International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

A Review on Video Based Vehicle Detection and Tracking using Image Processing

Suruchi Kumari, Deepti Agrawal

Department of Electronics and Communication Engineering, School of Research and Technology
Peoples University Bhopal, Madhya Pradesh, INDIA
geetasuruchi@gmail.com, er.deeptiagrawal@gmail.com

ABSTRACT

Intelligent vehicle systems and traffic management are becoming increasingly important in highway management today. As diverse cars seen on the street on a regular basis, the number of these vehicles is rapidly expanding. With such a large number of cars, identification and monitoring have become problematic, particularly in emerging and poor nations. With this in mind, this article discusses and addresses a review of identifying and tracking automobiles from video frames. Deep learning has been widely used in the field of object detection and tracking. This paper examines video-based vehicle detection and tracking.

Keywords:Video based detection, Image Processing, SSD.

1. Introduction

Intelligent transportation systems are crucial in today's climate for traffic management in order to establish an efficient and dependable transportation system. The precise identification and monitoring of vehicles is one use of the intelligent transportation system. Smart detection and tracking systems need the gathering of processed data from appropriate regulatory techniques. In this regard, security cameras have recently been installed in traffic monitoring and control. Image processing algorithms are frequently used to track the movement of cars, people, and other things. An example of an advanced cautioning or data extraction application for real-time vehicle analysis is video processing of traffic data acquired from pre-recorded footage.Traditional vehicle systems, on the other hand, may decline and fail to recognised vehicles because they are obscured by other vehicles or background obstacles such as road signals, trees, and weather conditions, and the performance of these systems is dependent on good traffic image analysis approaches to detect, track, and classify the vehicles. We examined and evaluated previous works in this subject, defined the scope of the study, grasped the process and methods used, and eventually suggested a model that may assist us in the accurate identification and tracking of cars.

Image

An image is a collection of square pixels (picture components) organised in columns and rows. An image is made up of picture elements, also known as pixels, each of which has a finite, discrete amount of numeric representation for its intensity or grey level, which is an output of its two-dimensional functions fed as input by its spatial coordinates denoted with x, y on the x-axis and y-axis, respectively. (See Figure 1).

Each picture element in a (8-bit) greyscale image has an intensity value ranging from 0 to 255. A grey scale image is similar to a black and white image, but the term emphasises that it will also feature various shades of grey. Each pixel has a value ranging from 0 (black) to 255 (white) (white). The available range of pixel values is determined by the image's colour depth, which in this case is 8 bit = 256 tones or greyscales. (See Figure 2.) A true-color image created by combining three greyscale images in red, green, and blue. An picture of this type can have up to 16 million distinct colours. (See Figure 3.)

Some greyscale pictures have more greyscales than others, for example, 16 bit = 65536 greyscales. In theory, three greyscale photos may be joined to create a single image having 281,474,976,710,656 greyscales.

* *Corresponding author.* Tel.:+919304214651;
E-mail address: geetasuruchi@gmail.com

Figure 1 An image - picture elements arranged in columns and rows

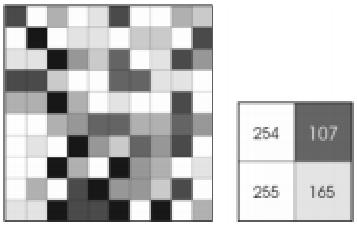


Figure 2 Greyscale image

Figure 3A true-color image

Vector graphics (or line art) and bitmaps (pixel-based or 'images') are the two broad categories of 'images.' The following are some of the most frequent file formats:

- **GIF:** A non-destructively compressed bitmap format with 8 bits (256 colors). Mostly used on the web. The animated GIF is one of numerous sub-standards.
- **JPEG:** A destructively compressed 24 bit (16 million color) bitmap format that is highly efficient (i.e. contains a lot of information per byte). Widely used, particularly for online and Internet applications (bandwidth-limited).
- **TIFF:** The standard publishing bitmap format of 24 bits. Lempel-Ziv-Welch (LZW) compression, for example, compresses nondestructively.
- **PS:** Postscript is a common vector format. There are several sub-standards that might make it difficult to move between platforms and operating systems.
- **PSD:** A Photoshop format that saves all of the information in a picture, including all of the layers.

