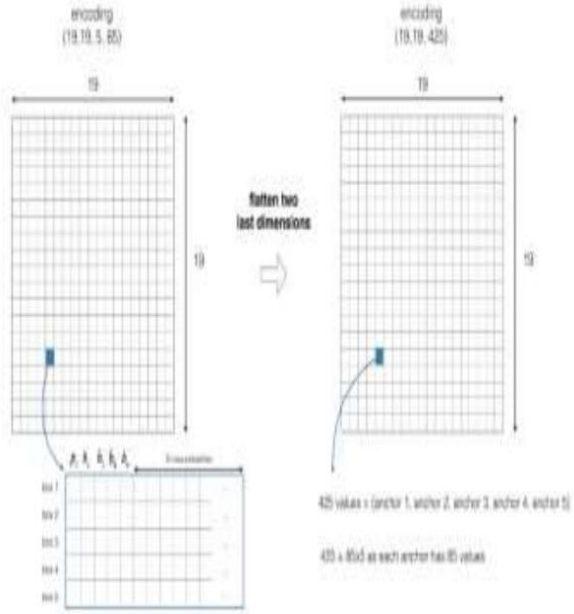


Fig. 5- Flatten last two dimensions

Now for each grid that is for each box of the cell compute the following elementwise product as well as the probability that the box contains a particular class.

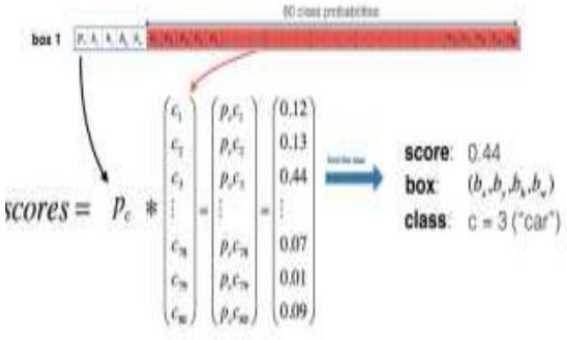


Fig. 6- Determining the probability

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

After plotting only the boxes that the algorithm had given of higher probability, there are too many boxes and hence filtering these boxes is very important for accuracy.

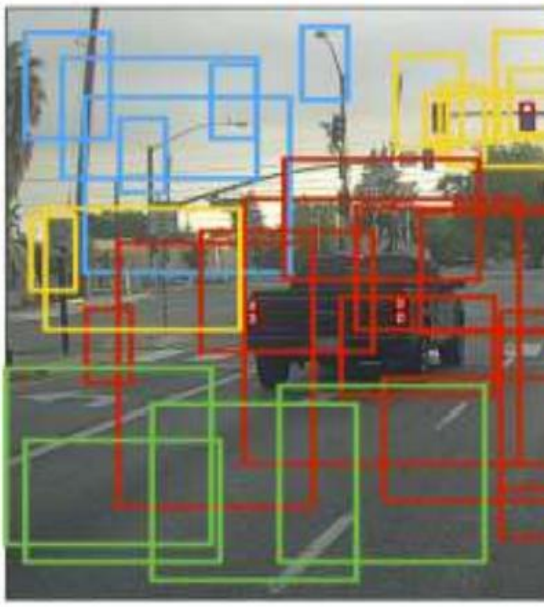


Fig. 7-Output without Filtering al

Each cell has 5 anchor boxes. So in total ifwe calculate, the model predicts: 19x19x5 = 1805 boxes. In the figure different colors denote different classes. So we filter the algorithm's output down to a less number of boxes i.e. much smaller number of detected objects. To do this we carry out two important steps:

• Get rid of boxes with a low score that is to remove the box which are not very confident about detecting a class

• Select only one box that overlaps many other boxes with each other and which detects the same object.

After the filtering based on the score of the classes, the second filter which is applied on the left boxes is the Non maximum Suppression (NMS).

