International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 11 Issue: 05 | May 2024 www.irjet.net p-ISSN: 2395-0072

Development of Smart Alerting System using Real Time Object Detection with Deep Learning

Satish Dekka[1], Karanam Prameela[2], Padala Mohan Anantha Rama Reddy[3], Manapuram Sai Charan[4], Kadali Varahala Bhagya Sri[5]

[1] Associate Professor, Dept. of Computer Science Engineering, Lendi Institute of Engineering and Technology Vizianagaram, India
[2-5]B.Tech Final Year, Computer Science Engineering, Lendi Institute of Engineering and Technology Vizianagaram, India

---***---

Abstract—*Deep learning is currently the mainstream method of object detection. This project introduces a novel approach that enhances real-time object detection by integrating the You Only Look Once (YOLO) algorithm with Fast SMS capabilities. Our proposed system leverages the strengths of YOLO to handle complex scenarios such as occlusion, deformation, and small object sizes. The system's architecture involves the integration of Deep learning with Fast SMS Services, enabling real-time detection and interaction with the physical environment. This fusion of advanced deep learning with Fast SMS offers a comprehensive solution for efficient and responsive object detection in dynamic environments. Trained on nearly 40 classes, the trained model uses Darknet for class implementation. As a real-time object detector, the system detects objects while the webcam is on and sends messages to mobile phones using Fast SMS Service. The integration of both deep learning and Fast SMS Service enables efficient and rapid identification of objects in streaming video feeds. The system leverages the power of YOLO's single-pass architecture for simultaneous detection of multiple objects with high accuracy. The detected object information is seamlessly communicated to different devices, facilitating quick decision making and response in various applications such as smart surveillance, automated monitoring, and real-time analytics.*

Keywords— Deep Learning, Object Detection, You Only Look Once (YOLO) Algorithm, Fast SMS, Smart Surveillance, Real-Time Analytics.

I. Introduction

Over the years, various strategies have been proposed to address the challenge of object identification, with a focus on multi-stage solutions. These approaches typically involve stages such as recognition, classification, localization, and object detection. However, despite technological advancements, these techniques have encountered challenges related to output accuracy, resource costs, processing speed, and complexity. The development of Convolutional Neural Network (CNN) algorithms, starting with the Neocognitron-inspired algorithm in the 1990s by Yann LeCun et al., marked a significant milestone in object detection. This was further advanced by AlexNet, which achieved a groundbreaking victory in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) in 2012.

These CNN algorithms have since been instrumental in offering solutions to object detection problems using various approaches. To enhance the accuracy and speed of the recognition, subsequent optimization-focused algorithms have been introduced. Notable examples include VGGNet, GoogLeNet, and Deep Residual Learning (ResNet), which have been developed over the years. These algorithms have played a crucial role in advancing object detection capabilities, pushing the boundaries of what is achievable in terms of accuracy and efficiency. Computer vision and visual processing of images and tones have traditionally been significant for large organizations with the resources to invest in advanced technology.

However, the advent of affordable computers and mass-produced devices like YOLO (You Only Look Once) has democratized these capabilities. YOLO, originally developed for commercial and academic purposes, now enables enthusiasts worldwide to develop software and real-time embedded programs that can process images using current hardware standards. The applications of software for object detection and computer vision are diverse and abundant. Examples range from tracking visitors in shopping malls and face recognition systems to monitoring equipment, assisting robots, analyzing product labels, and deep learning projects.

This Bachelor's Thesis aims to create a database on embedded systems, real-time image processing, and visual effects to explore further layers of complexity and research. The goal is to build projects based on more affordable and less real-time visual effects compared to sophisticated systems, such as self-learning robots and smart machines capable of visual feedback. Object detection is one of the most exciting applications of convolutional neural networks in computer vision. Apart from simple image classification, object detection is associated with various applications, including self-driving vehicles that integrate computer vision, LIDAR, and other technologies to create comprehensive street representations. There is a misconception that training-based analysis outperforms legal incentive methods when the training and test data come from the same source.

However, if this is not the case, the performance of the training-based compound could be poor, potentially worse than using a single detector. The main objective of the project is to organize a C/C++ program with the help of OpenCV libraries for use in UNIX-based embedded systems, specifically YOLO. The program's goal is to detect colored bulbs in a live camera feed, keep them in memory under certain conditions to monitor their interactions and properties, and send useful information about these objects as datagrams using network connections with the UDP protocol for any software application. The project's focus on affordable and less real-time visual effects compared to sophisticated systems opens up new possibilities for a wide range of applications.

By leveraging the capabilities of YOLO and OpenCV libraries, the project aims to enable the development of innovative solutions that were previously limited to high end systems. This approach not only reduces costs but also democratizes access to advanced computer vision technologies, allowing a broader community to participate in the development of intelligent systems. One key aspect of the project is its potential impact on industries that can benefit from real-time object detection and visual processing. For example, in manufacturing, the ability to accurately detect and track objects in real time can enhance quality control processes and improve overall efficiency.

