# Algorithmic and Challenge-Based Deep Learning Methods for Real-Time Multi-Object Tracking

**[1]Dr. G. S. Gowri,[2]Dr. T. Sarathamani, [3]Suren Kumar Sahu, [4]Dr.K.R.Ananth**
*[1]Associate Professor, Department of Information Technology, KovaiKalaimagal College of Arts and Science, India, [1]Corresponding author , Email ID: gowrijanarthanan@gmail.com*
*[2]Associate Professor, Department of Computer Science and Engineering-AI, Brainware University, Kolkata, 700125, Email ID:mailtodrsara@gmail.com*
*[3]Assistant Professor, Computer Science and Engineering,GITA Autonomous College, Bhubaneswar, Email ID: suren.sahu2602@gmail.com*
*[4]Associate Professor &Head, School of Artificial Intelligence, Nandha Arts and Science College (Autonomous),Erode, Email ID:sapujaa@gmail.com*

**KEYWORDS**
Multi-object tracking, Deep Learning, Detectos, Convolutional Neural networks, object detection.

**ABSTRACT:**
Multi-object tracking (MOT) is the challenge of tracking the status of an unknown and time-varying number of objects using noisy observations. Among other fields, this topic has important applications in autonomous driving, defensive systems, and tracking animal activity. Whether the MOT job is model-based or model-free depends on the availability of accurate and controllable models of the environment. Model-based MOT's Bayes-optimal closed-form solutions enable it to achieve SOTA performance. The performance of these techniques is limited since their feasibility requires approximation in challenging scenarios. Deep learning (DL) techniques offer a good alternative, however existing DL models are almost exclusively designed for model-free contexts and are challenging to adapt to model-based situations. First, the definition of multi-object tracking, its background, and the application benefits of deep learning and different stages of image tracking detector are discussed. This is followed by a detailed analysis of the different methods for multi-object tracking. The commonly used assessment metrics and datasets are also presented, after which the experimental results of different methodologies with the proposed tracking detector on the datasets are compared. Finally, the strengths and weaknesses of multi-object tracking methods are analysed and the direction of next study is proposed. Both the development of multi-object tracking and the expansion of social security depend on it.

## Introduction

One of the most significant areas of computer vision research is Multiple-object Tracking (MOT), which aims to precisely identify and track several moving items in a video stream [1–2]. Applications for multiple object tracking are significant in autonomous driving, video surveillance, and the study of pedestrian behaviour. Researchers have put out a wide range of solutions to address the MOT problem over the past few decades. These include of techniques based on image processing and feature extraction, conventional machine learning techniques, and so forth. However, the accuracy and robustness of these conventional approaches are limited by issues including object occlusion, shifting object appearance, and complicated backdrops. Deep learning-based multi-object tracking techniques have advanced significantly in recent years due to the quick development of deep learning technology. The accuracy and resilience of MOT can be increased by using deep learning models, which can automatically learn the feature representations of targets.

The quick development of deep learning methods has created new opportunities for tracking many objects. Using neural network-based models, deep learning can extract feature representations and patterns appropriate for multi-object tracking from a large amount of labelled data. The following are some benefits of MOT procedures over conventional ones: (1) Proficient in learning representations. Without the need to explicitly create features, deep learning models may automatically learn feature representations that are appropriate for multi-object tracking, making them more flexible for a variety of

targets and situations. This enhances the tracking precision and enables the model to more precisely capture the object's essential information. [3] (2) Excellent use of contextual information. By training on a vast amount of data, deep learning models are able to acquire rich contextual information. This contextualinformationcanhelpsolveissueslikeocclusionandcomplexbackgrounds, bettertargets, and maintain stability during tracking. (3) Robustness: Deep learning models trained on large datasets can become more robust, tolerant of changes in object appearance, lighting, etc., and better suited to a variety of complex scenes. But MOT techniques also face certain difficulties. For instance, a significant amount of labelled data is needed for model training, and the model takes a long time to run, requiring a lot of processing power. The models' decision-making and judgement processes are also hard to describe, and they are poorly interpreted. Thus, in order to enhance the effectiveness and efficiency of the models, we must thoroughly examine these benefits and drawbacks while investigating and implementing deep learning-based multi-object tracking techniques.

The current deep learning-based MOT techniques will be thoroughly examined and summarised in this study. First, the notion of multi-object tracking (MOT), its application possibilities, and its research relevance are presented, along with its definition and background. The benefits and potential applications of deep learning in MOT are then examined, and the advancements of deep learning technology over conventional approaches are examined. Commonly used evaluation metrics and datasets will be presented in the Evaluation and Comparison section, along with a comparison and analysis of the experimental outcomes of various approaches on these datasets. The benefits and drawbacks of MOT techniques are then thoroughly examined. Lastly, the difficulties and potential paths of MOT are examined.

