International Journal of Scientific Research in Science and Technology
Print ISSN: 2395-6011 | Online ISSN: 2395-602X (www.ijsrst.com)
doi : https://doi.org/10.32628/IJSRST229241

Machine Learning Based Theft Detection Using Yolo Object Detection

[1]Spruha Shamkuwar, [2]Priya Raghtate, [3]Aishwarya Khobragade, [4]Sonali Mohod, [5]Darshana Chokhandre, [6]Shraddha Raut

[1,2,3,4,5] Student of Computer Science Engineering, Nagpur Institute of Technology, Nagpur, Maharashtra, India
[6] Professor of Computer Science Engineering, Nagpur Institute of Technology, Nagpur, Maharashtra, India

Article Info
Volume 9, Issue 2
Page Number : 267-271

Publication Issue
March-April-2022

Article History
Accepted : 01 April 2022
Published : 10 April 2022

ABSTRACT

Object detection and tracking could be an immense, vivacious however inconclusive and trending area of computer vision. Due to its immense use in official surveillances, tracking modules applied in security and lots of other's applications have made researchers to devise a lot of optimized and specialized methods. For validation purpose live input video will be taken for the same where objects will be getting detected and it can be simulated same for real-time through external hardware added. In the end we see the proper optimized and efficient algorithm for object detection and alert for security. Object Detection is computer vision technique used to detect object and identify its localization. This technique is not only used to identify the location but also to identify which type of object it is. This CV technique is used to detect objects in real time while maintaining the level of accuracy. By bringing some advancement in it, this system can be very helpful for people to keep track of their precious things or devices which are very expensive and need to be protected. Open CV (Open-Source Computer Vision Library) is a library of programming functions mainly aimed at real- time computer vision. Open CV features GPU acceleration for real-time operations. This feature helps us to write computationally intensive codes in C/C++ and create a Python wrapper for it so that we can use these wrappers as Python modules.

Keywords: Object detection, vivacious, YOLOv3, Tensor Flow, security, Tracking modules

I. INTRODUCTION

Humans can easily detect and identify objects present in an image. The human visual system is fast and accurate and can perform complex tasks like identifying multiple objects and detect obstacles with little conscious thought. With the availability of large amounts of data, faster GPUs, and better algorithms, we can now easily train computers to detect and classify multiple objects within an image with high accuracy. In this, we will explore terms such as object detection, object localization, loss function for object

Copyright: © the author(s), publisher and licensee Technoscience Academy. This is an open-access article distributed under the terms of the Creative Commons Attribution Non-Commercial License, which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited

detection and localization, and also explore an object detection algorithm known as "You only look once" (YOLO).

Effective and reliable geographic point security is extremely necessary these days. Many times, individuals lose their most vital things and until the time they fight to report a grievance for a similar, time already goes from their hand. What if there's a machine which might facilitate individuals to stay secure for his or her necessary things and not solely keep secure but additionally if any of the necessary things gets miss from its place it provides alert at the same time, so the action will be taken presently against the theft? affirmative, it will be finished with the assistance of some tools and a software package cryptography, trough that we will implement a machine which might be used as a security system. Here we tend to square measure making system exploitation some hardware tools and properties from Machine learning Language. However, there square measure several issues with pictures in real-world shooting like tiny object detection, the accuracy of tiny object detection isn't therefore sensible and these issues have a very important impact on object Detection. So, they are resolved by exploitation YOLOv3 formula here we tend to are exploitation Open CV, Tensor Flow library and YOLOv3 formula and that we are going to be labeling the detected layers with accuracy being checked at a similar time.

1.1 Object Detection

Object detection is a computer vision technique in which a software system can detect, locate, and trace the object from a given image or video. The special attribute about object detection is that it identifies the class of object (person, table, chair, etc.) and their location-specific coordinates in the given image. The location is pointed out by drawing a bounding box around the object. The bounding box may or may not accurately locate the position of the object. The ability to locate the object inside an image defines the performance of the algorithm used for detection. Face detection is one of the examples of object detection.

1.2 Tensor Flow

Tensor flow allows developers to create a graph of computations to perform. Each node in the graph represents a mathematical operation and each connection represents data. Hence, instead of dealing with low-details like figuring out proper ways to hitch the output of one function to the input of another, the developer can focus on the overall logic of the application.

1.3 Yolo

YOLO which stands for "You only look once" is a single shot detection algorithm which was introduced by Joseph Redmon in May 2016. Although the name of the algorithm may sound strange, it gives a perfect description of this algorithm as it predicts classes and bounding boxes for the whole image in one run of the algorithm.