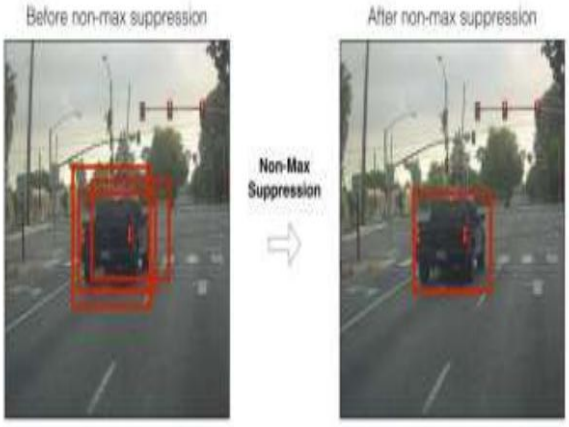


Fig. 8 -Non Max Suppression

It uses the concept of Intersection Over Union (IoU). IoU is the ratio of intersection of two boxes to the union of the boxes. This is shown in Fig 7.

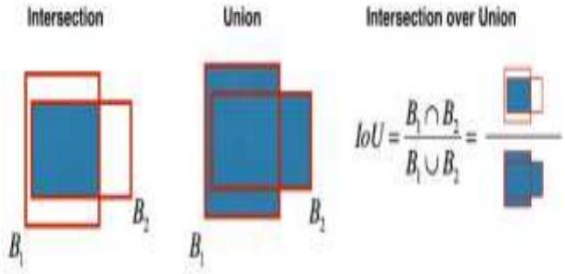


Fig. 9 -Intersection over Union

The steps in non-maximum suppression are: • Out of the left boxes select the box that has a highest score.

• Compute its overlap with all other boxes, and discard the boxes that overlap it more than IoU value.

• Go back to the step 1 and iterate until there are no more boxes with a less scores than the current selected

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal www.ijarst.in
ISSN: 2457-0362

box. This discards all boxes and only the best box remains in the last.

We have created a model that has 3 types of object that are 1.bottle, 2.car , 3.mobile.

Fig. 10- Example of 3x3 grid ir

Consider the above example, an image is taken and it is divided into 3x3 grid that is in the form of 3 x 3 matrixes. Each grid is labelled along with this each grid undergoes both image classification and objects localization techniques. The label is considered as Y. Y consists of 8 values.

y =	pc
	bx
	by
	bh
	bw
	c1
	c2
	c3

Fig. 11 -Elements of label Y

Pc – Represents whether or not an object is present in the grid or not. If present pc=1 else 0. bx, by, bh, bw – are the bounding boxes of the objects (if present).

c1, c2, c3 – are the classes. If the object is a car then c1 and c3 will be 0 and c2 will be 1. In our example image, the first grid contains no proper object. So it is represented as,

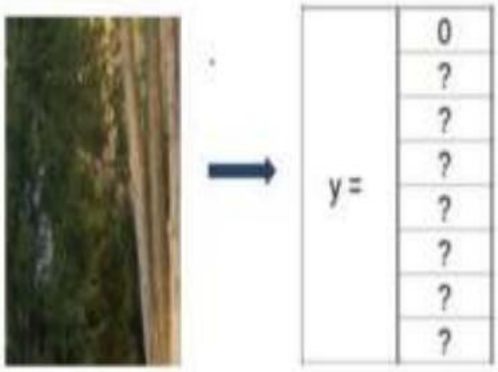


Fig. 12- Grid with no object

In this grid, there exists no proper object so the pc value is 0. Consider a grid with the presence of an object. Both 5th and 6th grid of the image contains an object. Let' consider the 6th grid, it is represented as.