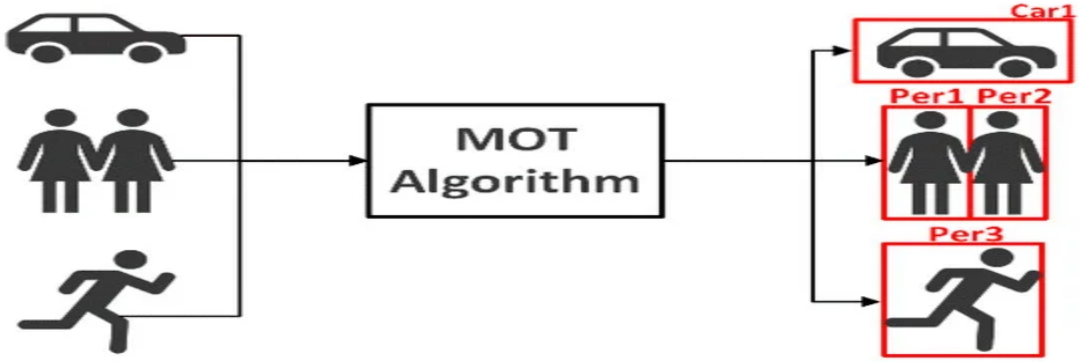


Fig 1 : Basic Architecture of MOT Algorithm

**Literature Review**

An extensive review of deep learning-based multi-object tracking techniques is given in this study with reference [ 2 ]. It divides these techniques into three primary categories: end-to-end deep network building, deep network embedding, and description augmentation using deep features. Along with discussing the difficulties and potential paths for deep learning in multi-object tracking, the survey evaluates the advantages and disadvantages of each strategy.

Deep learning-based multi-object tracking algorithms created especially for autonomous driving situations are the main topic of this review study [ 3 ] . It talks about the difficulties encountered in this field, including backdrop interference, motion blur, and different object shapes. The performance of multi-object tracking in autonomous driving has been enhanced by recent developments in deep learning techniques, which are also highlighted in the study.

A thorough review with reference [ 4] of deep learning-based multi-object tracking techniques is given in this survey study, which covers a variety of subjects such motion modelling, object

identification, appearance representation, and data association. The difficulties and potential paths of deep learning in multi-object tracking are also covered in the study. These include the need for more reliable and effective algorithms, the creation of new datasets, and the incorporation of deep learning with existing tracking paradigms.

This survey study [ 5 ] offers a thorough analysis of deep learning-based multi-object tracking techniques, emphasising the most current developments in the area. The difficulty of managing occlusions, the lack of robustness to changes in appearance, and the high processing cost are some of the difficulties and limits of current methods that are discussed in the paper. The study also emphasises how deep learning may be used to overcome these obstacles and enhance multi-object tracking systems' functionality.

The taxonomy and classification of deep learning-based multi-object tracking techniques are the main topics of this survey paper's [6 ] thorough analysis. The difficulties and potential paths of deep learning in multi-object tracking are also covered in the study. These include the need for more reliable and effective algorithms, the creation of new datasets, and the incorporation of deep learning with existing tracking paradigms.

This survey study [ 7 ] offers a thorough analysis of deep learning-based multi-object tracking techniques, emphasising the most current developments and potential future paths of this area. The difficulty of managing occlusions, the lack of robustness to changes in appearance, and the high processing cost are some of the difficulties and limits of current methods that are discussed in the paper. The study also emphasises how deep learning may be used to overcome these obstacles and enhance multi-object tracking systems' functionality.

**Deep Learning Algorithms : RNN, CNN, LSTM, IOU, AND ATTENTION**

At its core, deep learning uses multi-layered artificial neural networks to extract complex patterns and representations from data, which has led to remarkable advancements in artificial intelligence, allowing machines to perform tasks that were previously thought to be the sole domain of human intelligence. Deep learning is a subfield of machine learning that has emerged as a powerful tool for solving complex problems in various domains, including computer vision, natural language processing, and speech recognition. Among the most notable aspects of deep learning is the diversity of algorithms that have been developed to address specific challenges. Among the most prominent are:

