YOLO Object Detection Using Deep Learning

Y.Chandhana[1], G Nagalakshmi Ratna Manikyamma [2], K Princy [3],R Anusha [4]

[1] Professor, [2, 3 & 4] Students

[1] chandana.nrtnec@gmail.com@gmail.com, [2] nagamanigunda1@gmail.com, [3] princysumanvitha@gmail.com, [4] anusha.rudrubati@gmail.com
Department of Computer Science and Engineering,
Narasaraopeta Engineering College ,Narasaraopet,Andhra Pradesh,India

ABSTRACT-In the discipline of computer vision, identifying many objects in scene images or localising a single object is a demanding and dynamic operation. Finding an object's location within an image and classifying it correctly are incredibly difficult problems. One of the pre-trained deep learning models is employed in this study to build an object recognition method. The objective is to use a webcam to take images of the things and identify them recognising items in a video feed and displaying the quantity of that specific thing. This procedure is specifically designed to be efficient for visualisation related applications. Additionally, by repeating these procedures for every frame of a video stream, real-time object recognition throughout the film is made possible In the realm of computer vision, identifying several objects in a scene image or locating a single object is a demanding and dynamic challenge.. Finding an object in a photograph and categorising it correctlyare very challenging undertakings. In this work, a pre-trained deep learning model is used to build an object recognition algorithm. The major goal is to recognise things by taking pictures with a webcam, identifying them in a video feed, and showing their number. This process is especially meant to be effective for applications using visualisation. Throughout the entire movie, real-time object detection is made possible by repeating this process for each frame of the video feeds.

KEYWORDS **- YOLO, Convolutional neural networks, Bounding box, Grid cells, Anchor box, Non-maximum, suppression, Feature extraction, Objectness scope.**

I. INTRODUCTION

AI's object detection technology locates and identifies objects in images, videos, and even live webcam feeds. It is used in many industries, such as security, healthcare, and autonomous driving. Among the most popular deep learning algorithms for object detection are YOLO, Faster RCNN, and SSD[1]. To use deep learning for object detection, you must first compile a collection of pictures or videos which took place annotated employing the things you want to detect. Following training, the model can be applied to recognise objects in newly taken images or videos. To use deep learning for object detection, you must first compile a dataset of images or videos that have been annotated with the objects you want to detect. following training. You can use a model to recognise objects in new images or movies [2]. Detecting objects with DL is becoming more and more common as the3se algorithms get more accurate and efficient [3].

By enabling the robots to comprehend and interact with the visual environment, it offers up a world of possibilities for the many applications across sectors [4]. Today, we propose a Python script that makes full use of the corresponding computer-vision capabilities by using real-time item recognition and object counting. It provides machines with the ability to comprehend and engage with their visual surroundings, opening up a wide range of opportunities in various industries [5]. We now propose a Python script that rapidly makes use of computer vision capabilities, particularly for counting and item recognition. The first step in utilising DL for object find ing is to create a set of images or videos labelled with the entities you want to detect [6]. The trained model can be used to recognise objects in new images and videos.

In this work, YOLO models were used for two purposes: (1) people detection and counting; and (2) area estimation of the inside space's defined area. We evaluate the populated zone of the place and use a suggested technique to determine the maximum population that can be accommodated in a given area. [7]. To detect the items, a variety of models and algorithm versions are employed. Performance assessments of the models were carried out for this proposed algorithm. Thus, when it came to identifying people within a given region [7], the variants 3, 4, and 5 of the models accuracy rates were, respectively, 96.89%, 96.12%, and 94.57%.

Owing to its increased layer count, YOLO v2 moves more slowly. Each residual block is composed of numerous residual units, however it is devoid of the remaining bricks found in YOLO v3. The residual networks, or ResNets, employ an alternative methodology. To extract features, the approach divides the network into convolutional layers with sizes of 1×1 and 3×3. And the 53 more layers for detection put the cherry on top of this. As deep learning algorithms progress, object detection with deep learning is becoming a powerful technique that is becoming more and more popular [5]. The necessity of the indicated model Identifying items in machine vision involves challenging and dynamic process. It's difficult to find objects in images or videos and classify them properly.