Similarly, in 2 surveillance and security, real-time object detection can help identify potential threats and improve response times. By exploring these applications and pushing the boundaries of affordable visual processing, the project aims to contribute to the advancement of various industries and technologies. The project's exploration of object detection using convolutional neural networks (CNNs) and OpenCV libraries, particularly focusing on detecting-colored bulbs in live camera feeds, has practical implications. By successfully implementing such a system, it could be deployed in various settings. For example, in smart home applications, the system could be used to monitor and control lighting based on the presence or absence of individuals in a room, leading to energy savings and enhanced user comfort. In industrial settings, the system could be used for automated quality control, where the detection of colored markers on products could indicate specific characteristics or defects. Furthermore, the project's integration of real-time image processing with embedded systems opens up possibilities for applications in robotics and automation.

For instance, in warehouse environments, robots equipped with such technology could navigate through cluttered spaces more effectively, avoiding collisions and optimizing their paths. Similarly, in agricultural settings, drones equipped with these capabilities could be used for crop monitoring and management, providing farmers with valuable insights to improve crop yields.

The use of the User Datagram Protocol (UDP) for sending information about detected objects to software applications adds a layer of connectivity and versatility to the project. By utilizing network connections, the system can seamlessly integrate with other devices and systems, enabling complex interactions and workflows. This aspect of the project not only demonstrates the practical implementation of object detection but also showcases the potential for interconnected systems to enhance efficiency and functionality in various domains.

II. LITERATURE SURVEY

M Kristo et al., [1] proposed that Dangers from global psychological oppressors and illicit migrants have strengthened worries for the wellbeing of residents, and each exertion has been made to use all innovative advances to forestall significant occurrences and to ensure individuals and property. M Haroon et al., [2] examines that issue of high-goal advanced picture securing by distant detecting by presenting a medium-sized word reference novel with a satellite picture name, a multi-dimensional picture information base (SIMD) and a solitary travel object outline intended for multidisciplinary sees.W Fang et al., [3] proposed the Tinier YOLO, the fire module at SqueezeNet was chosen by examining the quantity of fire modules and their situations in the model to lessen the quantity of model boundaries and decrease the model size. S Birogul et al., [4] are Getting persistent measures and experiences in the visual markers of a theory mechanical assembly in the past requires a substitute point of view on the diagrams. improvement of the theory thing over the 2D light diagram. The model is set up in the front-line structure. W Wu et al., [5] It is planned for issues with the disclosure of the electrical connector, for example, low robotization, low obtaining precision, slow procurement speed, and heartiness, and This paper introduces an enhanced YOLOv3 algorithm designed for the detection of electrical connections. Q Xu et al., [6] The accessibility of traffic data during driving is critical to the driver. Be that as it may, drivers are regularly incapable to see numerous snippets of data immediately, which unavoidably builds certain dangers. X Wang et al.,[7] The proposed structure in this paper can be utilized in clinical analysis, since test settings are adaptable and recognition is quick. The data set out in this record is given by FPM itself. KJ Kim et al., [8] Ongoing exploration on inside and out learning has indicated a great deal of elite execution that has not been accomplished with conventional AI procedures. Particularly in the field of article procurement, business items with high exactness 1 0 in reality are utilized for top to bottom learning strategies. V Paidi et al., [9] Stopping has been a typical issue for quite a long while in numerous urban areas around the globe. Interest for stopping prompts blockage, disappointment and expanded air contamination. Numerous insides and out learning organizations. S Guennouni et al., [10] Item disclosure pulled in a ton of interest because of the wide scope of uses it employments. Visual sharpness innovation is driven by the intensity of persistent execution found in programming and equipment. M Kushal et al. [11] The fundamental reason for a PC vision is to distinguish and perceive an assortment of objects of various sizes, surfaces, and shapes. The most serious issues confronting PC vision are edificationand item insight, Harini et al., [12] This existing system proposed that R- CNN was one of the early approaches for object detection, which relied on region proposals followed by classification. It achieved state-of-the-art results but suffered from slow processing speeds due to its multi-stage architecture. KEYOU GUO et al.,[13] Approaches like feature pyramid networks (FPN) and spatial pyramid pooling (SPP) enable models to capture object information at various scales, facilitating accurate detection of objects of different sizes. Tanvir Ahmad et al., [14] system proposed that deep learning-based object detection methods like Faster R-CNN, SSD, and YOLO have shown superior performance compared to traditional methods, leveraging CNNs for feature extraction and classification. Dekka et al., [15][16] Crop yield forecasting is a crucial aspect of precision agriculture. Crop yield forecasts are required for thorough planning, policy development, and execution for choices regarding, among other things, the procurement, distribution, price fixing, and import-export of crops. The major goal of this research is to recommend to farmers the best crop based on site-specific information such as soil PH level, temperature, humidity. Breast cancer has recently become a highly serious disease, not just in India but also in other nations. The primary goal of this research is to diagnose breast cancer patients as early as possible. Three machine learning approaches Decision Tree, Support Vector Machine, and Logistic Regression are employed for the early detection and prevention of breast cancer patients.