2. Image Processing

Image processing seeks to convert an image into digital form and apply some procedure to it in order to obtain a better image or extract useful information from it. It is a technology that is being developed to transform images into digital form and execute various operations on them in order to obtain particular models or extract important information from them. This technique takes as input a video segment or a picture, such as a photograph. The output matches to the intended or attention-grabbing portion of the image. Digital image processing techniques aid in the alteration of digital pictures through the use of computers. Pre-processing, augmentation and presentation, and information extraction are the three main processes that all sorts of data must go through when employing digital techniques. (Source: Figure 4).

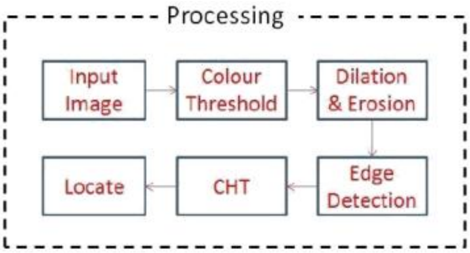


Figure 4 Basic Structure of Image Processing

Threshold

Thresholding is the most basic approach of picture segmentation in digital image processing. Thresholding may be used to produce binary pictures from grayscale photographs.

Dilation

Dilation adds pixels to the edges of objects in a picture, whereas erosion removes pixels from the edges of things. The state of every given pixel in the output picture is defined by applying a rule to the relevant pixel and its neighbours in the input image during the dilation and erosion procedures. The procedure is defined as a dilation or erosion by the rule used to process the pixels. The value of the output pixel in a dilation operation is the largest value of all pixels in the vicinity. A pixel in a binary picture is set to 1 if any of its neighbours contain the value 1. The value of the output pixel in an erosion process is the least value of all pixels in the vicinity. A pixel in a binary picture is set to 0 if any of its neighbours contain the value 0.

Edge Detection

Edge Detection is a technique for segmenting a picture into discontinuous parts. It is a popular approach in digital image processing applications such as pattern recognition, picture morphology, and feature extraction. Edge detection allows users to examine picture features for substantial changes in grey level. This texture marks the end of one section of the picture and the start of another. It decreases the quantity of data in a picture while preserving its structural features.

CHT

The circle Hough Transform (CHT) is a fundamental feature extraction approach in digital image processing that is used to find circles in defective pictures.

Locate

Identifying the position of one or more items in a picture and creating a bounding box around their extent is referred to as object localization. Object detection integrates these two objectives by locating and classifying one or more items in a picture.

3. Object Detection

Object detection is a computer vision approach for identifying and locating items in an image or video (From Figure 5). Object detection may be used to count items in a scene, determine and monitor their precise positions, and precisely label them using this type of identification and localization. Specifically, object detection creates bounding boxes around identified things, allowing us to locate objects in a given scene.

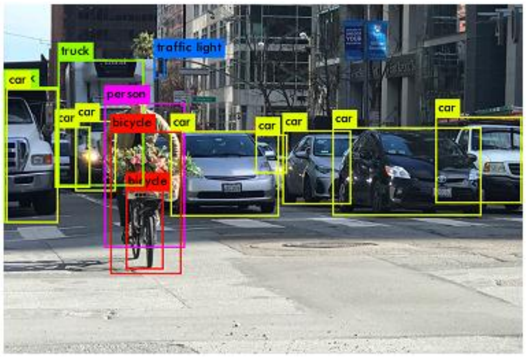


Figure 5 Object Detection

Object detection is related to other computer vision methods such as image recognition and image segmentation in that it assists us in understanding and analysing situations in photos or video. However, there are significant distinctions. Image recognition merely generates a class label for a detected object, but image segmentation generates a pixel-level comprehension of the parts in a picture. Object detection is distinguished from the other tasks by its ability to locate items inside an image or video.

3.1 Object Detection Work

Object identification may be divided into machine learning- and deep learning-based techniques. Computer vision algorithms are employed in more classic ML-based systems to examine at various aspects of an image, such as the colour histogram or edges, to identify clusters of pixels that may correspond to an object. These attributes are then entered into a regression model, which predicts the object's position as well as its label. Deep learning-based techniques, on the other hand, use convolutional neural networks (CNNs) to do end-to-end, supervised object recognition, which eliminates the need for characteristics to be created and extracted independently.