Usually, traditional methods cannot provide the precision and speed [2] required for real-time applications. It is obvious that considerably more accurate and effective object identification systems are needed, especially for applications

where items need to be swiftly detected and counted such traffic control, industrial automation.

The proposed model intends to address the growing demand for real time object detection. Numerous applications require the ability to recognise and count1certain items inside a live video stream.This could entail monitoring people, vehicles, or anything else of interest. The proposed method utilises the fast and accurate YOLO algorithms. Using a pre-trained DL model[8], it streamlines and improves object recognition for visualisation applications. This real-time monitoring [12] capability can be very beneficial in sectors like manufacturing and security where quick and precise item identification is crucial.

Automating and being precise must be the project's top priorities. The COCO dataset and a YOLO model that has already been trained are used to deliver an advanced object recognition solution [15]. On the perspectively, the object detection will identify and count. The dramatic visuals are displayed in accordance with the diagram, even though it will identify the things beforehand and using appropriate techniques. The graphic below illustrates how, in some cases, detection occurs.

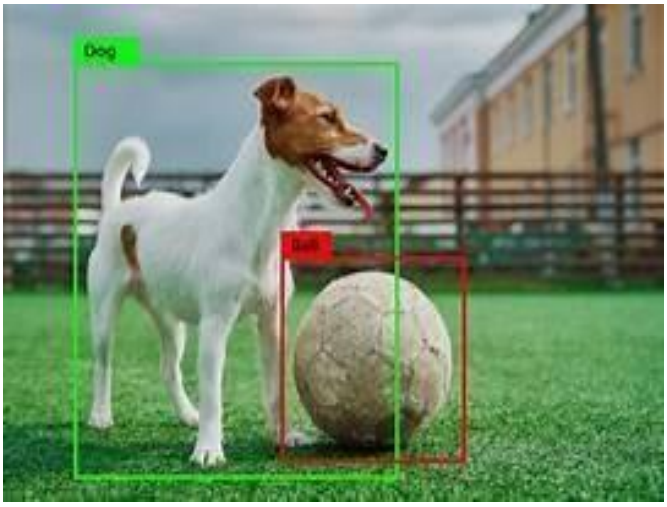

Fig. 1. Object Detection

Fig. 1 shows that the objects are detected using boundary boxes and are categorized the count of specific objects is displayed on the output window.

The suggested system implements real-time object detection in video streams by utilising Vid Gear, OpenCV, and cv lib. Vid Gear is used to handle video input efficiently, and cv lib helps with object detection in single frames. Bounding boxes are used by the system to outline objects as they are identified, providing a clear visual representation of their placements. Furthermore, a count of particular items is continuously tallied by the system according to their type, and this count is superimposed on the video frame. The processed video stream is displayed in full screen mode for the best possible user experience, guaranteeing that items that have been spotted are optimally visible.

Key components of the suggested work might be:

1. **Real-time Detection and Enhanced Accuracy**: To provide real-time responsiveness, the system provides fast object identification from a variety of video sources. The accuracy is greatly increased by using the object detection feature of CV Library, ensuring dependable outcomes. Bounding boxes are another tool used to emphasise identifiable objects and improve visual-clarity.

2. **Object Counting and Efficient Processing:** The system can accurately quantify certain objects that have been spotted, giving a clear count that may be used for monitoring or analytical purposes. This counting is made possible by Vid Gear's Cam Gear, which ensures low latency in video stream analysis through effective real-time processing.

3. **Flexibility and User-Friendly Display:** The system's design allows it to readily adapt to various sources and ambient conditions. Additionally, its easy-to-use interface guarantees that the processed video frames are both visually appealing and educational, which improves user understanding.

4. **User Interaction and Graceful Exit:** By utilising a fullscreen window display, the system guarantees that users are completely absorbed in the material and promotes an immersive experience. Furthermore, it has an integrated mechanism that guarantees system integrity and user control and enables users to end an application seamlessly and safely at any time.

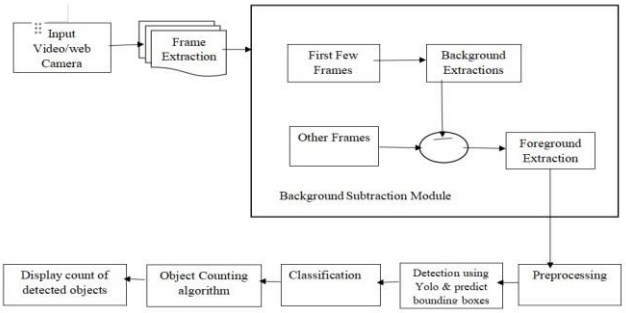

Fig. 2. Proposed system with YOLO implementation

The entire implementation process is shown above, based on Fig. 2.