III. DESIGN AND IMPLEMENTATION

The implementation of the Smart Alerting System using realtime object detection with deep learning involves several key steps and technologies. Firstly, the project leverages the You Only Look Once (YOLO) algorithm, which is known for its efficiency in detecting objects in images. YOLO sees the entire picture during both training and testing, allowing it to fully integrate the content of classes and their appearance. This is crucial for our project, as we need to accurately detect and classify objects in real-time scenarios.

The YOLO algorithm starts by taking an input image and dividing it into different grids. Each grid is then characterized and limited, with YOLO predicting bounding boxes and object class probabilities. This process enables the algorithm to learn automated representations of objects, making it effective for a wide range of applications, including real-time object detection.

To train the YOLO model, a large dataset of images containing various objects with bounding boxes and class labels is used. The network architecture is designed specifically for object detection, with a deep convolutional neural network (CNN) used to predict bounding boxes and class probabilities for potential objects in an image. The model is trained in two stages, with a predictor aiming to predict accurate bounding boxes and class probabilities, and a loss calculator evaluating the predictions' accuracy.

After training, the model is evaluated on a separate validation dataset, and fine-tuning may be done based on the evaluation results. In the object detection phase, the pre-trained YOLO model is loaded, and the webcam is initialized to capture video frames. The captured frames are preprocessed and passed through the YOLO model for object detection. Detected objects exceeding a minimum confidence threshold trigger SMS alerts containing the object information, providing real-time monitoring and alerting capabilities.

Overall, the Smart Alerting System integrates advanced object detection techniques with real-time monitoring and alerting capabilities, making it suitable for a variety of applications such as security surveillance, traffic monitoring, and more. The project demonstrates the power of deep learning and computer vision in creating intelligent systems that can enhance safety and efficiency in various environments.

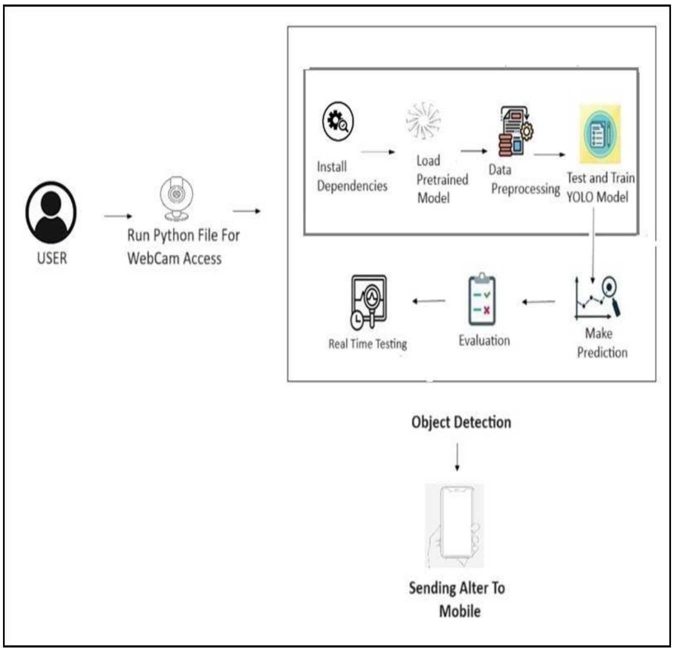

Fig. System Architecture of Smart Alerting System

The initial step involves executing a Python script that utilizes OpenCV or a similar library to access the webcam. This allows the system to capture a live video feed that will be used for real-time object detection. Before running the main code, it's essential to install the required libraries and dependencies. This typically includes libraries like OpenCV for computer vision tasks, NumPy for numerical computations, and requests for sending alerts. After, a pretrained YOLO (You Only Look Once) model which is trained on a coco dataset is loaded. YOLO is a popular 19 deep learning model known for its speed and accuracy in object detection tasks. The pretrained model has been trained on a large dataset called COCO Dataset and can detect a variety of objects. Data Preprocessing Before feeding the video frames to the YOLO model, it's essential to preprocess them. This may include resizing the frames to match the input size expected by the model, normalizing the pixel values, and possibly converting the color space if needed. Test and Train YOLO Model If the pretrained YOLO model does not provide satisfactory results for the specific object detection task, further training may be required. This involves using labeled datasets to train the model on specific objects or classes relevant to the smart alerting system.

Once the YOLO model is loaded and ready, it can be used to make predictions on the live video feed. The model will analyze each frame and detect objects present in the scene along with their locations. After making predictions, the system can evaluate the performance of the YOLO model. This typically involves calculating metrics such as accuracy, precision, recall, and F1 score to assess how well the model is detecting objects. With the YOLO model in place, the system can perform real-time testing by continuously analyzing the live video feed. The model should be able to detect objects as they appear in the scene and provide alerts if necessary.

When a specific object or event of interest is detected, the system can trigger an alert. This alert can be sent via various communication channels, such as email, SMS, or through a mobile app. In now, continuously detect objects in the environment using the pre-trained YOLO model. When a new object is detected, generate an alert message indicating the presence of the object. Message Sending: Send the alert message to predefined mobile phone numbers using the Fast2SMS API or a similar service. Wait for a confirmation or acknowledgment from the user to stop sending alerts for the detected object. End Monitoring: Stop monitoring and sending alerts when the user stops the execution by Esc Key.

