iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 12 **Issue:** V **Month of publication:** May 2024

DOI: https://doi.org/10.22214/ijraset.2024.61808

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 12 Issue V May 2024- Available at www.ijraset.com

Advanced method for Image detection using OpenCV

A.V. Sumanth Kumar
Department of CSE, SRM Institute of Science & Technology, Chennai

Abstract: ***In computer vision, object detection is an essential job with many real-world applications, such as robots, autonomous cars, and surveillance. Because of its great accuracy and speed, the YOLO (You Only Look Once) method is a well-liked real-time object recognition technique that has attracted a lot of attention. This technique is perfect for time-sensitive applications since it examines the full image at once and predicts bounding boxes and class probabilities for recognized items. YOLO has undergone many iterations, with YOLO v5 being the most recent and sophisticated version. It utilizes anchor boxes and a feature pyramid network (FPN) to increase the accuracy of object recognition. Our goal in this research is to apply YOLO v5 to real-time picture and object identification applications stage. The model will be trained using an appropriate dataset, and its performance will be assessed against a range of benchmarks and advanced object detection methods. The project's output will offer a reliable and effective real-time object detection system that will facilitate prompt decision-making in the identification of item types and their corresponding placements. It has useful uses in robotics, autonomous driving, and surveillance.***
Keywords: Image Detection, OpenCV, Computer Vision, YOLO (You Only Look Once), Neural Networks.

I. INTRODUCTION

Real-time object detection and picture identification are two crucial challenges in computer vision. While real-time object detection entails detecting and tracking things in real-time video feeds, image detection refers to the process of finding inside an image. These two tasks are critical in many domains, including as robots, autonomous cars, surveillance systems, and many more. The widely used YOLO algorithm is one method for object detection. YOLO is a real-time object detection system that uses a single neural network to predict the likelihood of object classes and bounding boxes in pictures or videos. In this work, we will introduce YOLO and talk about how it may be used for real-time object recognition and picture detection.

Sliding windows are the primary tool used by traditional object identification algorithms to find things in images. Nevertheless, this approach can be computationally hard and resource-intensive, particularly when dealing with huge photos or video streams. By using a single neural network to produce predictions for every data object in a picture at the same time, YOLO, in contrast, is much quicker than conventional techniques. The YOLO method predicts the likelihood of object classes and bounding boxes for each cell in an image by first splitting it into a grid of cells.

Real-time object detection algorithms and picture detection have significantly improved as a result of recent developments in deep learning. These techniques often employ convolutional neural networks (CNNs) to extract characteristics from pictures or video frames, which are then used to identify and locate objects in the scene.

In order to identify objects in a video stream as fast and precisely as possible, real-time object identification algorithms are made to operate in real-time.

All things considered, real-time object and picture identification are crucial components of many computer vision systems with a wide range of useful applications. It is anticipated that greater study in this area will lead to improvements in accuracy, speed, and efficiency, creating more sophisticated computer vision systems that can be used in a variety of situations.

II. METHODOLOGY

Finding and identifying objects of interest in an image or video is the challenge of object detection in computer vision. Numerous methods can be employed in the development of an object detecting system.

A. Approach 1 Using Yolo Algorithm

Steps:

1) Install the necessary prerequisites and libraries. You'll need PyTorch, an open-source machine learning toolkit, and Torchvision, a PyTorch library that provides a ton of pre-trained models, architectures, and datasets for computer vision applications.

©IJRASET: All Rights are Reserved

2) Preprocessing: To prepare the dataset, resize the images and normalize the pixel values. Additionally, this step includes data augmentation techniques like flipping, rotating, and adding noise to increase the dataset's unpredictability.
3) Train the Yolo model with the pre-processed dataset. The Yolo algorithm, a contemporary object recognition system, uses a self-attention mechanism to record long-range correlations between visual properties. You can use the pre-trained YOLO models that are available in Torchvision as a starting point for the training.
4) Testing: Apply the trained YOLO model to an alternative picture dataset to evaluate its performance. Use assessment metrics like as precision, recall, and F1 score to evaluate the model's accuracy.

We find that the Yolo Algorithm outperforms all other approaches in terms of performance speed and accuracy.

B. Approach 2 Using coco dataset and OpenCV

Steps:

1) Preparing the dataset: Get the COCO dataset by downloading it. It includes thousands of photos with annotations for items of different types. After deciding which object classes to identify, extract the photos and annotations for each class to generate a custom dataset.
2) Install OpenCV: Install the dependencies for OpenCV, a well-known computer vision library. For completing object identification tasks, OpenCV offers pre-trained models and routines.
3) Preprocessing: Adjust picture dimensions and standardize pixel values to prepare the dataset. In order to improve the variety of the dataset, this stage also includes data augmentation techniques like flipping, rotating, and introducing noise.
4) Training: Combine the custom dataset with previously taught OpenCV models to create a deep learning model.
5) Testing: Utilize an alternative photo dataset to evaluate the performance of the trained model. Assessing the model's accuracy may be done with measures like F1 score, precision, and recall.
6) If the object identification system's performance isn't up to par, you may fine-tune it by changing the model's hyperparameters or training it on fresh data.