1.4 Open CV

Open CV (Open supply laptop Vision Library)may be a library of programming functions primarily aimed toward period laptop vision. Originally developed by Intel, it had been later supported by Willow Garage then (which was later non-heritable by Intel). The library is cross-platform and free to be used beneath the ASCII text file Apache a pair of License. beginning with 2011, Open CV options GPU acceleration for period operations. This feature helps U.S. to put in writing computationally intensive codes in C/C++ and make a Python wrapper for it so that we will use these wrappers as Python modules. this provides U.S. 2 advantages: 1st, our code is as quick as original C/C++ code (since it's the particular C++ code operating in the background)and second, it's simple to code in Python. this can be however Open CV-Python works, it's a Python wrapper around original C++ implementation

II. LITRATURE REVIEW

1. Object Detection Based on YOLO Network
- Yolo is an algorithm which is used to detect and recognize small objects moreaccurately.
- In comparison to recognition algorithms, a detection algorithm does not only predict class labels but detects locations of objects as well.
2. Object Detection and Tracking using Tensor Flow
- Tensor Flow is the library of Python which plays important role in recognizing and detecting an image.
- TensorFlow is at present the most popular software library. There are several real-world applications of deep learning that makes TensorFlow popular. Being an Open-Source library for deep learning and machine learning, TensorFlow finds a role to play in text-based applications, image recognition, voice search, and many more.
3. Object Detection Tutorial in Tensor Flow: Real-Time Object Detection
- This system captures images and detects the object continuously in real time.
- Real-time object detection is the task of doing object detection in real-time with fast inference while maintaining a base level of accuracy.
4. Object Detection through Modified YOLO Neural Network
- Yolo is an algorithm which is used to detect and recognize small objects accurately and location as well.
- YOLO struggles with small objects. However, with YOLOv3 we see better performance for small objects, and that because of using short cut connections

III. PROPOSED METHODOLOGY

Here, we tend to square measure victimization parts like external camera, computer, Arduino controller, buzzer, and 20x4show|LCD|digital display| alphanumeric display} display. The camera can capture the pictures returning before of it and also the detection half is finished victimization cryptography within the comp/ portable computer. The Arduino nano used here is for serial communication between the pc, LCD, and buzzer. Buzzer used here provides us the alert of the object being incomprehensible and to show the names of explicit objects incomprehensible, we tend to square measure victimization LCD

Proposed Block Diagram:

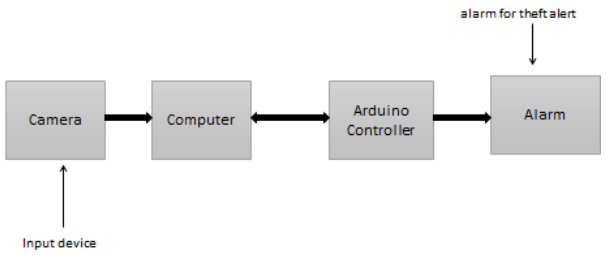


Fig. 3.1 Block diagram of Proposed methodology

Object detection is a computer technology related to computer vision and image processing that deals with detecting instances of semantic objects of a certain class (such as humans, buildings, or cars) in digital images and video methods for object detection generally fall into either machine learning-based approaches or deep learning-based approaches.

Here, we are using deep learning-based approach via YOLO (You only look once). Starting the concept, first the camera will work as an input for the system, then after taking in the image it will initialize and start capturing device.

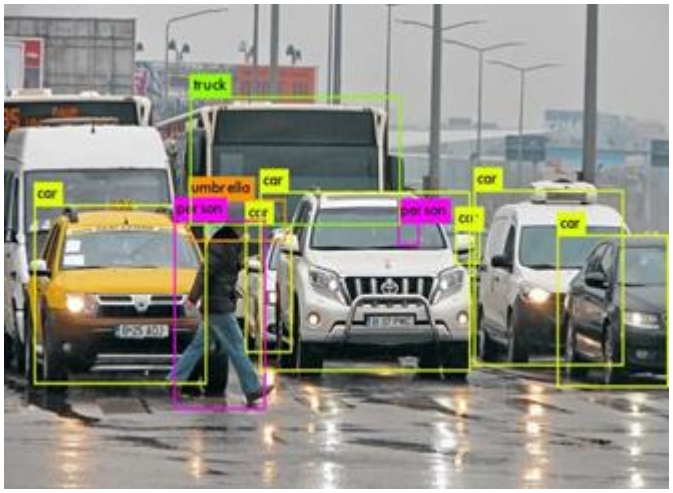
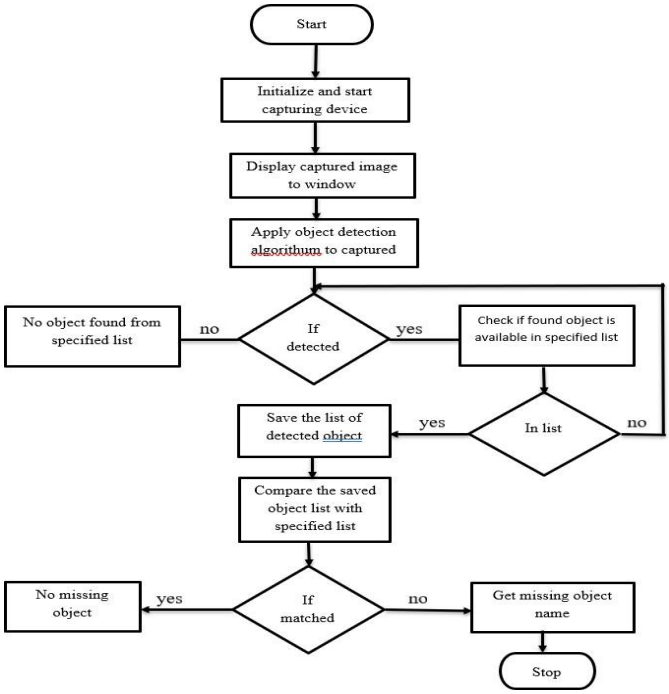