WORKING

YOLO sees the whole picture during training and test time and therefore fully integrates the content of the classes and their appearance. YOLO learns automated presentations of objects so that when they are trained in natural images and tested in art, the algorithm surpasses other advanced acquisition methods.

Now that we have understood why YOLO is such a viable foundation, let's get into its practical reality. In this section, I mentioned the steps followed by YOLO to find items in certain images.

- The YOLO begins with taking a input image:

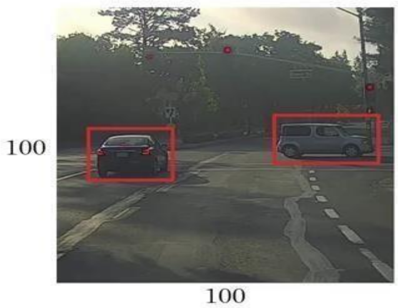

Fig. Creating bounding boxes for objects

- The frame work further divides an input into different grids:

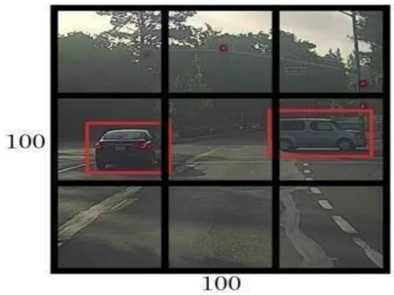

Fig. Grid Creation

- Characterization and constraints are applied to each grid in the image. YOLO predicts responsibility boxes and their comparing object learning openings (if accessible, obviously).
- Truth be told, right? How about we separate each progression so we can get a clearer comprehension of what we have recently perused.
- We need to move the marked data to the model to shape it. Assume we split the picture into a matrix size 3 X 3 and there is a sum of 3 classes that we need things to be isolated into. Assume the Pedestrian, Car, and Motorcycle classes individually.
  In this manner, for every cell of the framework,

y =	pc
	bx
	by
	bh
	bw
	c1
	c2
	c3

Fig. Labels of detected objects

- where, the pc in table specifies is an object exists in position or not the bx and by and bh and bw also specify a binding box if occurring an object and also for c1 and c2 and c3 also represent the classes. therefore, if an input is a car suppose, then c2 will be value of 1 and c1 and c3 will be going to 0, and go on. For example, if select the 1st box in above example:

Fig. image with no objects

- And there was no article in this above matrix, pc will be 0 and the point y mark for this framework will be:

y =	0
	?
	?
	?
	?
	?
	?
	?

Fig. Labels of zero object image

Single Object Detection:

Before compose the name y of this framework, it is significant that we initially see how YOLO decides if there truly is something in the matrix. In the image above, there are two items (two vehicles), so YOLO will take a point between the two articles and these articles will be given to the framework that contains the point between the two articles. The mark of the focal point of the left lattice and the vehicle will be:

y =	1
	bx
	by
	bh
	bw
	0
	1
	0

Fig. Car object detection with labels

- Since there is something in this lattice, the pc will be equivalent to in this manner, for every one of the 9 lattices, we will have a most extreme yield vector. This yield will have a 3 X 3 X 8 shape. So now we have the establishment picture and it compares to the predetermined vector. 1 9 We will lead forward and in reverse transmissions to prepare our model. During the test stage, we move the picture to the model and push it ahead until we get the outcome y. To keep things basic, I clarified this utilizing a 3 X 3 framework here, however normally in true circumstances we take bigger lattices. Regardless of whether a thing broadens more than one matrix, it might be allotted to one lattice where its main issue is. We can lessen the odds of different items showing up on a similar network cell by expanding the quantity of lattices
- points are determined comparative with the network cell we are managing. Consider the correct focus framework containing the vehicle:

Fig. Bounding box of car object

- Therefore, points will be calculated in relation to this matrix. Then it has:

y =	1
	bx
	by
	bh
	bw
	0
	1
	0

Fig. Labels of detected object

International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 11 Issue: 05 | May 2024 www.irjet.net p-ISSN: 2395-0072

Then point pc = 1 and also there is something in matrix and it is someone's car. Now,

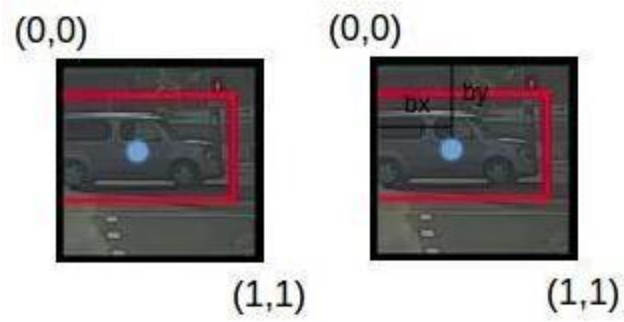

Fig. Finding Centre of bounding box

bh the proportion of the tallness of the coupling box red box in the model above to the stature of the relating lattice cell, which to us is around 0.9. Along these lines, bh = 0.9. bw the normal width of the associating box and the width of the framework cell. Thusly, bw = 0.5 (roughly). The mark for this matrix will be:

$$y = \begin{array}{|c|} \hline 1 \\ \hline 0.4 \\ \hline 0.3 \\ \hline 0.9 \\ \hline 0.5 \\ \hline 0 \\ \hline 1 \\ \hline 0 \\ \hline \end{array}$$

Fig. calculation of different labels

Note here that bx and by will stay a distance somewhere in the range of 0 and 1 as the middle point will consistently exist in the lattice. While bh and bw can be multiple if the size of the coupling box is more prominent than the size of the matrix.

