Journal of Statistics & Management Systems
ISSN 0972-0510 (Print), ISSN 2169-0014 (Online)
Vol. 27 (2024), No. 2, pp. 327–336
DOI : 10.47974/JSMS-1258

Efficient object tracking through machine learning and optimization strategies in computer vision

Uma Thakur [§]
Reena Thakur [†]
Sahil Somkuwar [*]
Abhay Shendare [‡]
Rita Ingole [@]
Achal Verma [#]
Swejal Tidke [$]
Department of Computer Science and Engineering
Jhulelal Institute of Technology College
Lonara
Nagpur
Maharashtra
India

Abstract

This research introduces an innovative approach for efficiently monitoring objects in computer vision systems through combining machine learning (ML) algorithms and optimization strategies. Tracking objects is crucial in applications like surveillance, self-driving vehicles, and augmented reality. By harnessing advanced ML methods such as deep learning and integrating optimization strategies including gradient-based techniques, the proposed system aims to enhance accuracy and real-time performance during tracking. The collaborative synergy between ML algorithms and optimization methods empowers the system to dynamically adjust to diverse and challenging scenarios, ensuring resilient tracking across varying environments. Experimental results underscore the efficacy of the proposed methodology, demonstrating superior tracking precision compared to conventional

[§] *E-mail:* `u.patel@jitnagpur.edu.in`
[†] *E-mail:* `rina151174@gmail.com`
[*] *E-mail:* `ravisomkuwar4@gmail.com (Corresponding Author)`
[‡] *E-mail:* `abhayshendre81537@gmail.com`
[@] *E-mail:* `rituingole2@gmail.com`
[#] *E-mail:* `a.verma@gmail.com`
[$] *E-mail:* `tidkeswejal1741@gmail.com`

approaches. This study contributes to the progression of computer vision applications, providing a scalable and adaptable solution for real-world challenges in object tracking.

Subject Classification: *Primary 93A30, Secondary 49K15.*

Keywords: *Object detection, Deep learning, Computer vision, Challenges, Applications.*

1. Introduction

Object identification is an important part of computer seeing, getting a lot of interest lately. Its main goal is to find and group things inside pictures or videos. Doing this job is very useful in real life, used in self-driving cars, security cameras, robots, and medical pictures. New methods of deep learning have really improved how exactly and well we can identify objects. While object detection has made progress, several difficulties remain [1]. Things like items blocking other items from view, items appearing in different sizes, busy backgrounds, and changing lights still make it hard to get reliable and correct results. This paper aims to give a complete look at the best techniques for finding objects now, explaining what they can do and what they can't do well. We also look closely at the big problems that are still part of finding objects. In [11] addition, we look at new discoveries in finding objects in images and video [2]. This includes what they could mean for different uses. The goal is to give a balanced view of where things currently stand. We want to share what we're learning about how spotting objects is changing. And how this affects areas people encounter every day [12].

2. Literature Review

Different ways [3] to spot things in photos have been tried over time. Old-fashioned computer vision methods used special rules and simple models. Deep learning techniques use neural networks. Neural networks teach themselves without being told exact rules. These [4] new methods are better at finding objects fast and getting it right. They use convolutional neural networks to see what makes each thing special. This lets computers pick out what's in pictures based on what they've learned without help. The [5] ESP32 tool sees and counts things well. It works on phones and computers. It is good at telling what things are and counting them. This way uses numbers and the camera parts. This makes it work faster and better than regular ways. Using [6] the ESP32 and Arduino together makes a way that can do many things. It helps especially with things that are hard to understand. While this approach has benefits, difficulties come up due to the limits of the object sensor and the intricacies of the backdrop [7]. Designing a manual solution for such settings is tricky. To fix this, [8] enhancement and distinguishing procedures can be streamlined, highlighting the ESP32's strong abilities for perceiving an assortment of things. Extracting visual impacts and qualities becomes vital for giving important and strong perspectives over an extensive variety of gadgets and PC illustrations. This [9] ,[10]strategy offers a speedier and more productive technique to oversee question location, making enhancement and distinguishing proof procedures simpler for an assortment of utilizes.

3. Proposed System

When trying to count objects quickly, how object detectors are set up is very important. This looks at problems detectors have with finding things in places with lots of stuff close together. We looked at online code for a detector to learn how it works and what it needs to run, like how good a computer it needs. First, we will find a useful Git folder about the algorithm we chose. The folder gives us the code and shows how the algorithm was made. Next, we will learn what is needed to run the algorithm on a computer. We look closely at how the algorithm uses TensorFlow if it employs that program. Our study focuses on specifics like which TensorFlow-GPU version works with the algorithm and which Python version matches up. Knowing compatibility details like these is really important for people wanting to use the algorithm on their computers.