Object identification and tracking remains a significant challenge today. The complexity level of this challenge is significantly dependent on how the item to be discovered and tracked is defined.

Illumination Changes

Illumination has a considerable influence on the look of the backdrop and can lead to false positive detections. This should be factored into the backdrop model.

Dynamic Background

Some components of the scenery may have movement (a fountain, cloud motions, swinging tree branches, waves of water, etc.), yet they should be considered background based on their significance. This movement might be regular or erratic (e.g., traffic lights, waving trees). Managing such background dynamics is a difficult task.

Occlusion

Occlusion may affect the process of computing the background frame.

Speed of the Moving Objects and Intermittent Object Motion

If the object moves slowly, the temporal differencing approach will fail to detect the areas of the object that maintain a uniform region. A extremely fast moving item, on the other hand, leaves a trail of ghost area behind it in the detected foreground mask. Detecting and monitoring objects that move then halt for a brief period of time before moving again is tough.

Presence of Shadows

Foreground object shadows frequently complicate subsequent processing steps after background subtraction.

Challenging Weather

When movies are shot in adverse weather circumstances (winter weather conditions, i.e., snow storm, snow on the ground, fog), air turbulence, and so on), detecting moving objects becomes extremely difficult.

3.2 Deep Learning in Object Detection

We have concentrated our effort on deep learning algorithms since they have become the state-of-the-art approaches to object detection. Neural networks have evolved into cutting-edge technologies for object identification. Deep learning-based object identification models are usually divided into two components. An encoder takes an image as input and processes it via a sequence of blocks and layers that learn to extract statistical data that are used to find and name things. The encoder's output is then sent to a decoder, which predicts bounding boxes and labels for each item. A pure regressor is the most basic decoder. The regressoris attached to the encoder output and directly predicts the position and size of each bounding box (From Figure 6). The model's output is the object's X, Y coordinate pair and its extent in the picture. Despite its simplicity, this paradigm has limitations. The number of boxes must be specified. If the image contains two dogs but the model was only built to recognise one, one will be left unidentified. Pure regressor-based models, on the other hand, may be a viable alternative if the number of items to forecast in each image is known ahead of time. A region proposal network is an extension of the regressor technique. The model in this decoder suggests parts of an image where it feels an item may dwell. To determine a label, the pixels in these areas are put into a classification sub network. The pixels containing such areas are subsequently sent through a classification network. The advantage of this strategy is that it produces a more accurate, flexible model that may suggest an arbitrary number of areas that may include a bounding box. However, the increased precision comes at the expense of computing efficiency.

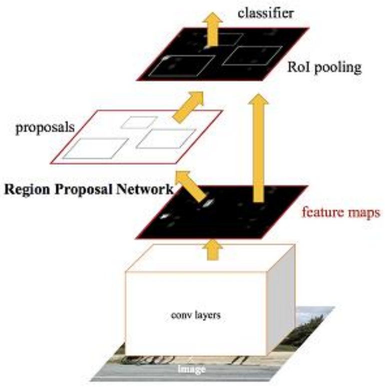


Figure 6 Basic Structure of Object Detection

4. Algorithms for Object Detection

Convolutional neural networks (R-CNN, Region-Based Convolutional Neural Networks), Fast RCNN, and YOLO are some common techniques for object identification (You Only Look Once). The R-CNNs belong to the R-CNN family, whereas YOLO belongs to the single shot detector family. The following are some of the most common deep learning algorithms:

YOLO

You Only Look Once (YOLO) is a well-known deep learning strategy for object identification. Using dimension clusters as anchor boxes, the YOLO technique predicts bounding boxes. Because most bounding boxes have specific height-width ratios, YOLO predicts off-sets from a specified collection of boxes with specific height-width ratios, known as anchor boxes. The anchor boxes are created by grouping the dimensions of the original dataset's ground truth boxes to determine the most prevalent forms and sizes. Using logistic regression, the network predicts an objectness score for each bounding box when the bounding box prior overlaps a ground truth object by more than any other bounding box prior. Unlike sliding window and area proposal systems, YOLO observes the full image during training and testing, implicitly encoding contextual information about classes as well as their appearance. The YOLO algorithm employs the following three techniques:

- The picture is split into grids by residual blocks. Each grid has a S x S dimension. The graphic below demonstrates how an input image is separated into grids.
- Bounding box regression: A bounding box is an outline in a picture that emphasises an item. Each bounding box in the picture has the following attributes: Width (bw), Height (bh), Class, and Center of the Bounding Box (bx,by)
- Union vs. Intersection (IOU) IOU is an object detection phenomena that defines how boxes overlap. YOLO use IOU to create an output box that properly surrounds the items. Each grid cell is in charge of forecasting the bounding boxes as well as their confidence ratings. If the expected and actual bounding boxes are the same, the IOU is equal to one. This approach removes bounding boxes that are not the same size as the actual box.

A convolutional neural network is used to build this model. The detection network is made up of 24 convolutional layers, which are followed by two fully connected layers. The features space from preceding layers is reduced by alternating 1 X 1 convolutional layers. On the ImageNet classification challenge, the convolutional layers are built at half the resolution (224 X 224 input picture) and then doubled for detection (From Figure 7).

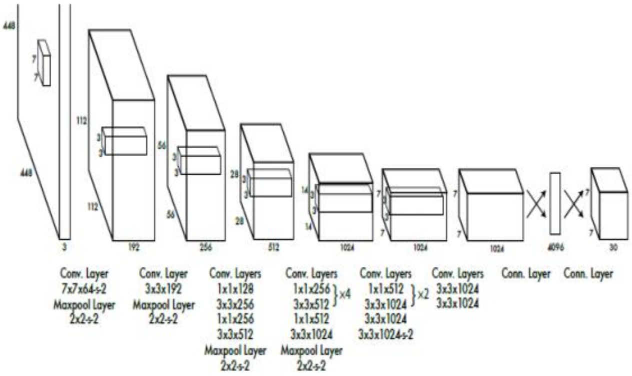


Figure 7 The Architecture of YOLO

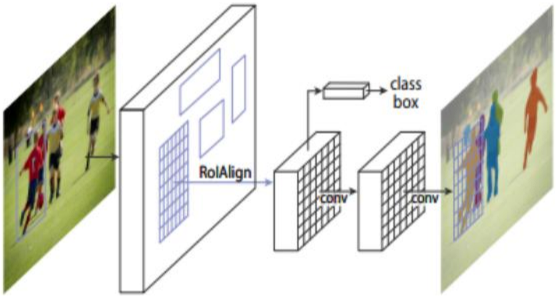
Mask R-CNN

Region-based convolutional neural networks (R-CNNs) are a ground-breaking technique to object identification that uses deep models. R-CNN models begin by selecting multiple suggested areas from an image, then labelling their categories and bounding boxes. These labels are generated depending on the program's established classes. They then use a convolutional neural network to perform forward computation on each suggested region to extract features. The input picture is initially segmented into roughly two thousand area portions, and each region is then processed using a convolutional neural network. The areas' sizes are computed, and the appropriate region is added into the neural network. The R-CNN family includes a number of models. One of them is Fast R-CNN. Mask R-CNN is a step up from Fast R-CNN.

The work flow of the model (From Figure 8) is given below:

- Pre-train a CNN network on image classification tasks.
- End-to-end fine-tune the RPN (region proposal network) for the region proposal task, which is started by the pre-train image classifier. Positive samples have IoU greater than 0.7, whereas negative samples have IoU less than 0.3.
- Using the current RPN's ideas, train a Mask R-CNN object detection model.

A third branch for predicting an object mask is added alongside the current classification and localization branches.