In the following segment, we will take a gander at certain thoughts that can help us make this calculation work better.

Crossing point over Union and Non-Max Suppression

Here are some provocative feeds - how might we decide whether an anticipated box gives us a positive (or negative) impact. That is the place where the Intersection over Union comes in. Figures the connection between the genuine restricting box and the anticipated restricting box. Consider the genuine and prescient responsibility boxes of a vehicle as demonstrated as follows:

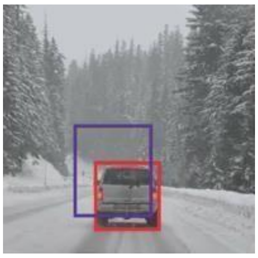

Fig. Finding cross point over union

Here, the red box is a genuine restricting box and the blue box is the thing that is anticipated. How might we decide whether a decent expectation or not? The IoU, or Union Crossroads, will compute the intersection at the intersection of these two boxes. That spot will be:

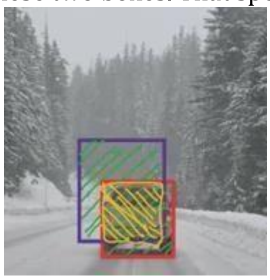

Fig. IOU calculation

IoU = Meeting place/Union spot,

IoU = Yellow box area/Green box area

Multiple Object Detection:

- In the event that the IoU is above 0.5, we can say that the expectation is adequately exact. 0.5 is the constraint of the contention we have taken here, however can be changed relying upon your particular issue. As a matter of course, the bigger the breaking point, the better the expectation.
- There is just a single method to improve your YOLO yield - Non-Max Pressure.
- Perhaps the most widely recognized issues with object identification calculations is that as opposed to getting an item once, they can get it again and again. Consider the image underneath:

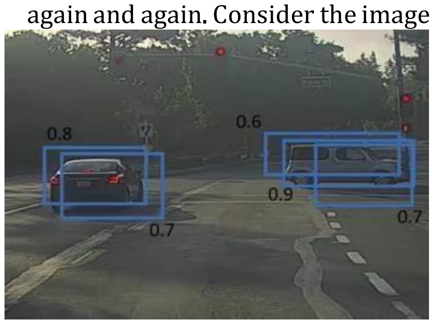

Fig. Detecting multiple objects with coordinates

Here, vehicles are recognized more than once. The Non-Max Suppression Process tidies up this so we can get a solitary identification of every thing. We should perceive how these functions.

1. Its beginnings by taking a gander at the open doors related with every obtaining and afterward takes the greatest one. In the figure above, 0.9 is likely, so a crate with a likelihood of 0.9 will be chosen first:

Fig. Examining first object's bounding boxes

2. Now, examine all the remaining boxes in the image. Boxes with high IoU and current box are compacted. Hence, boxes with a likelihood of 0.6 and 0.7 will be squeezed in our model:

Fig. Examine 1st object's another bounding box

3. After the containers are squeezed, select the following box of all the cases with the most elevated possibility, it is 0.7 for us:

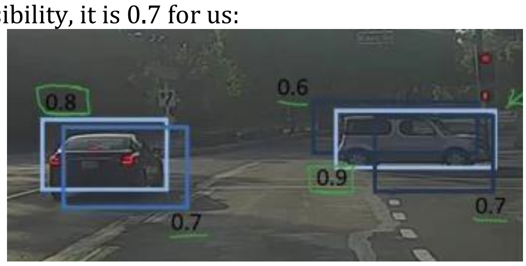

Fig. Examine 2nd object's bounding boxes

4.It will likewise check the IoU of this container and the excess boxes and press the cases with the top IoU:

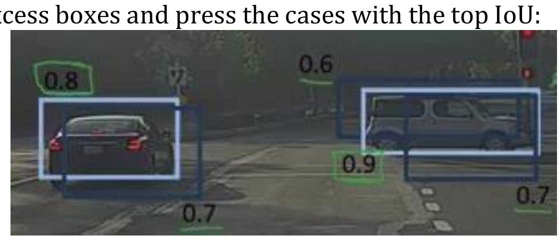

Fig. Examine 2nd object's another bounding box

5.We recurrent these means until all the containers have been chosen or squeezed and get the last restricting boxes:

Fig. Object detection based on confidence score

Here is the thing that Non-Max Suppression implies. We take boxes with extraordinary potential and press boxes extremely near non-max openings. Let us rapidly sum up the focuses we have found in this segment with respect to the non-Max concealment calculation:

Dispose of all cases with not exactly or equivalent to the edge recently depicted (state, 0.5).

