Intelligent Video Surveillance using YOLO Object Detection

Prajwal Bhoyar[1], Aditya Butley[1], Sumedh Dhole[1], Nishad Joshi[1], Prof. Dr. Aniruddha Rumale[2]

[1] UG Students, [2]Professor, Computer Engineering Department, G H Raisoni College of Engineering and Management, Pune

ABSTRACT

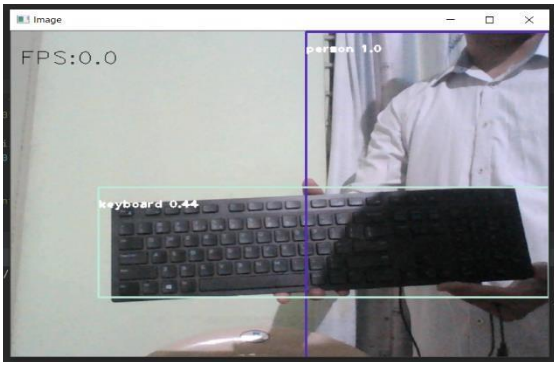
The aim of this intelligent surveillance system is to detect, track and classify objects detected. The intelligent video surveillance allows segregation of all the instances present in the image or a video according to their classes and their respective positions. A class of an instance in an image is the identification of any entity and correlate it with the real world it might be a living or non-living thing or a specific part of any instances in an image that can be uniquely identified such as, car on a road or mobile phone on desk. Also, along with the identification of class of an entity the respective position of an instance in an image or video is also an important aspect while extracting information from the image let us consider that a tiger is chasing a boar or a boar is chasing a tiger in this example the respective position of tiger and boar has a huge impact to ones understanding about the situation. An approach towards intelligent surveillance system along with object detection is proposed in this paper

Keywords - Image processing, CNN, Deep learning, YOLO. Object Detection, Surveillance

I. INTRODUCTION

In Computer Vision Object detection is an advanced technological field. It is a computer vision technique for locating instances of objects within images, videos or live video feeds and also recognizing the objects or items.

Implementation of object detection have numerous uses wherein few implementations are as follows. It can be utilized for automation of video surveillance, Anomaly detection in a production process, to help specially-abled people or for detection of diseases in their primary stages and so on.



Unlike image recognition object detection not only identify the item or an object in a given image but also intelligently draws bounding boxes around the identified items, which allows us to locate the position of where said objects or items are in a given frame or a scene. Object detection is often confused with image recognition so we need to get an understanding of their distinctiveness. Image Recognition effectively reward a label to an image. Example a picture of car is labelled as a car, a picture of two cars is also labelled as a 'car'. Whereas object detection draws a so called bounding box around those particular cars. Things get complicated when there are more than one object in a single frame or scene, in that case image recognition fails to label all the items, in contrast object detection only needs to draw multiple bounding boxes around the items. Moreover object detection decides which labels to assign to an individual item in an image. Hence

Object detection gives more information about an image than recognition.

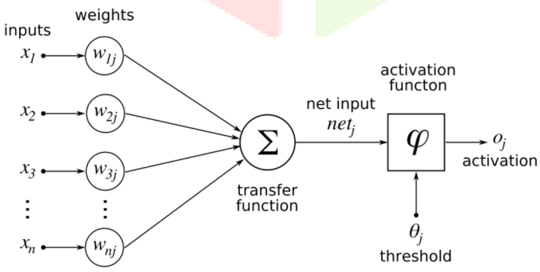
Object Detection is divided into types based on their approach and those are deep learning – based approaches and Machine Learning – based approaches. In traditional Machine Learning approach, computer vision techniques are used to look at the properties and features of an image i.e. edges, pixel density, color, etc. and further using regression model with help of the features extracted the location of an item or an object along with its label is detected. Whereas in deep learning based approaches Convolutional Neural Networks (CNNs) are used too perform end-to-end, unsupervised object detection, in which an individual features are not extracted separately.

In this Project of ours we use deep learning approach to identify objects in an image and video. Convolutional Neural Network is a deep learning algorithm which takes an image as an input, assigns weights and biases to that image. These biases and weights are used as learnable parameter of a learning model. Neurons are considered as a basic unit of the deep learning neural network. Biases are constants that are used as an input into next layer. And weights on the other hand are used to check the influence of the input on the output.

$$Y = \sum (weight * input) + bias$$

A neuron.