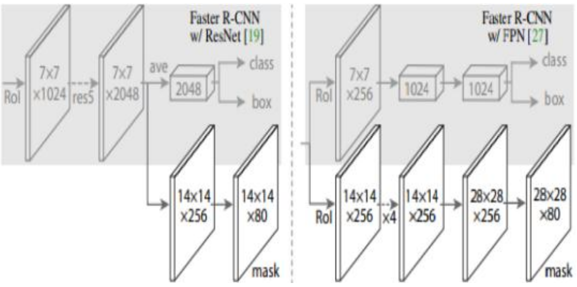


Figure 8The Mask R-CNN framework for instance segmentation

Figure 9Network architecture of Mask R-CNN

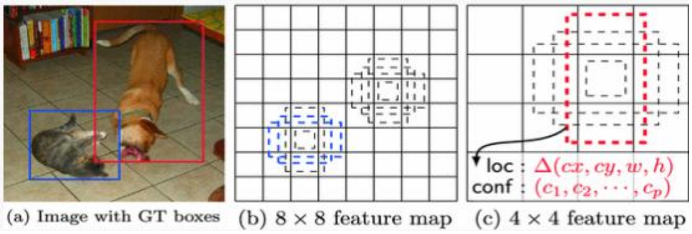
Mask R-multi-task CNN's loss function combines classification, localization, and segmentation mask losses: L=L(cls)+L(box)+L(mask), where L(cls) is the log loss function across two classes and L(mask) is the average binary cross-entropy loss.

Mask R-CNN is fundamentally straightforward (Figure 9): For each candidate item, Faster R-CNN produces two outputs: a class label and a bounding-box offset; to this, we add a third branch that produces the object mask. The additional mask output differs from the class and box outputs and necessitates the extraction of a much finer spatial arrangement of an item. The fundamental parts of Mask R-CNN are then introduced, including pixel-to-pixel alignment, which is the main missing piece in Fast/Faster R-CNN.

SSD

Detection of a Single Shot The Single Shot Detector (SSD) technique detects objects in photos by employing a single deep neural network. The SSD method divides the output space of bounding boxes into a series of default boxes with varying aspect ratios. The approach scales per feature map position after discretization. To naturally manage objects of varying sizes, the Single Shot Detector network integrates predictions from numerous feature maps with varied resolutions.

During training, SSD just requires an input picture and ground truth boxes for each item (Figure 10). We test a small collection of default boxes with varying aspect ratios at each position in multiple feature maps with different sizes (8x8, 4x4). We anticipate the shape offsets and confidences for all item categories for each default box. We begin training by matching these default boxes to the ground truth boxes. For example, we've matched two default boxes with the cat and one with the dog, which are considered positives and the others as negatives. The model loss is a weighted average of the localization losses.


loc : $\Delta(cx, cy, w, h)$
conf : $(c_1, c_2, \cdots, c_p)$

(a) Image with GT boxes (b) 8 × 8 feature map (c) 4 × 4 feature map

Figure 10The SSD framework

The SSD method is based on a feed-forward convolutional network (Figure 11), which generates a fixed-size collection of bounding boxes and scores for the existence of object class instances in those boxes, followed by a non-maximum suppression phase to get the final detections. The main difference between training SSD and training a standard detector that employs region suggestions is that ground truth information must be supplied to particular detector outputs from a predetermined set. End-to-end loss function and back propagation are used. Training also entails selecting a set of default detection boxes and scales, as well as hard negative mining and data augmentation procedures.