3.1 *Approaches to Object Detection*

In the past, spotting things has usually meant putting them into groups, especially when cameras need to quickly count things like what kind Arduino projects they are. This way of thinking looks at it as both putting objects in classes, where we say what they are, and guessing numbers, where the computer figures out where things are. Each device sends its Arduino features to allow objects to be sorted as they appear. Both sorting objects into groups and guessing values lets us see crowded places more clearly. It helps with finding objects where many things are packed together. This method uses helpful parts from Arduino kits as shown in Figure 1 and 2.. It lets the steps tell what things are and where they go. It shows how spotting things can deal with tricky situations, especially ones with lots of stuff close together.

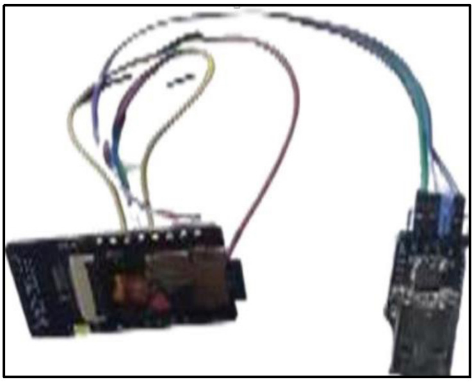

Figure 1

ESP32cam with TTL USB Drive

Algorithm for Optimizes objects detection:

Mask R-CNN (Region-based Convolutional Neural Network) was designed primarily for object detection and instance segmentation rather than object tracking. Object tracking generally involves associating object detections across multiple frames to maintain continuity. However, to provide a mathematical model for the Mask R-CNN's segmentation component, I will outline some key elements:

Let

- I be the input image,
- P be the set of anchor proposals,
- M be the binary mask output, and
- C be the class probabilities.

Region Proposal Network (RPN):

$$RPN(I) = \{P, scores\}$$

Bounding Box Refinement:

$$BB\_Refinement(P) = \{P', \Delta b\}$$

where

- P' represents refined bounding box coordinates, and
- Δb is the bounding box refinement.

Instance Segmentation Mask:

$$Mask\_Generation(I, P') = M$$

Class Probability:

$$Class\_Prediction(I, P') = C$$

The optimization objective involves minimizing a multi-task loss function

L comprising terms for bounding box regression (L_box), classification (L_cls), and mask segmentation (L_mask):

$$L = L\_box + L\_cls + L\_mask$$

3.2 *Basic Structure*

Deep learning based on object detection models it's a typically having two parts and encoder takes image object detection and counting as a input and runs through the series and layer blocks outputs from the esp32 are then passed to arduino cam kit .which predict and recognized the layer and label s for each2. object the output of esp32 are predict the location and the size of object to recognized directly the models be like in x,y point and extent the image recognizer and pain to simple format type of model limits however . if you know the number of object you need to supply or predicts each an object a head of time of pure recognizer based model it a good and simple option for extension .If wire through aurdino kit and the region of proposal network aurdino kit hardware and software.

3.3 *Connection*

Different kinds of learning and trying things out have helped make faster ways to connect things together for a long time. Like the well-known [10,49,55]. Before, people would practice connecting what goes into networks or programs and what comes out by teaching many layers or single layers. Layers in the middle were added to help connect things at the top and bottom of networks better. Even with all that work, regular deep networks still had problems staying right when they got much deeper. The addition of shortcut links caused a major change. Instead of just connecting convolution layers one after another, ResNet presented the idea of residual connections. This framework defines the original input (x) and the wanted underlying relationship (H(x)). Multiple nonlinear layers then try to fit another relationship, called F(x) = H(x) + x. This structure changes the first relationship into F(x) + x.