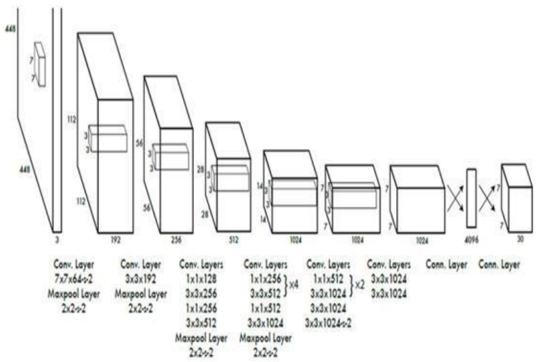
[1] Convolutional neural networks are composed of multiple layers of artificial neurons. Artificial neurons, a rough imitation of their biological counterparts, are mathematical functions that calculate the weighted sum of multiple inputs and outputs an activation value.



The structure of an artificial neuron, the basic component of artificial neural networks (source: Wikipedia)[2].

Intelligent Video Surveillance System uses YOLO object detection Algorithm. YOLO stands for You Only Look Once. It is a Simple Convolutional Neural Network approach which offers high speed as well as better results, with help of YOLO's high speed processing object detection is made possible with great efficacy. Darknet is an open source neural network framework written in C and CUDA which is readily available on GitHub. Darknet framework is used for as a CNN configuration. It is fast, easy to install, and supports CPU and GPU computations. It comes under the category of Artificial Neural Network Software. Darknet has various features like ImageNet Classification, RNNs in Darknet, DarkGo, TinyDarknet and one of them is YOLO. It requires any of two dependencies that are OpenCV for wider variety of supported types of images or CUDA if GPU computation is needed. But the intelligent video surveillance system does not uses GPU computation yet delivers a good object detection efficacy.

II. ARCHITECTURE OF THE YOLO OBJECT DETECTION ALGORITHM



[11] This architecture of YOLO algorithm has 24 convolutional layers followed by 2 fully connected layers. Alternating 1 × 1 convolutional layer reduces the feature spaces from preceding layers. The convolutional layers are pre-trained on the ImageNet classification task at half the resolution which is 224 × 224 input image and then double the resolution for detection. [12] We optimize in the output of our model for the sum-squared error. Reason for using the sum-squared error is that it is easy to optimize, however it does not perfectly align with our goal of maximizing average precision. Localization error is weighted equally by it with classification error which might not be ideal. Also, many grid cells do not contain any object in every image. This pushes the "confidence" scores of those cells that do contain objects towards zero, often overpowering the gradient from cells. This can lead to model instability, which causes training to diverge early on. To overcome this, we need to increase the loss from bounding box coordinate predictions and decrease the loss for boxes that don't contain objects from confidence predictions.

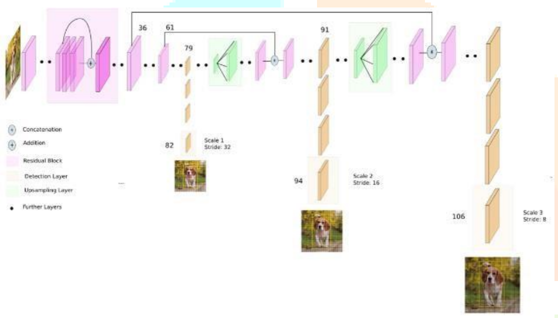
[13]. SSD was proposed by W. Liu et al. in 2015 which was the second one-stage detector in deep learning era. Introduction of the multireference and multi-resolution detection techniques (to be introduce in Section 2.3.2),

which significantly improves the detection accuracy of a one-stage detector, especially for some small objects are the main contribution of SSD.

YOLOv3 is the most alteration of the You Only Look Once (YOLO) approaches. The most well-known computer vision tasks are image classification, aiming at assigning each image in a dataset to one of the two or more categories.