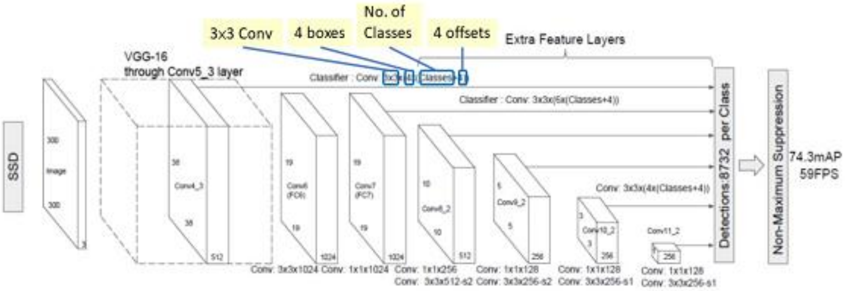


Figure 11SSD framework

5. Object Tracking

Object tracking is a deep learning application in which the software takes an initial set of object detections and creates a unique identifier for each of the initial detections before following the detected objects as they move across frames in a movie. (See Figure 12) Object tracking, in other terms, is the job of automatically recognizing objects in a video and interpreting them as a series of high-accuracy trajectories. Often, there is an indicator around the tracked item, such as a surrounding square that follows the object and shows the user where the object is on the screen. The primary principle behind tracking things in digital video is to deduce information about object or even camera movements. We can forecast (or estimate) the object's location and/or orientation based on the data obtained for a video frame. In contrast, the purpose of frame-based video object recognition is to locate the position of the item of interest in the scene without using its motion information. Object tracking has a clear benefit over object detection in that, in the event of several

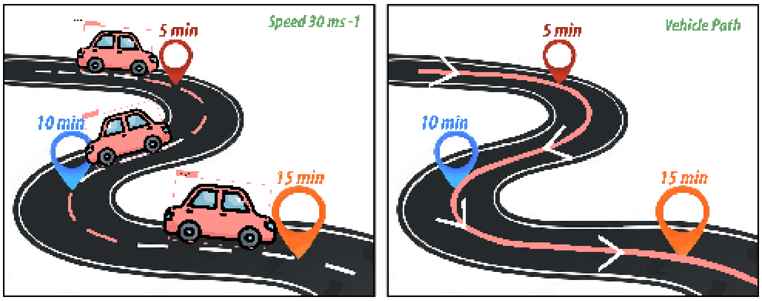


Figure 12 Object tracking

objects, the former may frequently give automated object identification as they move over time. Object tracking is further motivated by the fact that object detection is typically computationally sluggish or prone to detection mistakes. Even though object recognition is the ultimate aim, tracking can greatly minimise the search region inside a frame and hence the calculations needed.

Levels of Object Tracking

Because object tracking is such a broad application, it has several subcategories. Object tracking levels vary based on the number of items monitored. Object tracking levels are divided into two types:

- Single Object Tracking and
- Multiple Object Tracking (MOT)

Single Object Tracking (SOT)

Single Object Tracking generates bounding boxes for the tracker to use based on the first frame of the input picture. SOT indicates that a single item is monitored, even in contexts with several objects. Single Item Trackers are designed to focus on a single object rather than several. In the first frame, the item of interest is determined, and the object to be tracked is initialised for the first time. The tracker is then tasked with identifying that one-of-a-kind target in all subsequent frames. SOT is classified as detection-free tracking, which indicates that it requires human initialization of a certain number of objects in the first frame. These items are then localised in the next frames. One disadvantage of detection-free tracking is that it cannot handle circumstances in which new objects occur in the middle frames. Any given item should be able to be tracked by SOT models.

Multiple Object Tracking (MOT)

Many object tracking is described as the challenge of automatically detecting and portraying multiple objects in a video as a series of trajectories with high accuracy. As a result, multi-object tracking seeks to monitor many objects in digital photographs. It is also known as multi-target tracking since it analyses movies in order to detect objects ("targets") that correspond to more than one specified class. Multiple object tracking is critical in autonomous driving because it detects and predicts the behaviour of people and other vehicles. Multiple object tracking frequently has little to no prior training in terms of target appearance and quantity. The height, breadth, location, and other properties of bounding boxes are used to identify them.

5.1 Object Tracking Algorithms

The majority of multiple item tracking algorithms use a technique known as tracking-by-detection. The tracking-by-detection approach employs an independent detector that is applied to all picture frames in order to gather likely detections, followed by a tracker that is performed on the set of detections. There are two types of algorithms: batch algorithms and online algorithms. When determining the identify of an item in a certain frame, batch tracking algorithms employ information from future video frames. Non-local information about the item is used by batch tracking methods. This practise leads to improved tracking quality. While batch tracking algorithms access future frames, online tracking algorithms only use present and previous data to make decisions about a certain frame. They perform poorer than batch techniques due to the fact that online methods are limited to the current frame. They outperform in real-time situations involving object tracking.