For the leftover boxes:

- Select the container with the most elevated likelihood and accept that as the yield forecast
- Dispose of some other box with IoU bigger than the breaking point and the yield box from the above advance
- Rehash stage 2 until all the crates are viewed as an expectation of release or disposed of
- There is another way we can improve the presentation of the YOLO calculation

Anchor Boxes

We've seen that every framework can highlight just a single thing. In any case, imagine a scenario in which there are an excessive number of things in a single matrix. That can regularly occur. Furthermore, that drives us to moor boxes. Think about the accompanying picture, separated by a 3 X 3 network:

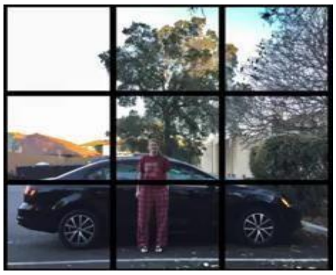

Fig. Grid formation

- We supplanted the item and took a gander at its area, allotting the item to the relating network in the model.
- over, the midpoint of the two articles lies in a similar framework. This is the way genuine bundling boxes will resemble:

Fig. Anchor box creation

We will just get one of the two boxes, either via vehicle or by individual. In any case, in the event that we use stop boxes, we can pull out both boxes! How would we do this? To begin with, we depict ahead of time two distinct shapes called suspension boxes or anchor box shapes. We can generally build the quantity of anchor boxes. I have taken two here to make the idea more obvious:

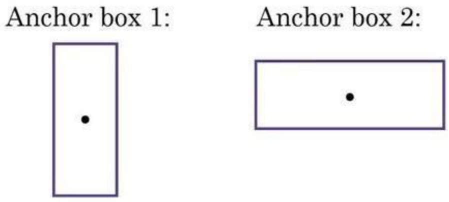

Fig. Anchor box centers

This course is the thing that the YOLO mark without anchor boxes resembles:

$$y = \begin{bmatrix} pc \\ bx \\ by \\ bh \\ bw \\ c1 \\ c2 \\ c3 \end{bmatrix}$$

Fig. Object labels for single object

What do you figure the mark y will be on the off chance that we have two anchor boxes? I need you to pause for a minute to think about this prior to perusing further. The name will be:

$$y = \begin{bmatrix} pc \\ bx \\ by \\ bh \\ bw \\ c1 \\ c2 \\ c3 \\ pc \\ bx \\ by \\ bh \\ bw \\ c1 \\ c2 \\ c3 \end{bmatrix}$$

Fig. Object labels for both objects

The initial 8 lines are for the anchor box and the excess 8 are for the anchor box 2. The items are given suspension boxes dependent on the comparability of the coupling boxes and the state of the anchor box. The yield for this situation, rather than 3 X 3 X 8 (utilizing a 3 X 3 matrix and segments 3), it will be 3 X 3 X 16 (since we utilize 2 anchors). In this way, for every framework, we can discover at least two things dependent on the quantity of anchors. How about we sum up all the thoughts we have composed up until now and connection them to the system.

METHODOLOGY

This project is carried out to analyze the performance of Detection with YOLO and to record its properties under various test conditions. The condition will vary by changing the following parameters.

- Color of Image.
- Quantity of Image.
- Various Image distances.
- Different Image size.

The above-mentioned conditions are tested in this setup to determine the performance of Object Detection using YOLO. The following properties of YOLO are measured with the practical experiment of the device setup. They are

- Efficiency
- Security
- Capacity

In the object detection module, the system uses a pretrained deep learning model, such as YOLO (You Only Look Once), to detect objects in real-time video streams. This module processes the video frames, identifies objects, and draws bounding boxes around them. The alerting module is responsible for sending notifications to the user's mobile phone when specific objects are detected. It uses Fast SMS Service to send alerts, ensuring timely communication of important events. Together, these modules enable the system to detect objects of interest and notify the user, enhancing situational awareness and enabling prompt action.

RESULTS

Picture arrangement includes anticipating the class of one item in a picture. Confinement of an item alludes to seeing the area of at least one articles in a picture and drawing a crate brimming with their width. Article securing consolidates these two capacities and is restricted and isolates at least one items in the picture.

Article securing, a fundamental task in computer vision, plays a crucial role in various applications such as object detection, image segmentation, and scene understanding. It involves identifying and delineating the boundaries of objects within an image, typically by drawing bounding boxes around them. This process enables machines to understand and interact with visual data, facilitating tasks like object recognition, tracking, and classification. By combining the capabilities of object classification and localization, article securing provides a foundational step towards more complex visual understanding tasks, enabling machines to interpret and respond to visual information in a manner similar to humans.

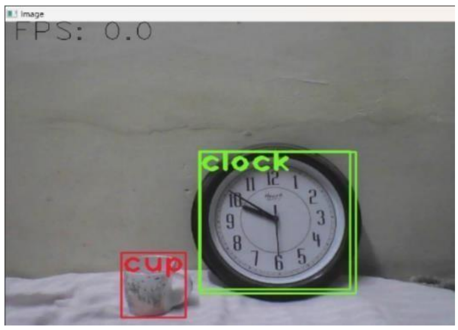

Fig. Multiple Object Detection in live webcam video

In this image it recognizes the multiple objects like clock and cup with good and fine accuracy this makes sense that it doesn't matter how many objects are there. The main goal is to find and detect the object individually with high accuracy.