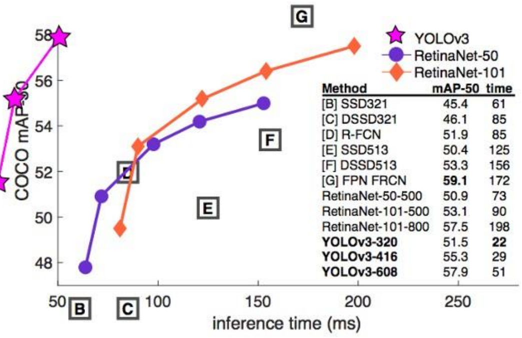
If you not only want to make use of detecting the presence of an object, but also of locating where this object is within the picture, you can use of the method called as "object localization".

Now, the most complex, and interesting process of the three is object detection (leaving out object segmentation). In most real-life examples, we want our model to go beyond recognizing and locating only one object, and detect multiple objects from the given same image. Object detection does just this operation and draws a so-called bounding box around each individual object.



The most current of the main versions is the third iteration of the approach, namely YOLOv3. In each version, improvements have been made over the previous one. The initial version proposed the general architecture, after which the second variation improved accuracy significantly while making it faster. YOLOv3 refined the design further by using tricks, such as multi-scale prediction and bounding box prediction through the use of logistic regression. While the accuracy increased dramatically with this version, it traded off against speed which reduced from 45 to 30 frames per second.

As shown in the graph below, YOLOv3 achieves the best speed-accuracy trade-off on the MS COCO dataset, a large-scale object detection dataset. With a mean average precision (mAP) of 57.9% in 51ms, the YOLOv3 exceeds the RetinaNet-101 by 57.5% in 198ms. The YOLOv3 is thus almost four times faster with an equal mAP.



III. LITERATURE SURVEY

A lot of research is going on the computer vision field. Upon researching the current knowledge including substantive finding and theoretical contribution in the field of object detection using the YOLO Algorithm we got some information, et al.[3] That a fast and simple approach of detecting real-time images was introduced in this paper as You Only Look Once. The model was built to identify images more efficiently and accurately and to successfully differentiate between real and fake images. Chegji Liu et al. [4] A generalized Object Detection network was developed by using and applying complicated degradation processes on training sets like noise, rotation, cropping, and blurring of the images. The model trained were resulting in better ability and high robustness. [5]. The YOLO Algorithm's network structure was improved and a new structure was proposed to increase the ability to retrieve information form of the pedestrian features, which were treated as objects, by adding pass-through layers to the actual YOLO Network. [6]. YOLO Object Detection algorithm and the light field camera were combined to classify objects and highlight them in the image. [7]. The current best version of YOLO is pretty fast. At 320x320 YOLOv3 runs in 22 ms at 28.2 mAP. YOLOv3 is accurate as SSD but three times faster. As compared to the old .5 IOU mAP detection metric, YOLOv3 is quite good.

IV. PROPOSED SYSTEM

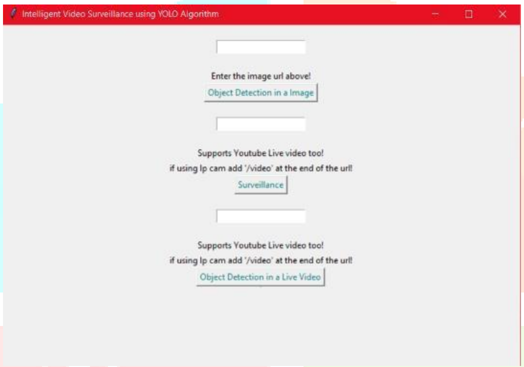
The surveillance system can identify objects via live video stream over a URL, in an image or an IP webcam stream. It also offers surveillance features as motion detection in the scene.

OpenCV is a huge open-source library for computer vision, image recognition, video and image processing and machine learning it plays an important role in today's object detection system. Being an open-source library OpenCV has a large base of keeping it upated is used in large scale. Hence the proposed system used opencv at it fullest.

Run the Python '.exe' file to start the GUI.

(The GUI is developed using Tkinter Toolkit.)