As we can see, the coco model only has 80 classes and a reasonable accuracy rate. Since it cannot identify objects outside of its class, we must move on to the next strategy.

C. Approach 3 Using caffe model dataset and OpenCV

Steps:

1) Dataset preparation: Download caffe dataset from GitHub.
2) Install Caffe and OpenCV: Caffe is a deep learning framework for object recognition and picture classification, while OpenCV is a well-known computer vision library.
3) Preprocessing: To prepare the dataset, resize the images and normalize the pixel values. This step also includes data augmentation techniques like flipping, rotating, and adding noise to increase the dataset's unpredictability.
4) Training: Train a deep learning model with the pre-processed dataset with Caffe.
5) Testing: Apply the trained model to another batch of image data to evaluate its performance. Use evaluation metrics such as the F1 score, recall, and precision to determine the model's accuracy.

It is evident that the accuracy of the Caffe model is low, with only a few classes.

III. RESULTS AND DISCUSSIONS

A. Using COCO Dataset:



Fig-1: Initial UI Design of the DATA set.



Fig-2: UI Design of the DATA set when image is Uploaded



Fig-3: Results of uploaded image.

B. Using YOLO Algorithm:

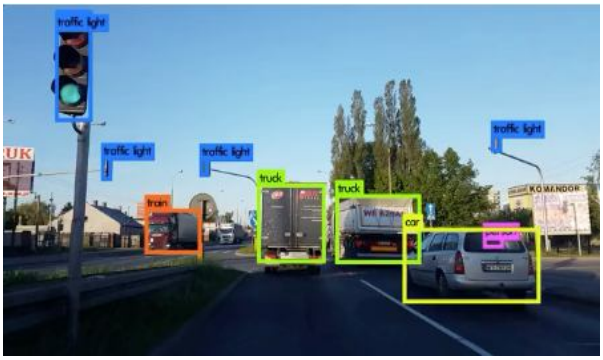


Fig-4: Results of processed image



Fig-4: Results of processed image

IV. CONCLUSIONS

Object detection is an essential part of computer technology, with applications ranging from robots and autonomous cars to security and surveillance. Deep learning-based methods have led to a notable boost in object identification speed and accuracy in recent times. Using the COCO dataset and the YOLO technique, we developed an object detection system. which are widely used object identification methods based on deep learning. In real time, YOLO is able to swiftly and precisely recognize a variety of items. The COCO dataset is a large object detection dataset. including more than 2.5 million object instances dispersed over more than 330,000 images and 80 categories. Our findings show that the COCO dataset in conjunction with the YOLO technique is an effective tool for item detection. We were able to quickly and accurately identify a wide range of things in a variety of settings. The YOLO technique is appropriate for applications that need real-time object identification, such robots and autonomous cars, since it can identify many things at once.

V. FUTURE SCOPE

Although object detection has advanced significantly in the last several years, there is still more space for development. Future research may concentrate on enhancing the precision and resilience of object detection systems. Even while algorithms like YOLO have produced amazing outcomes, they are not flawless and can constantly be improved. Scholars may explore novel methods for object detection, including employing reinforcement learning or attention processes.

The creation of more representative and varied datasets may be a further focus of future research. The generalizability of models trained on object detection datasets may be limited because to the bias present in many of these datasets, which favour particular item kinds or scenarios. Scholars should endeavour to create more varied datasets that more accurately represent the variety of items and situations seen in practical applications. This might entail utilizing synthetic data to produce more diversified train sets or gathering data from a wider range of sources.

VI. ACKNOWLEDGMENT

I would like to thank Faculty of Department of CSE, SRM Institute of Science & Technology, Chennai for their support in completing and publishing this paper.

REFERENCES

[1] Rushikesh Lakhotiya, Mayuresh Chavan, Satwik Divate, Soham Pande. “Image Detection and Real Time Object Detection”: International Journal for Research in Applied Science & Engineering Technology (IJRASET) – 2023. Pg: 2785 - 2790.

[2] K. Vaishnavi, G. Pranay Reddy, T. Balaram Reddy, N. Ch. Srimannarayana Iyengar and Subhani Shaik. “Real-time Object Detection Using Deep Learning”: Journal of Advances in Mathematics and Computer Science – 2023. Pg: 24 - 32.

[3] Ayoub Benali Amjoud; Mustapha Amrouch. “Object Detection Using Deep Learning, CNNs and Vision Transformers: A Review” IEEE Access (Volume:11) - 2023. Pg: 35479 – 35516.

[4] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, ‘‘Object detection with discriminatively trained part-based models,’’ IEEE Trans. Pattern Anal. Mach. Intell., vol. 32, no. 9, pp. 1627–1645.