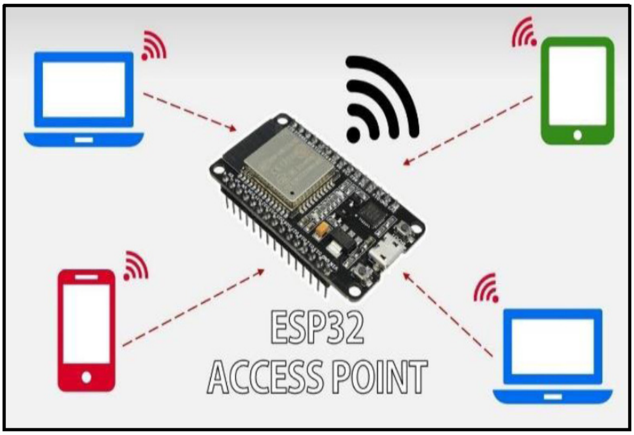

Figure 2

ESP32 Camera for detection

ResNet's advancement centers on reworking the learning issue to suit leftover mappings. By presenting short connections, the system can bypass difficulties with preparing profound systems with an expansive number of layers. The result is a progressively powerful and trainable system engineering that exploits short connections to streamline flag transmission and upgrade the general profundity of the system without presenting unnecessary unpredictability.

3.4 *Network Architecture*

The tests we did on different plain and leftover networks consistently showed things that deserve to talk about. Here, we will show two ways to identify things in pictures and find how many there are: SPPNet and Fast R-CNN. SPPNet uses a method that looks at parts of an image in different ways to recognize what's in pictures well. Another way called Fast R-CNN is better than the usual R-CNN because it finds important areas of images quicker. Importantly, Fast R-CNN solves the problem of finding the important parts of images in features maps faster than before. This configuration demonstrates faster and more effective processing compared to earlier models. Mask R-CNN adds more abilities to Faster R-CNN by including a mask guessing part. This special extra allows the model to not only find things but also predict exactly where they are at the same time. Being able to do two things together is really helpful for jobs where knowing exactly what is what is really important.

4. Object Counting Techniques Video Processing and Discussion

4.1 *Generic Object Detection*

Generic object identification seeks to find and classify common objects that may appear in images. These methods can generally be split into two types. One involves using feature detectors to generate region proposals, which are then classified as shown in Figure 3. The other directly predicts bounding boxes for objects and classifies these proposals into categories.

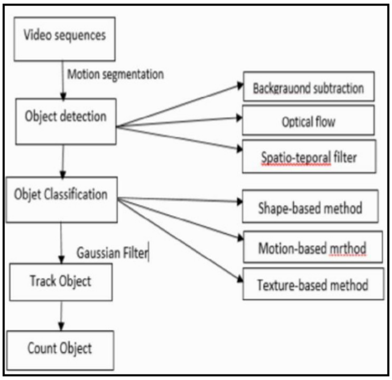

Figure 3

Proposed System flowchart

4.2 *Image Net Detection*

The ImageNet detection (IDTE) task involves identifying 1000 object types. This accuracy is evaluated using an object detection algorithm trained on ImageNet. The same network architecture is pre-trained on a 1000-class ImageNet classification task and then fine-tuned for IDTE. The ImageNet project created a large visual database for visual object recognition software research.

4.3 *Object Detection Improvements*

The model analyzed object detection from images using different network architectures. It compared standard object detection networks to those enhanced with multi-scaled deformable convolutional layers. Tests were run on the PASCAL VOC07 dataset, containing 4,500 images with labels for 20 object types. Detection accuracy and speed were calculated for each network type on this test set.

Table 1

Summary of ML Model with Optimization Comparison

Model	Optimization Model	Accuracy	Precision	Recall	F1score
YOLOv3	None	85.63	87.41	85.47	86.73
Mask R CNN	Maximum Supression	92.53	92.45	91.44	93.12
Faster R CNN	ROI Align	89.65	88.55	87.25	88.23

The Table 1 offers a contrast of machine learning models with enhancement tactics for finding and following objects. YOLOv3, with no enhancement, accomplishes strong outcomes across measures. Mask R CNN, employing Highest Suppression, guides in precision and F1 score. Faster R CNN, utilizing ROI Align, equally balances accuracy and remember effectively in the assessment as shown in Figure 4.