Such studies aims to calculate The scope of a specific spatial unit seen in a clip. Equation (1) provides a formula for calculating the pixels that make up these zones. However, the area's true size is not represented by the size of the pixels because of factors including The quality of footage, viewpoint, and picture size. There should be square metres for actual size. Converting pixels to square metres should be done for this. One difficult difficulty is estimating the actual size (m2) from an image. In order to get around this, a reference item in the image of known size is used. A human

is the reference item in this investigation. One can estimate the extent of the region inhabited by these people if they known space used by the typical individual (Eq. 2).

Bounding boxes among the populace that the YOLO models identified serve as the basis for the methodology that this study suggests. For every individual in the picture that it recognises, YOLO models create a bounding box with a rectangular form. The model provides the bounding box height and width values in pixels. These rectangles' areas are computed using Eq. 2 and averaged A square to pixel conversion coefficient T metres is determined if the computed Measure of mean surface (px) is proportional to the square footage (m2) that a typical individual occupies. Any region's area in the image can be approximated using this coefficient.

Geographical factors affect data like shoulder width and height. For instance, anthropometric measurements taken in Turkey for a study [38] revealed that the average height of males and women was 1708 mm and 1598 mm, respectively. According to comparable survey, A typical hip breadth for men and women was 475 m m, 366 m m, respectively. For purposes of this study, an average person's area is 0.66 square metres (Eq.2).

The footage is examined to see if the people shown there are inside the designated boundaries. To approach 1m^{2}, The surface of every individual (equation,2) is extended using the widths (w) and heights (h) of the humans found in the area (Eq,3). Equation 5 estimates the square metre of the designated area and calculates the necessary population for this area. Fig. 5 displays the area computation algorithm for the designated region.

A=abs~x_{1}y_{2}-x_{2}y_{1}+x_{2}y_{3}
x_{3}y_{2}+...x_{n}y_{1}-x_{1}y_{n}\everytime- (1)

Equation 1 helps determine place of designated where in pixels, which comes out to be R. px. Bearings for each apex of line designated location are indicated by the and y values provided here. The approximate area of an individual can be determined using Eq. 2 if their vicinity is shown by depth and p_{1} level h. w. A person's surface is determined to be 0.66 m² using by the Equation-.2.

$$P_1 = h \times w \quad (2)$$

Since the threshold value is 1 person per square m, the extent of the designated region can be obtained in m2 suppose spot that oneself occupies exceeds increased to one m2 (Eqn. 3). To account for this, one must mix 3 to 5 of an individual's domain to their realm (Equation 3).

P2=p1+p1*3/5 (3)
p2=1m*2

Eq. 3 causes locatuon P2 to grow to about 1 m². Pixels are used to determine breadth as well as length that are people that're found within the designated area. encompassing box's mean the room (B) is computed using Eq 2. Equation 4, equivalent to the transformation in Equation 3, need to be utilised for this particular value (B_{x}). One must divide geographic size in pixels2 half the factor B_{X}. in order to determine the area of the designated zone in m^{2}.

Location of designated room is given in square metres by the R value that is produced in this manner. Additionally, the R_{m} value indicates that the region's human potential.

Time-wise, B_{x_{2}}=B_{x_{1}}
Let B_{x_{1}} = \frac{3}{5} (4)
R_{at}=R_{I}/B_{I} (5)

II. LITERATURE SURVEY

The literature review is the most crucial phase in the software development process. This will outline some initial study that was done on this relevant topic by a number of writers. We will also explore some significant articles and expand on our work.

Vishwanatha et al. [1], The paper gives object identification methods, with special attention to the YOLO and its variants. Evaluating and comprehending the variations and parallels between different iterations of YOLO and between YOLO and convolutional neural networks (CNNs) is the stated problem.

Akshara Guptha et al. [2], This study presents object detection and a method to identify things on a range of cars, as well as other modes of transportation and footing, when the object is located inside the provided input image.