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

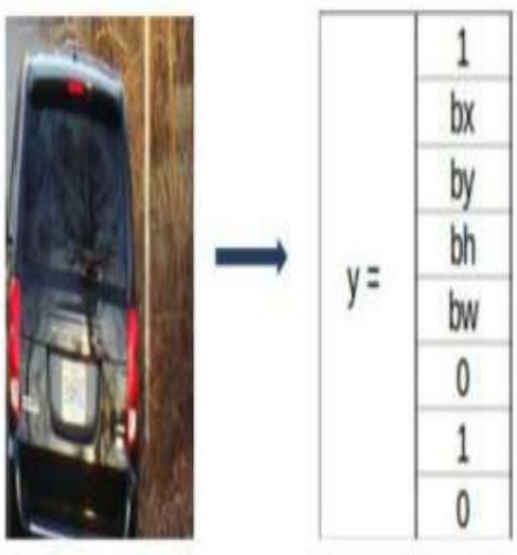


Fig. 13- Grid with object detected

The above image shows that, 1 represents the presence of an object. And bx, by, bh&bw are the bounding boxes that represent object in the 6th grid. And the object in that grid is a car so the classes are (0,1,0). The matrix that is formed in this is Y=3x3x8. If two or more grids contain the same object then the center point of the object is found and the grid which has that point is taken. For this, to get the accurate detection of the object we can use to methods. They are Intersection over Union and Non-Max Suppression. In IoU, it will takes the actual and predicted bounding box value.If the value of IoU is more than or equal to our threshold value (0.5) then it's a good prediction. The threshold value is just an assuming value. We can also take greater threshold value to increase the accuracy or for better prediction of the object. The other method is Non-max suppression, in this, the high probability boxes are taken and the boxes with high IoU are suppressed. Repeat this until a box is selected and consider that as the bounding box for that object. After getting the co-ordinates of the bounding boxes, they are drawn over the image and the voice feedback of the detected classes is provided using gTTS (Google Text-to-Speech). Along with that, whenever an object is detected in a frame, a screenshot of the view is saved in the local database. This feature can be useful for various security purposes. 4. TRAINING The training was done using Google Colab so that we could get Tesla K80 GPU for faster and efficient training of the network. After preprocessing dataset i.e. creating label file for each image, both images and their respective label files are to be kept together. The yolo.cfg file was used for training configurations which include three yolo layers. As a traditional method, each object is to be trained for at least 2000 iterations. Hence, the dataset was trained for 6000 iterations as 3(total classes) * 2000 = 6000. The values of batch and subdivisions were set to 64 and 8 respectively for optimal training speed.

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

The width and height values were set at 416 each for optimum speed and better accuracy of detection. The number of filters used in convolution layer were set to 24 as the value is dependent on total number of classes as, filters = (classes + 5)*3. The total amount of time required to train the network with the above configurations was approximately 7-8 hours. The weights thus generated after 6000 iterations were used to carry out detections and analyzing the performance.

4. PERFORMANCE OF ALGORITHMS

The parameters used for testing the completeness of model are mAP, IoU and f1 score. Mean Average Precison(mAP) is the mean value of average precisions and Intersection Over Union(IoU) is the average intersect over union of objects and detections for a certain threshold and f1 score depends on the precision and recall and can be calculated based on confusion matrix.

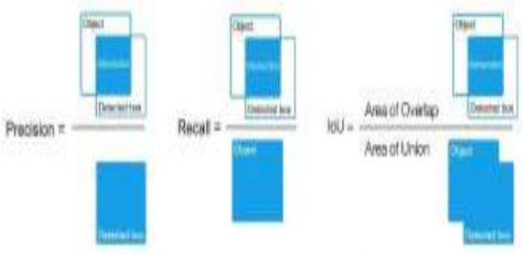


Fig. 14- Performance metrics graphical representation

The following values are the performance metrics obtained on the training dataset:

```
 detections_count = 6685, unique_truth_count = 3796
class_id = 0, name = bus, ap = 98.06%          (TP = 926, FP = 13)
class_id = 1, name = bottle, ap = 97.96%       (TP = 1199, FP = 18)
class_id = 2, name = mobile, ap = 98.41%       (TP = 1305, FP = 55)

 for conf_thresh = 0.25, precision = 0.98, recall = 0.90, F1-score = 0.94
 for conf_thresh = 0.25, TP = 3430, FP = 86, FN = 366, average IoU = 83.19 %

 IoU threshold = 50 %, used Area-Under-Curve for each unique Recall
 mean average precision (mAP@0.50) = 0.981417, or 98.14 %
Total Detection Time: 1200.000000 Seconds
```