The Dialogue box greets with three Blank options which allows user to enter either an url of the image or a video, or IP webcam address of a streaming camera connected to a same WiFi network and has functionalities of detecting objects in a video or in an image and a surveillance mode. To capture video using webcam user is needed to enter '0' in the blank space or the 'ip webcam' link also user can keep the blank space 'as it is' before choosing any of the option to experience the prototype or an example of how the system works. Youtube videos can also be used or live streamed over the system all user need is to enter the video url. The image support is also vast ex: jpg, jpeg, png, jiff, etc are readily supported by the system if provided with exact url.



Based on a given type of input you will be able to detect or surveillance. This allows user to choose different options based on their needs.

[8]. COCO Dataset is used in this system which is a large-scale object detection, captioning and segmentation dataset with various features as follows recognition in context, super pixel stuff segmentation and around 1.5 million object instances. It also feature 25000 people with key points. coco.names files is used in this system which is responsible for playing a crucial role in detecting around 80 classes of object. It is one of the most challenging object detection datasets currently available today. Microsoft-COCO-17 contains 164 thousand images and 897 thousand annotated objects from 80 categories. On top of the bounding box annotations, Using per-instance segmentation to aid in precise localization each object is further labeled. Also, MS-COCO contains more small objects and more densely located objects than VOC and ILSVRC. All these features make the object distribution in COCO closer to those of the real world. Just like ImageNet in its time, MSCOCO has become the de facto standard for the object detection community.

V. TECHNOLOGIES USED

Python: Python is an interpreted high-level general-purpose programming language. Python's design philosophy emphasizes code readability with its notable use of significant indentation. [Source: Wikipedia]

Tkinter: Tkinter is a Python binding to the Tk GUI toolkit. It is the standard Python interface to the Tk GUI toolkit and is Python's de-facto standard GUI. Tkinter is included with standard Linux, Microsoft Windows and Mac OS X installs of Python. The name Tkinter comes from the Tk interface. [Source: Wikipedia]

VI. SOFTWARE AND HARDWARE REQUIREMENT

Hardware requirements:

a. High Processing Power is needed. Intel i5+ 8th generation or AMD Ryzen 3 or above
b. For computational resources, GDDR5 Ram Graphics card is needed least 4GB Software requirements:

Python 3.7 +

Libraries: Tkinter, Matplotlib, opencv, numpy, PIL

VII. APPLICATIONS

Few examples of the Applications are Given Below:

- Video Surveillance.
- Crowd monitoring. Anomaly detection.
- Multiple Object Observation.
- Tracking Object.
- Pedestrian Detection. Alerting System. Automated CCTV Object Counting.
-

VIII. RESULT AND DISCUSSION

Upon completion of the training and loading of the Convolutional neural network using yolov3 weights; videos and images were used to test the efficacy and to check how well GUI created performs. AS expected images were perfectly operated on and each object in the scene was independently detected with more than 90% of an average prediction.

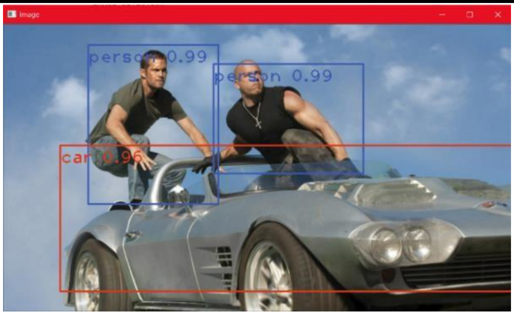
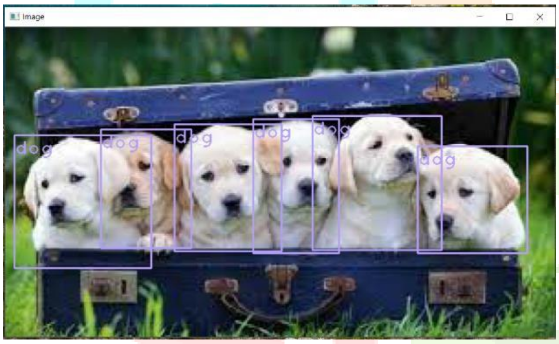