Fig. 3.2 : Forming of Bounding Boxes around Detected Objects & Labelling them



According to flow diagram, first the camera will work as an input for the system, then after taking in the image it will initialize and start capturing device. After capturing the image to window we will apply object detection algorithm to captured image.

Then condition applies if object isn't detected then system will say no object found from specified list, if detected is a yes, then check if found object is available in specified list, if in list then save the list of detected objects and compare the saved object list with specified object list, if match found then system will display no object missing and if match not found then system will show alert on display and buzzer will ring.

IV. IMPLEMENTATION

The following subsequent specs have been used to build this module: • Quantum QHM495LM Web Camera interpolated to 2 Megapixels. • Arduino nano Atmega328P Processor • Display 20(x4 LCD Module)• Computer / Laptop: Windows 8 i5 8th gen Intel Core Processor 64(-bit)

V. RESULT

The following figures show the output when the objects are present in front of the camera and when it is missed from its place. The GUI of the output screen is designed in Qt source.

Fig 5.1 : When all objects are present

In figure 5.1, every object is being detected when placed in front of the camera, along with that we can see their labels (names), bounding boxes and confidence in %. On the top right corner, we can see the found objects block which shows the detected objects name. The objects which we want to keep in our database are being added in saved objects block. These three objects are being saved in our database camera, CCTV, drone, etc. would like we'd like we want to contemplate that we'll need object detection systems that will explore areas that haven't been seen by humans. In such cases, a period of open-world intelligence is going to be important, thus we want a system that is adaptive to the quickest technologies that leading.

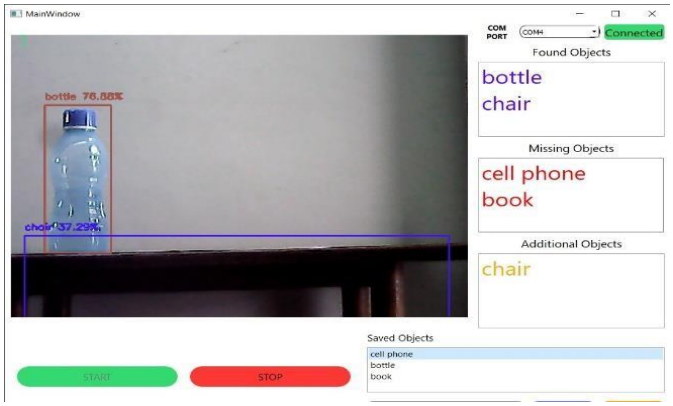


Fig 5.2: When objects are missing and additional objects are added.

In figure 5.2, the two objects are missing and are not in the frame so we can see their names in missing objects block. If the camera detects an object apart from the database the object comes under the additional object list. Here, chair is not in our database so it has been shown in additional objects block

VI. CONCLUSION AND FUTURE SCOPE

Despite fast development and achieved promising progress of object detection, there are several areas that we can add future to the primary one is little(small) object detection like occurring in the COCO palm dataset to enhance localization accuracy on little(small) objects. Another potentially beneficial extension would be to modify the algorithm to perform recursive, coarse-to-fine segmentation, thereby achieving better performance, and allowing the algorithm to segment higher-resolution images.

VII. REFERENCES

[1]. Chengji Liu, Yufan Tao, jaiwei Liang "Object Detection Based on YOLO Network" 2018 IEEE 4th Information Technology and Mechatronics Engineering Conference (ITOEC) 10.1109/ITEOC.2018.874064

[2]. R. Sujeetha, Vaibhav Mishra "Object Detection and Tracking using Tensor Flow" ISSN: 2277-3878, Volume-8, Issue-1, May 2019

[3]. Kislay Keshri - "Object Detection Tutorial in Tensor Flow: Real- Time Object Detection"

[4]. Tanvir Ahmad - "Object Detection Through Modified YOLO Neural Network" International Journal of Engineering Research & Technology (IJERT), volume 2020 | Article ID 8403262

Cite this article as :

Spruha Shamkuwar, Priya Raghtate, Aishwarya Khobragade, Sonali Mohod, Darshana Chokhandre, Shraddha Raut, "Machine Learning Based Theft Detection Using Yolo Object Detection", International Journal of Scientific Research in Science and Technology (IJSRST), Online ISSN : 2395-602X, Print ISSN : 2395-6011, Volume 9 Issue 2, pp. 267-271, March-April 2022. Available at doi : https://doi.org/10.32628/IJSRST229241
Journal URL : https://ijsrst.com/IJSRST229241