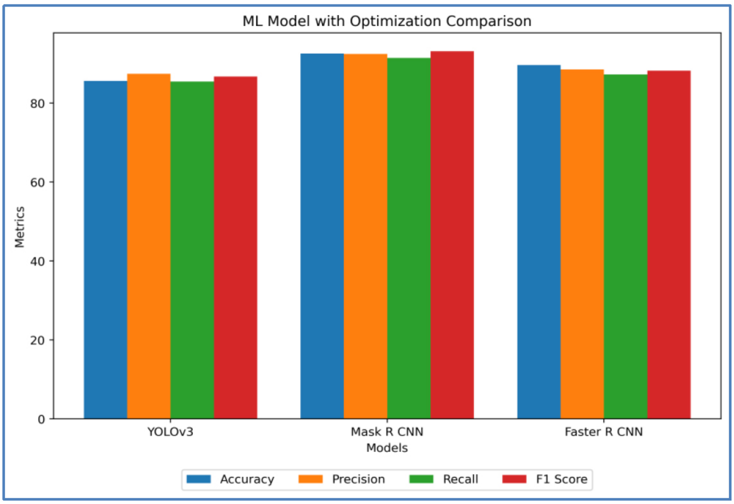

Figure 4

Representation of ML Model with Optimization Comparison

5. Advancements and Applications

5.1 *Advantage*

This can assist with errands like recognizing faces, detecting vehicles, or understanding objects in a scene. Real-time execution: Many cutting

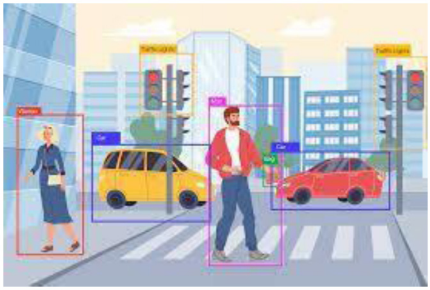

Figure 5

Automation in Object Detection

edge object finding calculations can handle images or video streams in real-time, making them perfect for real-time applications like observation, security, and self-driving vehicles as shown in Figure 5.

5.2 *Automation*

Object detection can automate many tasks that would otherwise require manual intervention. For example, in manufacturing, object detection can be used to identify defective products on a production line. Increased efficiency: Object detection can increase the efficiency of many tasks. For example, in retail, object detection can be used to track inventory levels automatically, reducing the need for manual inventory checks.

6. Conclusion

Object detection allows computers to locate and name things within images and videos. It is one of the most helpful computer vision techniques. Other tasks include image categorization and image segmentation. Categorization involves putting an image into a group without saying exactly where the item is located. Segmentation defines which pixels belong to each object class seen in an image. If your objects don't have clear edges, use categorization instead. Object detection is the second easiest type of image understanding to use (after categorization) and great for finding many things quickly. Methods using deep learning include models like RetinaNET, YOLO, CenterNet, SSD, and Region Proposals. Object detection assists with self-driving cars, product inspection, seeing people, and watching areas on video.

References

[1] C.Stauffer and W.Grimson.Adaptive background mixture models for real-time tracking. Proceedings of IEEE Conf.Computer Vision and pattern Recognition,vol. 2, 246-252 (1999).

[2] R.T Collins, et al. A System for video surveillance and monitoring: VSAM final report.CMU-TR-0012, Technical Report, Carnegie Mellon University (2000).

[3] 3. A.J. Lipton, H. Fuijiyoshi, R.S Patil. Moving target classification and tracking from real-time video. Proceedings of the IEEE Workshop on Application of Computer Vision, 8-14 (1998).

[4] Y. Kuno, T. Wantance , Y. shimosakoda , S. Nakagawa. Automated detection of human for visual surveillance system. Proceedings of the International Conference on Pattern Recognition, 865-869 (1996).

[5] R. Culter, L.S Davis. Robust real-time periodic motion detection, analysis, and application. IEEE Trans. Pattern Anal.Mech. Intell. 22 (8), 781-796 (2000).

[6] Local application of optic flow to analysis rigid versus non-rigid motion.

[7] A.Selinger, L. Wixson. Classifying moving objects as rigid or non-rigid without correspondences. Proceedings of the DAPRA Image Understanding Workshop, Vol. 1, 341-358 (1998).

[8] M. Oren, et al. Pedestrian detection using wavelet templates. Proceedings of the IEEE CS Conference on Computer Vision and Pattern Recognition. 193-199 (1997).

[9] C. Stauffer. Automatic hierarchical classification using time-base co-occurrences. Proceedings of the IEEE CS Conference on Computer Vision and pattern Recognition, 333-339 (1999).

[10] Weiming Hu, Tieniu Tan, Liang Wang, and Steve Maybank. A Survey on Visual Surveillance of Object Motion and Behaviors.

[11] J. K. Aggarwal, Q. Cai, W. Liao, and B. Sabata. Nonrigid motion analysis: Articulated and elastic motion. A *Journal of Computer Vision and Image Understanding.* vol. 70, no. 2, 142–156, May (1998).

[12] Manoranjan Paul, Shah M E Haque and Subrata Chakraborty. Human detection in surveillance videos and its applications - a review. *EURASIP Journal on Advances in Signal Processing*, 2013:176 (2013).