Image from the movie Fast and Furious

Predictions are used using yolov3 and not yolov3-tiny weights

In case of object detection in video, please visit the link below to check the actual sample - https://github.com/Nishad-Raj/Intelligent-Video-Surveillanceusing-YOLO-object-detection/tree/main/demo The model was able to detect objects in a live video as well as already upload/loaded video with good accuracy score. The FPS was a bit on the lower end but can be increased depending on the hardware on which the model is running. Yolov3-tiny weights and yolov3-tiny cfg was used for video processing for increasing the FPS and output rate only in object detection in videos



Surveillance on an IP camera streaming as well as pre-loaded video resulted in detecting movement with perfect bounding boxes around the object. As shown in the below figure. Surveillance resulted in displaying a message of movement detection with as expected fps of max around 60 FPS.

YOLO has good performance on VOC 2007 and its AP degrades less than other methods when applied to artwork. Like DPM, YOLO models the size and shape of objects, as well as relationships between objects and where objects commonly appear. Artwork and natural images are very different on a pixel level but they are similar in terms of the size and shape of objects, thus YOLO can still predict good bounding boxes and detections. Our combined Fast R-CNN + YOLO model is one of the highest performing detection methods. Fast RCNN gets a 2.3% improvement from the combination with YOLO, boosting it 5 spots upon the public leader board.

IX. CONCLUSION

Intelligent video surveillance using the YOLO object detection algorithm attained exceptional performance. Once models were trained on the available data. The system was able to perform with higher accuracy and while identifying objects and classes as well detecting motion based on optimized algorithms that were used. The neural networks and algorithms used performed well with confound predictions. GUI created allowed user to use features more efficiently and practically and each module was working efficiently and increased overall user experience overall.

References

[1]. Ben Dickson. (2020) What are Convolutional Neural Networks. https://www.experfy.com/blog/ai-ml/what-areconvolutional-neural-networks-cnn/

[2]. The structure of an artificial neuron, the basic component of artificial neural networks.
Source: Wikipedia.

[3]. Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi(2016). You Only Look Once: Unified, Real-Time Object Detection.
arXiv:1506.02640v5 [cs.CV]

[4] Chengji Liu, Yufan Tao, Jiawei Liang, Kai Li1, Yihang Chen Object (2018). Detection Based on YOLO Network

[5] Wenbo Lan, Jianwu Dang, Yangping Wang, Song Wang (2018). Pedestrian Detection Based on YOLO Network Model

[6] Rumin Zhang, Yifeng Yang (2018). An Algorithm for Obstacle Detection based on YOLO and Light Filed Camera

[7] Joseph Redmon, Ali Farhadi (2018). YOLOv3: An Incremental Improvement, arXiv:1804.02767

[8] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, Piotr Dollár (2015) Microsoft COCO: Common Objects in Context arXiv:1405.0312v3 [cs.CV]

[9] Manish Chablani (2017**)** YOLO — You only look once, real time object detection explained.

Source: https://towardsdatascience.com/yoloyou-only-look-once-real-time-objectdetection-explained-492dc9230006

[10] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, “Ssd: Single shot multibox detector,” in European conference on computer vision. Springer, 2016, pp. 21–37

[11] YOLO Real-Time Object Detection
Source: https://pjreddie.com/darknet/yolo/

[12] Bochkovskiy, Alexey (2020). "Yolov4: Optimal Speed and Accuracy of Object Detection". arXiv:2004.10934 [cs.CV]

[13] Geethapriya. S, N. Duraimurugan, S.P. Chokkalingam (2019). “Real-Time Object Detection with Yolo”. International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-8, Issue3S, [5]

[14] Zhong-Qiu Zhao, Member, IEEE, Peng Zheng, Shou-tao Xu, and Xindong Wu, Fellow, IEEE. (2019). “Object Detection with Deep Learning: A Review”. arXiv:1807.05511v2 [cs.CV]

[15] Jiuxiang Gua, ∗ , Zhenhua Wangb, ∗ , Jason Kuenb , Lianyang Mab , Amir Shahroudyb , Bing Shuaib , Ting Liub , Xingxing Wangb , Li Wangb , Gang Wangb , Jianfei Caic , Tsuhan Chenc(2017). “Recent Advances in Convolutional Neural Networks” arXiv:1512.07108v6 [cs.CV].