Zhimin Mol et al. [3], This domain is associated with quality control and the manufacturing of automobiles. Improving the productivity and adaptability of the production process required real-time identification and detection of solder junctions in car door panels.

Rumin Zhang et al. [4], By integrating a light field camera with the YOLO object detection algorithm, this work probably solved the demand for obstacle detection, particularly in indoor contexts. Its goal was to identify and categorise the objects in the picture.

Wenbo Lan et al. [5], Pedestrian detection looks to be the domain, and the Yangon model's YOLO network is being used to improve this feature. The text you submitted does not provide specific details about the issue.

Glenn Jocher et al. [6], In contrast to its predecessors, YOLOv5 placed a strong emphasis on deployment, simplicity, and ease of use. It offered a simpler design based on the EfficientDet framework and strongly emphasised a modular and scalable training pipeline. Despite its simplicity, YOLOv5 performed better than comparable state-of-the-art detectors.

Chien-Yao Wang et al. [7], Taking into consideration the available computational resources, this research introduced Scaled-YOLOv4, a scaling technique that aims to establish a compromise between model size and accuracy. While utilising less resources, its performance was comparable to that of YOLOv4.

Zheqi He et al. [8], YOLO Nano was developed in response to the need for efficient and low-weight object identification models suitable for use on devices with constrained resources. It proposed an incredibly compact design with reasonable accuracy, making it suitable for edge computing applications.

These publications offer a thorough history of YOLO based object identification techniques, emphasising improvements in deployment appropriateness, accuracy, speed, and efficiency.

III. PROPOSED SYSTEM

The model proposed is based on the criteria as follows:

Dataset Analysis
Process Of Object Detection
Implementation
Result and Analysis

A. DatasetAnalysis

The dataset we have taken for making predictions is coco dataset [16]. A popular computer vision dataset called COCO (Common Objects in Context) includes photos with captions, segmentation masks, and object labels. Of course, this is a list of every one of the COCO dataset's eighty classes. The data set contains all the objects Which are used in the project . Popular items in Context, or Coco for short, is a massive understanding objects, slicing, and tagging collection that is commonly utilised for computer vision tasks

wine glass	broccoli	dining table	toaster
cup	carrot	toilet	sink
fork	hot dog	tv	refrigerator
knife	pizza	laptop	book
spoon	donut	mouse	clock
bowl	cake	remote	vase
banana	chair	keyboard	scissors
apple	couch	cell phone	teddy bear
sandwich	potted plant	microwave	hair drier
orange	bed	oven	toothbrush

TABLE I . COCO dataset

COCO dataset: Frequently Used Items in Setting, or Coco for short, is an immense object detection, categorising, and labelling set that is commonly utilised for computer vision tasks. It contains the images of everyday scenes with multiple objects annotated with bounding boxes, segmentation masks, and captions. It is widely used for training and evaluate algorithms in activities like division and recognition of objects, and captioning in the field of computer vision.

Each img is annotated with object labels, bounding boxes, and in some cases, segmentation masks. And the dataset is primarily applied towards the benchmarking, training deep learning models in all aspects of perception technology.

B. Process Of Object Detection

YOLO models often need input photographs that are a predetermined size. A grid of cells divides up the image.If an object's centre lies inside a particular cell, then that cell is in charge of anticipating the object. YOLO forecasts the envelops surrounding every unit block. A Level of assurance indicates likelihood providing each bounding boxes includes an item, and its measurements (x, y, thickness, and altitude) for all boundaries are used to represent the component.. For every bounding box, Yolo projects an objectnes score that indicates the probability of the square. having an item. For every bounding box, YOLO primarily forecasts the probability distribution among several classes along the bounding boxex. Absence of maximum attenuation is used to eliminate detected duplicates.