Now some popular object tracking algorithms are going to be discussed below:

Open CV Object Tracking

Because Open CV has so many algorithms built in that are especially tailored for the demands and aims of object or motion tracking, it is a popular solution. BOOSTING, MIL, KCF, CSRT, Median Flow, TLD, MOSSE, and GOTURN are several Open CV object trackers. Each of these trackers is better suited to a certain aim. For example, CSRT is ideal when the user demands better object tracking precision and is willing to accept slower FPS throughput. The pros and disadvantages of any Open CV object tracking method must be considered before selecting one.

- BOOSTING Tracker: This tracker is based on the same algorithm that powers the machine learning underlying Haar cascades (Ada Boost), but it is over a decade old, much like Haar cascades. This tracker is sluggish and underutilised. Only for historical reasons and to compare to other algorithms. (Open CV 3.0.0 is required).
- MIL Tracker: It is more accurate than BOOSTING Tracker, although it reports failures poorly. (Open CV 3.0.0 is required)
- KCF Tracker: The KCF tracker is less precise than the CSRT but delivers a greater FPS. It also outperforms BOOSTING and MIL. Does not handle complete occlusion well, like MIL and KCF do. (Open CV 3.1.0 is required)
- Channel and geographical dependability are included in the CSRT Tracker. It is significantly slower than KCF but slightly more accurate. (Open CV 3.4.2 minimum)
- Median Flow Tracker: It performs a good job of reporting failures; but, if there is a major change in motion, such as fast moving objects or items that change appearance quickly, the model will fail. (Open CV 3.0.0 is required)
- Mosse Tracker: The MOSSE tracker is extremely quick, but its accuracy is significantly poorer than that of KCF tracking. MOSSE is still a viable option if you're seeking for the quickest object tracking Open CV technique. (Open CV 3.4.1 minimum)
- GOTURN Tracker: Open CV's only deep learning-based object detector. To execute, extra model files are required. The original Caffe version of GOTURN has been adapted to the Open CV Tracking API. (Open CV 3.2.0 minimum)

Deep SORT

Deep SORT is an excellent object tracking algorithm and one of the most extensively used object tracking frameworks. It is an addition to SORT (Simple Real time Tracker). Essentially, we track not just distance and velocity, but also appearance. Deep SORT allows us to add this feature by calculating deep features for each bounding box and considering deep feature similarity into the tracking algorithm. The purpose of the model is to simply track a single item inside a defined picture crop. They use a two-frame CNN architecture to properly regress onto the object utilising both the current and preceding frames. Deep SORT is a fast tracker that focuses on simple, effective algorithms and a realistic approach to multiple item tracking. Deep SORT uses Kalman Filtering and the Hungarian Algorithm to increase tracker speed. It averaged 16 frames per second while retaining exceptional accuracy, making it an excellent choice for multiple item recognition and tracking. The model is divided into four stages: detection, feature extraction/motion prediction, affinity, and association.

- Detection stage: Objects are identified from each input frame and bounding boxes are constructed around them in this stage.
- Feature extraction/motion prediction stage: A feature extractor collects feature vectors from target objects, and a motion predictor anticipates the eventual location of each monitored target.
- Affinity stage: Using feature vectors and motion predictions, a similarity/distance score is produced between pairs of detections and/or tracks.

The similarity/distance metrics are used in this step to correlate detections and tracks pertaining to the same target by assigning the same ID to detections that indicate the same target.

6. Conclusion

With data analytics and data mining, information and communication technologies aid in decision-making based on historical data. The amount of data accessible is massive, and extracting information and creating an attractive pattern from the accumulated data is a difficult undertaking. The use of artificial intelligence to tackle computer vision challenges surpassed traditional image processing methodologies. Some method of Image processing, algorithm of image processing have seen. Object detection and tracking of object using video signal are studied. The popular method of object detection CNN model. The large amount of data on which it will trained from each class accounts for the excellent validation accuracy. Images' performance data are recorded. For the KITTI and COCO datasets, multiple object detection is accomplished using YOLOv5.

References

[1] Ambardekar, M. Nicolescu, and G. Bebis, “Efficient vehicle tracking and classification for an automated traffic surveillance system,” Signal and Image Processing, 2008, pp. 1-6.

[2] Appathurai, R. Sundarasekar, and C. Raja, “An efficient optimal neural network-based moving vehicle detection in traffic video surveillance system,” Circuits, Systems and Signal Processing, 2020, vol. 39, no. 2, pp. 734-756.