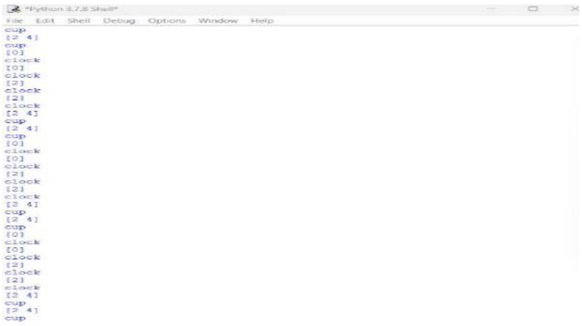

Fig. FPS of Detected Objects

The image shows the successful results after detecting multiple objects, including, clock and cup. This demonstrates the model's ability to detect and classify various classes of objects simultaneously, essential for applications like traffic surveillance and safety.

Fig. Sent Message of Detected Objects

After detecting clock and cup objects, the system has sent a message as an alert or notification, indicating that the objects have been detected. This functionality demonstrates the real-time capabilities of the system in identifying and responding to specific objects of interest, which can be crucial for applications such as security monitoring, inventory management, or environmental sensing.

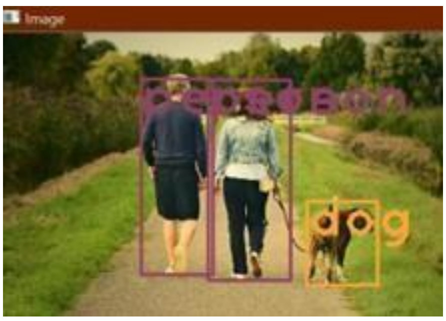

Fig. Object Detection from behind

Even though the image was taken from backside the recognizer that means the YOLO algorithm that takes every side accuracy and grid calculation so that it known for best algorithm for object detection with good accuracy.

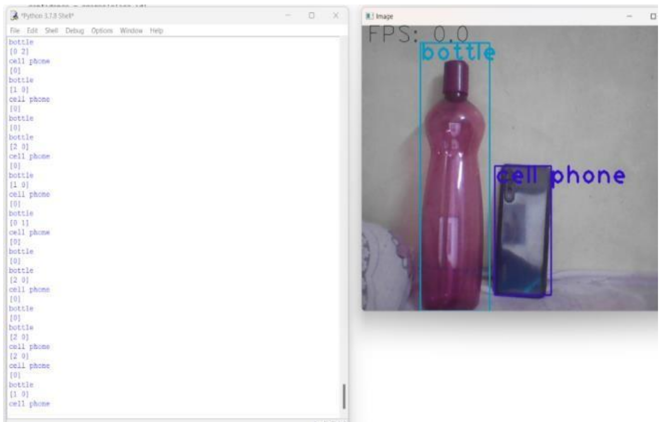

Fig. Bottle and Mobile Detection

The fig illustrates the detection of a mobile phone as an object by the system, with the detection result and frames per second (fps) displayed in the Pythoninteractive shell.

```
Python 3.7.8 (tags/v3.7.8:4b47a5b6ba, Jun 28 2020, 08:53:46) [MSC v.1916 64 bit (AMI
Type "help", "copyright", "credits" or "license()" for more information.
>>>
======= RESTART: C:/Users/SAI CHARAN/OneDrive/Desktop/objectdetection.py =======
Mobile Detected
return:true request_id:gt1m02efaksz4hw message:[Sms sent successfully]}
Bottle Detected
return:true request_id:gt1m02efaksz4hw message:[Sms sent successfully]}
>>> |
```

Fig. Acknowledgement of sent Message

The fig indicates the successful approval of the request to send two different messages, confirming the system's readiness to transmit the specified alerts.

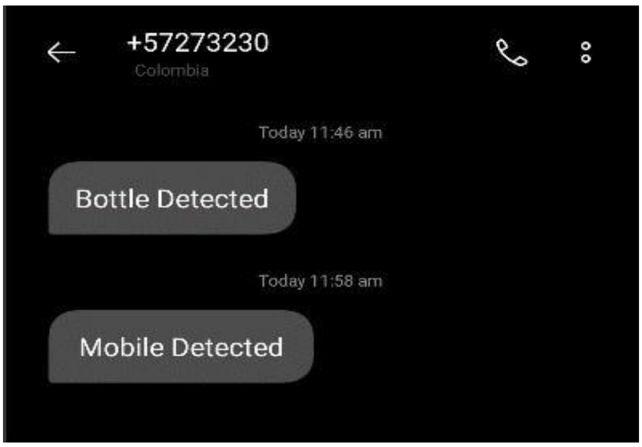

Fig. Acknowledgement of sent Message

RESULT ANALYSIS

The results of the project demonstrate the successful implementation of the smart alerting system using real-time object detection with deep learning. The system effectively detects and classifies various objects, such as clocks, cups, mobile phones, and bottles, in live webcam video streams with high accuracy. The ability to detect multiple objects simultaneously, as shown in the results, highlights the system's robustness and reliability in real-world scenarios.