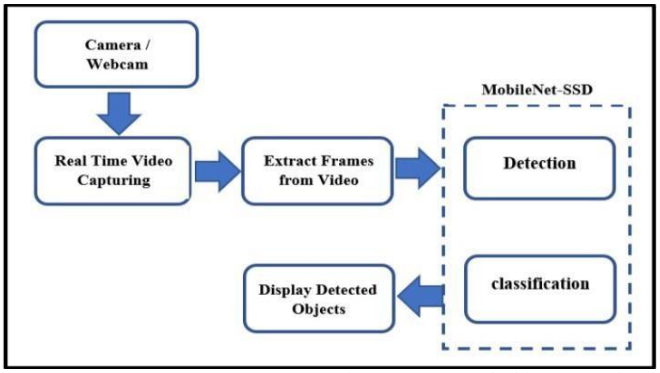

Fig. 3. Process Of Object Detection

Fig. 3 Tells that the object detection procedure. A spotting things is an automated perception technique includes recognising and categorising entities in depicts and movies. In order to identify things, the method usually entails taking a picture or video frame, extracting elements from the image, and then categorising those features. The specific procedure depicted in the graphic makes use of an object detection model called YOLO. It is a single-stage detector that forecasts defining boundaries and classes probabilities for every cell in the image by dividing it into a grid of cells. Duplicate detections are then eliminated via non-max suppression.

After taking into account the overlap and confidence scores of each bounding-box, it retains just the most reliable ones for each object. The maintained bounding boxes, together with the confidence ratings and accompanying class labels, make up the final result. YOLO can now recognise many objects using an isolated artificial brain in contemporaneity pass thanks to this method. Additionally, by optimising the efficiency of deep learning in object detection, these preprocessing steps aid in getting the dataset ready for

training. The particular method used may change based on the attributes.

C . Implementation

These are the main steps in implementing real-time object detection:

1. Input Video/web Camera: This component indicates the video stream's source, which can be either a physical camera recording in real-time or a web camera recording from a web source.

2. Frame Extraction: Frames are individual images taken from the video stream, comprising both the first few frames (for system initialization) and the other frames (for various processing purposes).

3.Foreground and Background Extraction in the Background Subtraction Module: One essential part of video processing systems is the Background Subtraction Module. It includes two essential procedures:

3.1 Background Extraction: Locating and preserving the video's motionless backdrop.

3.2 Foreground Extraction: This technique separates the video's dynamic items from the background by isolating them.

4.Preprocessing: The foreground frames are subjected to various tasks like as picture improvement, noise reduction, and other approaches aimed at improving identifier performance.

5. Applying YOLO & Predict Bounding Boxes: Preprocessed frames undergo determination of objects by the use of YOLO, a deep learning model. In order to find the detected items, bounding boxes are formed around them. frame = draw_bbox (frame, bbox, label, conf) detect_common_objects(frame) = cv. bbox, label, conf

6. Classification: The observed items are categorised into specified classes or categories in this stage. Identifying between automobiles, pedestrians, and other things, for instance.

7. Object Counting Algorithm: To determine how many particular objects of interest are present in each frame, an algorithm is used. Counting the number of cars in a traffic scene or persons in a crowd are two examples of this.

```
object_counts={} for obj in label:
 object_counts[obj] = object_counts.get (obj, 0) + 1
```

8. Display Count of Specific items: The system's ultimate output is the count of specific items, which may be seen and utilised for a number of purposes, including management, crowd control, and traffic monitoring. text = ', '.join([f"{obj}={count}" for obj, count in object_counts. items()])

```
cv2.putText(frame, text, (50, 60),
cv2.FONT_HERSHEY_PLAIN, 3, (255, 0, 0), 3)
cv2.namedWindow("FRAME",
cv2.WND_PROP_FULLSCREEN)

cv2.setWindowProperty("FRAME",cv2.WND_PROP_FULLS
CREENcv2.WINDOW_FULLSCREEN)
        cv2.imshow("FRAME", frame)
```

Yolo Object detection code:

This Python script uses computer vision methods to recognise objects in video streams in real time. It makes use of CamGear to record frames from a camera source and the OpenCV and cvlib libraries for object detection. The application reads frames constantly, resizes them to a stand ard screen resolution, and uses the detect_common_objects function from cvlib to identify objects. Counts and labels are applied to objects that are detected, and the results are shown on the frame. In fullscreen mode, the processed frame remains visible until the user hits the 'Esc' key to end the programme.

D. Result and Analysis

YOLO divides the estimates bounding boxes and class probabilities for every grid cell using an input image. Using this method, YOLO may identify several objects in a single run. through the neural network, making it faster compared to other algorithms that use sliding windows or region proposals.