[3] I. B. Parico and T. Ahamed, “Real time pear fruit detection and counting using yolov4 models and deep sort,” Sensors, 2021, vol. 21, no. 14, p. 4803.

[4] Yca, B. Bm, and C. Hong, “Part alignment network for vehicle re-identification -Science Direct,” Neuro computing, 2020, vol. 418, no. 5, pp. 114-125.

[5] Sudha and J. Priyadarshini, “An intelligent multiple vehicle detection and tracking using modified vibe algorithm and deep learning algorithm,” Soft Computing, 2020, vol. 24, no. 21, pp. 1-13.

[6] Bochinski, T. Senst, and T. Sikora, “Extending iou based multi-object tracking by visual information,” in 2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), IEEE, 2018, pp. 1-6.

[7] H. Tao and X. Lu, “Automatic smoky vehicle detection from traffic surveillance video based on vehicle rear detection and multi-feature fusion,” IET Intelligent Transport Systems, 2019, vol. 13, no. 2, pp. 252-259.

[8] J. Athanesious, V. Srinivasan, and V. Vijayakumar, “Detecting abnormal events in traffic video surveillance using super orientation optical flow feature,” IET Image Processing, 2020, vol. 14, no. 9, pp. 1881-1891.

[9] K. He, G. Gkioxari, P. Dollar, and R. Girshick, “Mask r-cnn,” in Proceedings of the IEEE international conference on computer vision, 2017, pp. 2961-2969.

[10] M. Sankaranarayanan, C. Mala, and S. Mathew, “Pre-processing framework with virtual mono-layer sequence of boxes for video based vehicle detection applications,” Multimedia Tools and Applications, 2020, vol. 5, no. 6, pp. 1-28.

[11] Mohana and HV Ravish Aradhya, “ Object Detection and Tracking using Deep Learning and Artificial Intelligence for Video Surveillance Applications,” International Journal of Advanced Computer Science and Applications, vol. 10 no.12, 2019, pp.517-530.

[12] P. Martinez and M. Barczyk, “Implementation and optimization of the cascade classifier algorithm for UAV detection and tracking,” Journal of Unmanned Vehicle Systems, 2019, vol. 7, no. 4, pp. 296-311.

[13] P. Priyadharshini and P. Karthikeyan, “Vehicle data aggregation from highway video of madurai city using convolution neural network,” Procedia Computer Science, 2020, vol. 171, no. 4, pp. 1642-1650.

[14] Q. Zhang, H. Sun, X. Wu, and H. Zhong, “Edge video analytics for public safety: a review,” Proceedings of the IEEE, 2019, vol. 107, no. 8, pp. 1675-1696.

[15] R. Feng, C. Fan, and Z. Li, “Mixed road user trajectory extraction from moving aerial videos based on convolution neural network detection,” IEEE Access, 2020, vol. 8, no. 4, pp. 43508-43519.

[16] Shaba Irram, Sheikh Fahad Ahmad, “Research on Object Detection in Video Streaming Using Deep Learning,” International Journal of Computational Engineering Research, vol. 09, July-2019, pp. 34-43.

[17] 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, 2016, pp. 21-37.

[18] W. Zhan, C. Sun, M. Wang, J. She, Y. Zhang, Z. Zhang, and Y. Sun, “An improved yolov5 real-time detection method for small objects captured by UAV,” Soft Computing, 2021, pp. 1-13.

[19] Xiangqian Wang, “Vehicle Image Detection Method Using Deep Learning in UAV Video,” Hindawi Computational Intelligence and Neuroscience, vol. 2022, pp. 1-10. February-2022.

[20] Y. Huang and H. Zhang, “A safety vehicle detection mechanism based on yolov5,” in 2021 IEEE 6th International Conference on Smart Cloud (Smart Cloud), IEEE, 2021, pp. 1-6.

[21] Z. Liu, D. Lu, W. Qian., “A method for restraining gyroscope drift using horizon detection in infrared video,” Infrared Physics & Technology, 2019, vol. 101, no. 3, pp. 1-12.