The system demonstrates its real-time capabilities by promptly sending alerts or notifications upon detecting specific objects of interest. This functionality is crucial for applications requiring immediate action or response, such as security monitoring or inventory management. The successful transmission of messages, as indicated in the results, confirms the system's ability to interact with external devices or services, enhancing its utility and versatility.

The high accuracy and efficiency of the object detection algorithm, particularly the YOLO algorithm, are evident in the results. The algorithm's ability to detect objects from various angles and perspectives, as demonstrated in the results, makes it a suitable choice for object detection tasks requiring comprehensive coverage and precision. Overall, the results showcase the effectiveness and practicality of the smart alerting system, highlighting its potential for deployment in various real-world applications requiring object detection and alerting functionalities.

S.No.	Description	Input	Expected Output	Actual Output	Result
1	Detection of a traffic light	Video feed	System detects atraffic light andsends an alert.	Traffic Light detected and alert sent.	Pass
2	Detection of a cat	Video feed	System detects thecat and sends an alert.	Cat detected and alert sent.	Pass
3	Detection of Laptop	Video feed	System detects thelaptop and sends an alert.	laptop detected and alert sent.	Pass
4	Detection of Person	Video feed	System detects theperson and sends an alert.	person detected and alert sent.	Pass
5	Detection of person	Video feed	System detects thetraffic light and sends an alert.	person doesn't detect because of far distance.	Fail

Fig. Test Cases

IV. CONCLUSION

The development of our project, represents a significant advancement in safety and security technology. By combining state-of-the-art object detection algorithms with real-time alerting mechanisms, we have created a system that can accurately identify and respond to potential threats in various environments. The use of deep learning techniques, such as the YOLO (You Only Look Once) algorithm, allows for fast and efficient object detection, ensuring that alerts are triggered promptly. Our system's integration with the Fast SMS Service enables immediate alert delivery to mobile phones, enhancing situational awareness and enabling quick responses to emergencies. Overall, our project contributes to the enhancement of safety and security measures, making environments safer for everyone.

The system's ability to detect multiple objects simultaneously and send alerts in real-time adds to its effectiveness in scenarios where quick action is essential, such as traffic monitoring or crowd surveillance. The seamless integration of the object detection system with the alerting mechanism ensures that potential threats are identified and addressed promptly, minimizing the risk of incidents or accidents. Additionally, the system's flexibility allows for easy customization and scalability, making it suitable for deployment in a wide range of environments and applications.

In conclusion, our project represents a significant step forward in leveraging deep learning and real-time technologies for enhancing safety and security. By providing a reliable and efficient means of detecting and responding to potential threats, our system contributes to creating safer and more secure environments for individuals and communities. The successful implementation and testing of our project demonstrate its potential for real-world applications and its capacity to make a positive impact on safety and security practices.

REFERENCES

[1]M Kristo et al., "Thermal Object Detection in Difficult Weather Conditions Using YOLO", 2020.

[2]M Haroon et al., "Multi Sized Object Detection Using Spaceborne Optical Imagery", IEEE, 2020.

[3]W Fang et al., "Tinier-YOLO: A Real-Time Object Detection Method for Constrained Environments", 2019.

[4]S Birogul et al., "YOLO Object Recognition Algorithm and ''Buy-Sell Decision'' Model Over 2DCandlestick Charts", 2020.

[5]W Wu et al., "Machine Vision Inspection of Electrical Connectors Based on Improved Yolo v3," 2020.

[6]Q Xu et al., "Research on Small Target Detection in Driving Scenarios Based on Improved Yolo Network," 2020.

[7]X Wang et al., "SO-YOLO Based WBC Detection with Fourier Ptychographic Microscopy," 2018.

[8]KJ Kim et al., "Multi-Scale Detector for Accurate Vehicle Detection in Traffic Surveillance Data," 2019.

[9]V Paidi et al., "Deep learning-based vehicle occupancy detection in an open parking lot using thermal camera," 2020.

[10]S Guennouni et al., "Multiple object detection using OpenCV on an embedded platform", 2014.

[11]M Kushal et al. "ID Card Detection with Facial Recognition using TensorFlow and OpenCV," 2020.

[12] S. Harini et al., "Object Detection Using Yolo and TensorFlow using Machine Learning", 2020.

[13]KEYOU GUO et al., "Real-Time Vehicle Object Detection Method Based on Multi-Scale Feature Fusion", 2019.

[14]Tanvir Ahmad et al., "Object Detection through Modified YOLO Neural Network", 2019.

[15]Satish Dekka, D ManendraSai, Mohammad Rafi, Maddala Rama Durga Apparao, Talachendri Suryam and Gatte Ravindranath "Machine Learning Techniques Based Prediction for Crops in Agriculture", 2023.

[16] Satish Dekka, Dr. K. Narasimha Raju, Dr. D. ManendraSai, Mr. Mohammad Rafi "Model to Detect Breast Cancer Based On Patient Symptoms", 2022.