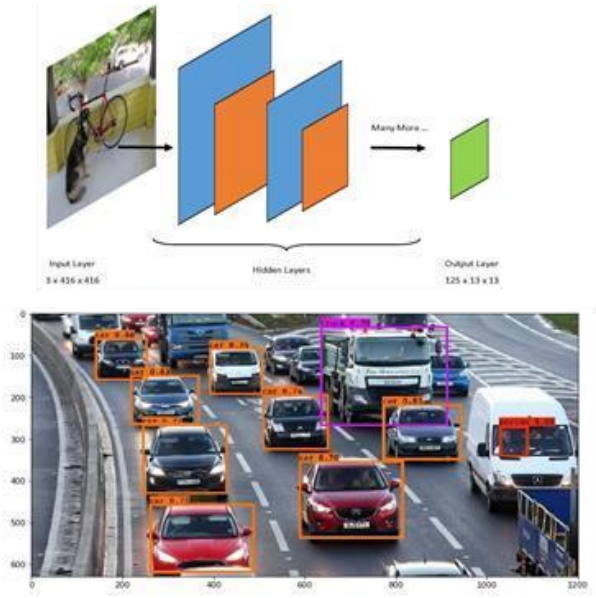

Fig. 4. Sample Det

In the Fig. 4 tells the coloured boxes around objects in the image are bounding boxes and forms a Grid for the Perspective Objects .

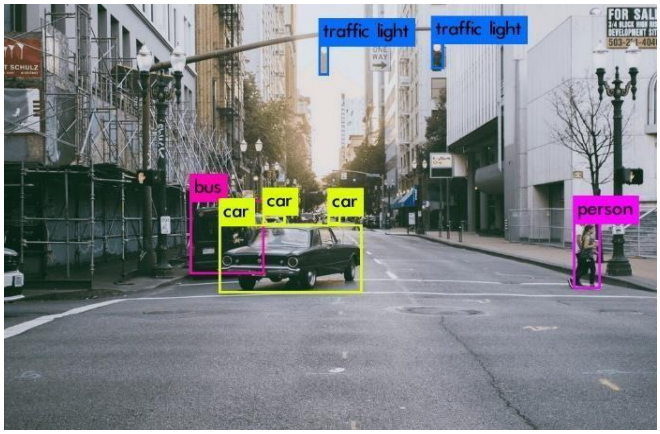

Fig. 5. Objects Detection in traffic

Fig. 5 Showing the detects cars and people in traffic jam scene. This is a type of object detection detection and localize specific objects within an image.

Fig. 6 Detection With Confidence

From the Fig . 6 you can typically visualize the detected objects by drawing bounding boxes around them in the input image or video frames. Additionally, you can display labels indicating the class of each detected object along with confidence scores. This score reflects how certain the model is about the classification within the bounding box

Fig. 7. Count Of the Each Object

Fig. 7 tells us about the count of the each object by using grids. And the coloured boxes around the object in the image are bounding boxes. These are mainly indicate the location of the detected object.

These are likely the numbered shapes mentioned in the top right corner of the image. Bounding boxes are the common way to indicates the locations of the detected objects in object detection tasks.The confidence scores describes the scores reflects how certain the model is about the classification for that the particular object.The task is to identify and count object in an image, detected objects and their confidence score.

IV. CONCLUSION

Compared to previous models, Yolo's algorithm reduces the amount of time needed to recognise objects in web and video cameras, according to trial data. Additionally, a variety of uses in a broad range of applications, including traffic management, retail and marketing, security and surveillance, and many more, benefit from our function of recognising and showing the count of objects. You look just once has shown to have a lot of potential for real-time item recognition across a range of applications, offering insightful information and facilitating wise decision-making. Overall, object identification in films and web cameras using the yolo model is a powerful tool for recognising objects, and with continuous research and development, it is expected to become even more successful and useful in the future. Ultimately, the speed and accuracy of detection have risen. The accuracy of the current system is 90.86%, and the accuracy of our proposed model is 96.12%.

V. FUTURE SCOPE

Object detection systems are becoming more and more necessary as mobile robots and autonomous machines in general (such as drones, quadcopters, and soon service robots) start to be used more frequently. Lastly, we must take into account the requirement for object detecting systems for nano-robots. The CCTV camera should have an integrated night vision mode for nighttime visual tracking. The system can be encouraged to be utilised for object detection in underground mining factories if the climate is suitable for it. identifying complicated goods for security and safety reasons, such as guns and weapons .