[5] C. Nobata, J. Tetreault, A. Thomas, Y. Mehdad, and Y. Chang, ‘‘Abusive language detection in online user content,’’ in Proc. 25th Int. Conf. World Wide Web, Montreal, QC, Canada, Apr. 2016, pp. 145–153.

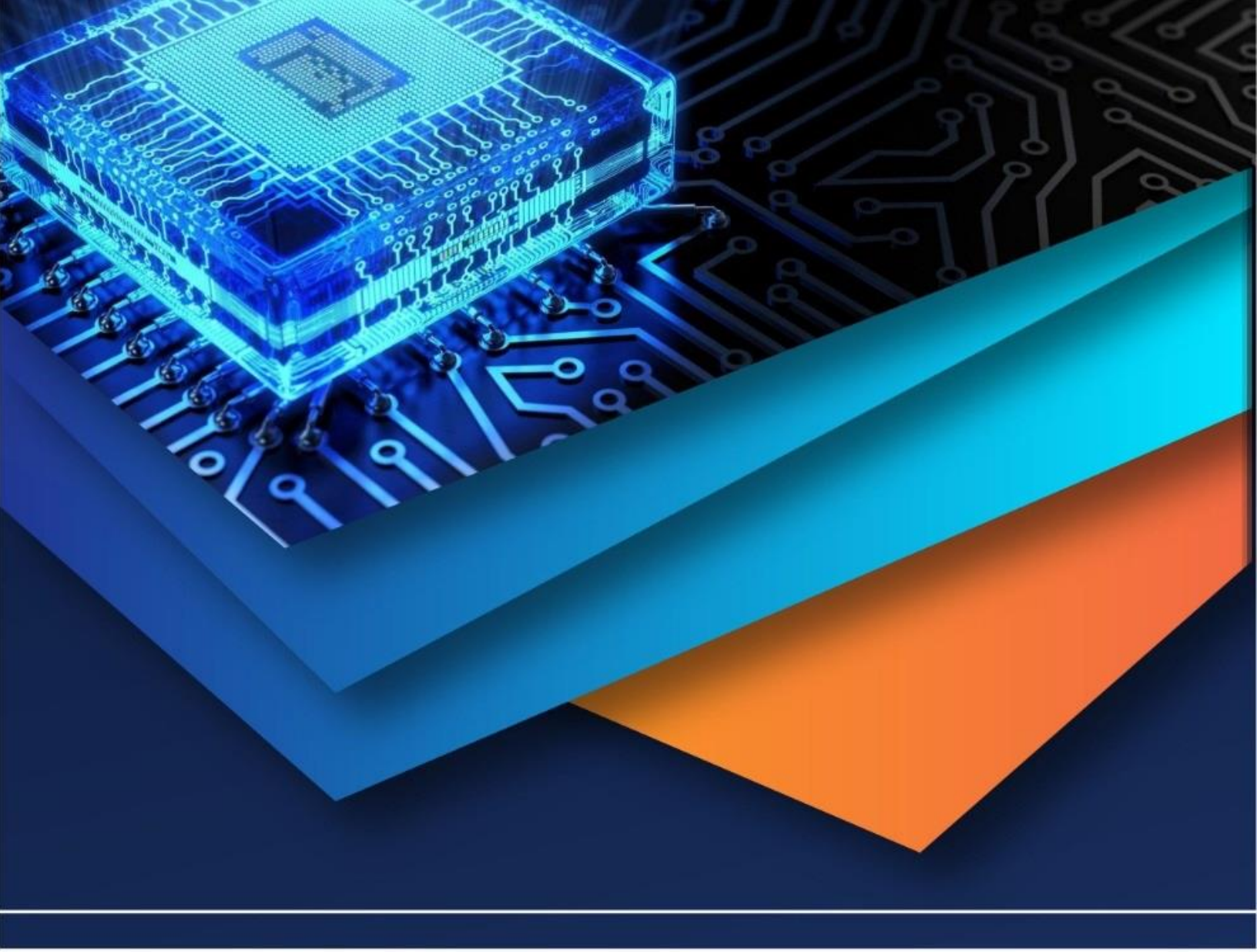
[6] A. B. Nassif, M. A. Talib, Q. Nasir, Y. Afadar, and O. Elgendy, ‘‘Breast cancer detection using artificial intelligence techniques: A systematic literature review,’’ Artif. Intell. Med., vol. 127, May 2022, Art. no. 102276.

[7] N. Dalal and B. Triggs, ‘‘Histograms of oriented gradients for human detection,’’ in Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR), San Diego, CA, USA, Jun. 2005, pp. 886–893.

[8] S. Agarwal and D. Roth, ‘‘Learning a sparse representation for object detection,’’ in Computer Vision—ECCV 2002, A. Heyden, G. Sparr, M. Nielsen, and P. Johansen, Eds. Berlin, Germany: Springer, 2002, pp. 113–127.

[9] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, ‘‘You only look once: Unified, real-time object detection,’’ in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Las Vegas, NV, USA, Jun. 2016, pp. 779–788.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)