Fig. 15 -Performance metrics obtained

5.1 Confusion Matrix: A confusion matrix is a summary that gives us the prediction results on a classification problem. The number of correct as well as number of incorrect predictions are summarized with counted values and broken down class by class. This is the key to the confusion matrix. The confusion matrix shows the ways in which your model is confused when it makes predictions. It gives us the insight not only into the errors being made by a classifier but also more importantly the types of errors that are being made.

For Bus (ClassId = 0), TP = 926 and FP = 13

For Bottle (ClassId = 1), TP = 1199 and FP = 18

For Mobile (ClassId = 2), TP = 1305and FP = 55

IoU: For confidence threshold 0.25, the average IoU is 83.19%

mAP: For IoU threshold of 0.5 i.e. 50%, the mAP is 98.14%

F1-score: For confidence threshold 0.25, the f1-score is 0.94

As the code does not use GPU capabilities of the system for image processing, the required to process the frames by CPU is large. The model requires about 8 seconds to process a frame and display the bounding box over the detected objects. The performance can be substantially improved by utilizing the GPU in the respective system

As the images in the training dataset had the objects to be detected in focus and thus had more object body to size of image ratio, the detection fails for the objects kept far from the camera view. The model performs better in environment with optimum lighting conditions.

TYPES OF TESTS

Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

Functional test

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures : interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

System Test

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

Unit Testing

Unit testing is usually conducted as part of a combined code

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

IntegrationTesting

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

Acceptance Testing

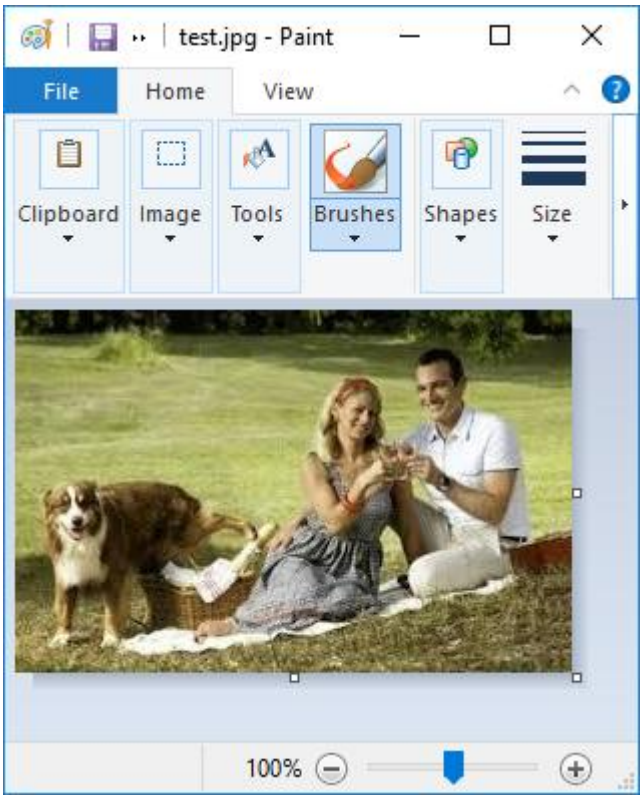
User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