VI. VREFERENCES

[1] Tausif Diwan, G. Anirudh ,Jitendra V. Tembhurne (2023) “Object detection using YOLO: challenges, architectural successors, datasets and applications”. link.springer.com Multimedia Tools and Applications **82**, 9243–9275. doi.org/10.1007/s11042-022-13644-y

[2] Vijayakumar, A., Vairavasundaram,(2024) “YOLO-based Object Detection Models: A Review and its Applications.” Multimed Tools Appl (2024). doi.org/10.1007/s11042-024-18872-y

[3] N. M. Krishna, R. Y. Reddy, M. S. C. Reddy, K. P. Madhav and G. Sudham (2021) "Object Detection and Tracking Using Yolo," Third International Conference on Inventive Research in Computing Applications (ICIRCA), Coimbatore, India, pp. 1-7, doi: 10.1109/ICIRCA51532. 2021. 9544598.

[4] Sirisha, U., Praveen, S.P., Srinivasu, P.N. et al(2023) " Statistical Analysis of Design Aspects of Various YOLO Based Deep Learning Models for Object Detection. Int J Comput Intell Syst **16**, 126 doi.org/10.1007/s44196-02300302-w.

[5] Katyayani, K. Bhardwaj and T. Poongodi (2023) "Deep Learning Approach for Multi-Object Detection Using Yolo Algorithm," 6th International Conference on Contemporary Computing and Informatics *(IC3I)*, Gautam Buddha Nagar, India, 2023, pp. 689-693, doi: 10.1109/IC3I59117.2023. 10398124.

[6] C. Liu, Y. Tao, J. Liang, K. Li and Y. Chen (2018) "Object Detection Based on YOLO Network," IEEE 4th Information Technology and Mechatronics Engineering Conference (ITOEC), Chongqing, China,pp. 799-803, doi: 10.1109/ ITOEC.2018.8740604.

[7].Vidyavani , A. , Dheeraj , K., Rama Mohan Reddy , M., Kumar, K.N. (2019). "Object detection method is based on YOLOv3 using deep learning networks", International Journal of Innovative Technology and Exploring Engineering, api.semanticscholar.org/CorpusID:212538578

[8]. Bhattacharya S, Maddikunta PKR, Pham QV, Gadekallu TR,Chowdhary CL, Alazab M, Piran MJ (2021) "Deep learning and medical image processing for coronavirus (COVID-19) pandemic: a survey" Sustain Cities Soc. 2021 Feb;65:102589. doi: 10.1016/j.scs.2020.102589.

[9] Yoshua Bengio, Aaron Courville, and Pascal Vincent (2012) "Unsupervised feature learning and deep learning: a review and new perspectives." CoRR, abs/1206.5538, 1(2665)

[10] Dr. Suwarna Gothane (2021) "A Practice for Object Detection Using YOLO Algorithm" www.ijsrcseit.com "International Journal of Scientific Research in Computer Science, Engineering and Information TechnologyISSN: 2456-3307 ,India, doi.org/10.32628/CSEIT217249

[11] Rohini Goel, Avinash Sharma, and Rajiv Kapoor (2019) "Object Recognition Using Deep Learning" Journal of Computational and Theoretical Nanoscience 16(9):4044-4052 DOI:10.1166/jctn.2019.8291

[12] Hossain S, Lee DJ (2019) "Deep learning-based realtime multiple object detection and tracking from a aerial imagery via a flying robot with GPU-based embedded devices. Sensors" 19(15):3371

[13] Upulie H.D.I, Lakshini Kuganandamurthy (2021) "RealTime Object Detection Using YOLO: A Review" DOI:10.13140/RG.2.2.24367.66723

[14] Albelwi S, Mahmood A (2017) "A framework for designing the architecture of deep the convolutional neural networks."Entropy 19 (6):242

[15] Gavali P, Banu JS (2019) "Deep convolutional neural network for image classification on CUDA platform. In: e Deep learning and parallel computing environment for bio engineering systems."

[16] DatasetLink https://www.kaggle.com/code/rahulkumarpatro/yoloobjectdetection

.