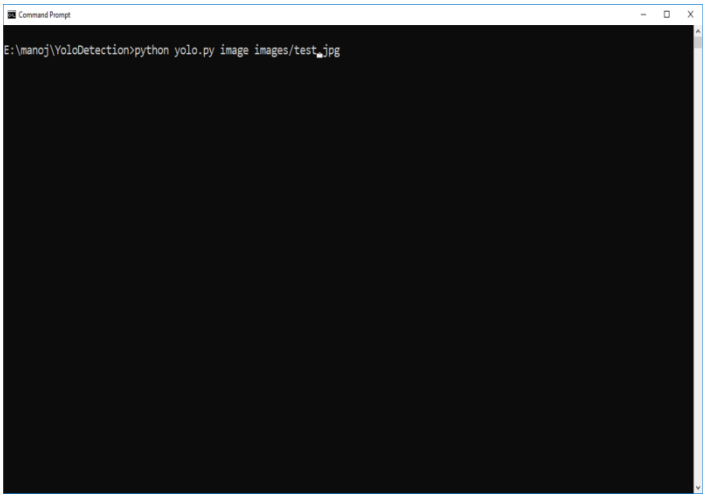
Test Results: All the test cases mentioned above passed successfully. No defects encountered.

RESULTS:

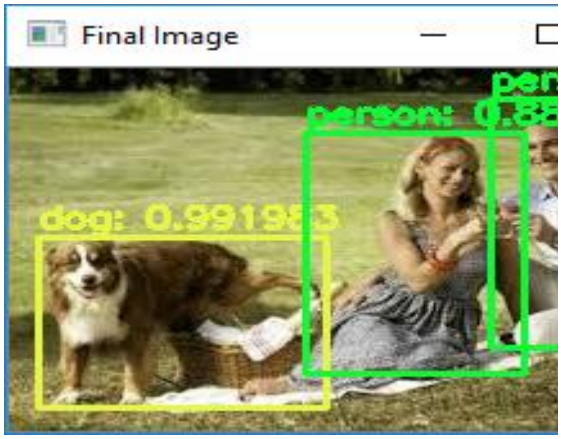
8.1 OUTPUT SCREENS:

For testing I am using below image

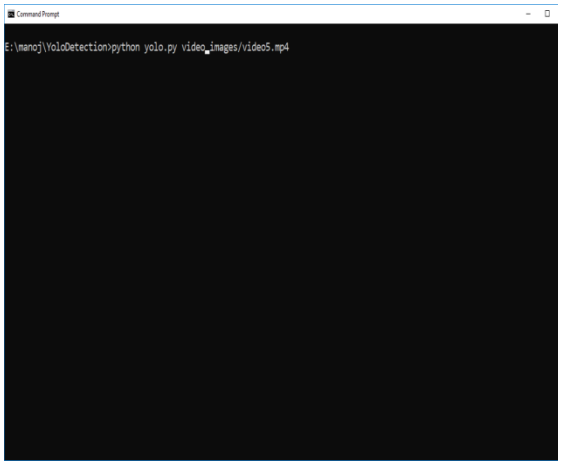




International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362



Use below screen command to run with video



After running above command it will start generating new video when we play will get below screen

CONCLUSION :

In this paper, we have applied and proposed to use YOLO algorithm for object detection because ofits advantages. This algorithm can be implemented in various fields to solve some real-life problems like security, monitoring traffic lanes or even assisting visually impaired people with help of audio feedback. In this, we have created a model to detect only three objects which can be scaled further to detect multiple number of objects.

REFERENCES:

[1] Joseph Redmon, SantoshDivvala, Ross Girshick, "You Only Look Once: Unified, Real-Time Object Detection",The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 779-788.

[2] YOLO Juan Du1,"Understanding of Object Detection Based on CNN Family",New Research, and Development Center of Hisense, Qingdao 266071, China

[3] Matthew B. BlaschkoChristoph H. Lampert, "Learning to Localize Objects with Structured Output Regression", Published in Computer Vision – ECCV 2008 pp 2-15.

[4] Xinyi Zhou, Wei Gong, WenLong Fu, Fengtong Du 'Application of Deep Learning in Object Detection' Information Engineering School, Communication University of China,CUC ,Neuroscience and Intelligent Media Institute, Communication University of China

[5] Allan Zelener - YAD2K: Yet Another Darknet 2 Keras

[6] Official_YOLO_website (https://pjreddie.com/darknet/yolo/)

[7]Andrew Ng's YOLO explanation - https://www.youtube.com/watch?v=9s_FpMpdYW8