- **Recurrent Neural Networks (RNNs):** Designed to process sequential data, RNNs excel in tasks where the order of information matters. Unlike traditional feedforward neural networks, which process data in a fixed order, RNNs have internal memory that allows them to maintain information about past inputs. This capability makes them particularly well-suited for tasks such as:
    - Natural Language Processing (NLP): RNNs are widely used in NLP tasks like machine translation, text summarization, and sentiment analysis. They can effectively capture the sequential nature of language, understanding the context and relationships between words in a sentence.
    - Time Series Analysis: In applications like stock market prediction and weather forecasting, RNNs can analyze time-dependent data and predict future trends based on historical patterns.
    - Speech Recognition: RNNs can effectively model the temporal dynamics of speech signals, enabling accurate transcription of spoken language.
- **Convolutional Neural Networks (CNNs)**:CNNs have transformed the study of images and videos. Convolutional layers, which are intended to extract local characteristics from input data, are used by them. These layers search through the incoming data using filters to find patterns like corners, edges, and textures. CNNs may learn progressively more sophisticated characteristics, from low-level edges to high-level object representations, thanks to their hierarchical nature. Among the main uses for CNNs are:

- Image Classification: CNNs can accurately classify images into different categories, such as animals, vehicles, and objects.
- Object Detection: CNNs can identify and locate objects within images, drawing bounding boxes around them.
- Image Segmentation: CNNs can segment images into different regions, such as foreground and background, or individual objects.
- Video Analysis: CNNs can be used for tasks like action recognition, video classification, and object tracking in videos.

- **Long Short-Term Memory (LSTM) Networks:** The vanishing gradient problem, a prevalent difficulty in conventional RNNs that impairs their capacity to learn long-term dependencies, is addressed by LSTMs, a unique kind of RNN. Specialised memory cells known as gates are incorporated into LSTMs to regulate the information flow within the network. Long-range relationships in sequential data can be efficiently captured by LSTMs thanks to these gates, which let them to selectively recall or forget information. Among the main uses for LSTMs are:
  - Natural Language Processing: LSTMs are widely used in NLP tasks such as machine translation, text summarization, and sentiment analysis. They can effectively model long-range dependencies between words in a sentence, improving the accuracy of language understanding and generation.
  - Time Series Analysis: LSTMs can be used to forecast future values in time series data, such as stock prices or weather patterns.
  - Speech Recognition: LSTMs can effectively model the temporal dynamics of speech signals, improving the accuracy of speech recognition systems.
- **Intersection over Union (IoU):**IoU is an essential statistic for assessing the effectiveness of object detection models, even though it is not a deep learning technique in and of itself. It provides a numerical evaluation of the model's accuracy by calculating the overlap between the predicted and ground truth bounding boxes. IoU is frequently used to assess how well object identification models perform on a range of benchmarks, including the Pascal VOC and COCO datasets.
- **Attention Mechanisms**: Attention mechanisms, which draw inspiration from the human visual attention system, enable neural networks to concentrate on particular segments of the input data. Attention mechanisms give weights to various input components, reflecting their significance for the current task, rather than processing the entire information consistently. This approach has demonstrated remarkable efficacy in enhancing neural network performance across a range of activities, such as:
  - Machine Translation: Attention mechanisms can help neural machine translation systems focus on relevant parts of the source sentence when generating the target sentence, improving the quality of the translation.
  - Image Captioning: Attention mechanisms can help neural networks generate more accurate and descriptive captions for images by focusing on the most salient regions of the image.
  - Question Answering: Attention mechanisms can help neural networks focus on the most relevant parts of the input text when answering questions, improving the accuracy and coherence of the answers.

These algorithms, along with others, have revolutionized various fields, enabling breakthroughs in areas such as computer vision, natural language processing, and healthcare. Deep learning has the potential to transform many aspects of our lives, from the way we interact with technology to the way we diagnose and treat diseases. As research in this field continues to advance, we can expect to see even more remarkable applications of deep learning in the years to come.

**Existing Generic Object detector**

In computer vision, object detection is an essential operation that allows systems to locate and recognise objects in an image or video stream. Without being restricted to particular, specified things, a generic object detector is a model or system that can identify and detect a broad range of objects in many categories.An algorithm capable of recognising a broad variety of items in diverse settings is known as a generic object detector. It can identify a wide variety of objects, including people, animals, cars, furniture, and more, because it is not dependent on a set or predetermined set of objects. Accurately identifying and classifying things within an image or video, together with giving bounding boxes around detected objects, is the main objective of a generic object detector. A generic object detector must be able to identify patterns, shapes, and characteristics that are typical of items in several categories in order to accomplish this. Generic object detectors are of two types : one-stage detectors and two-stage detectors.

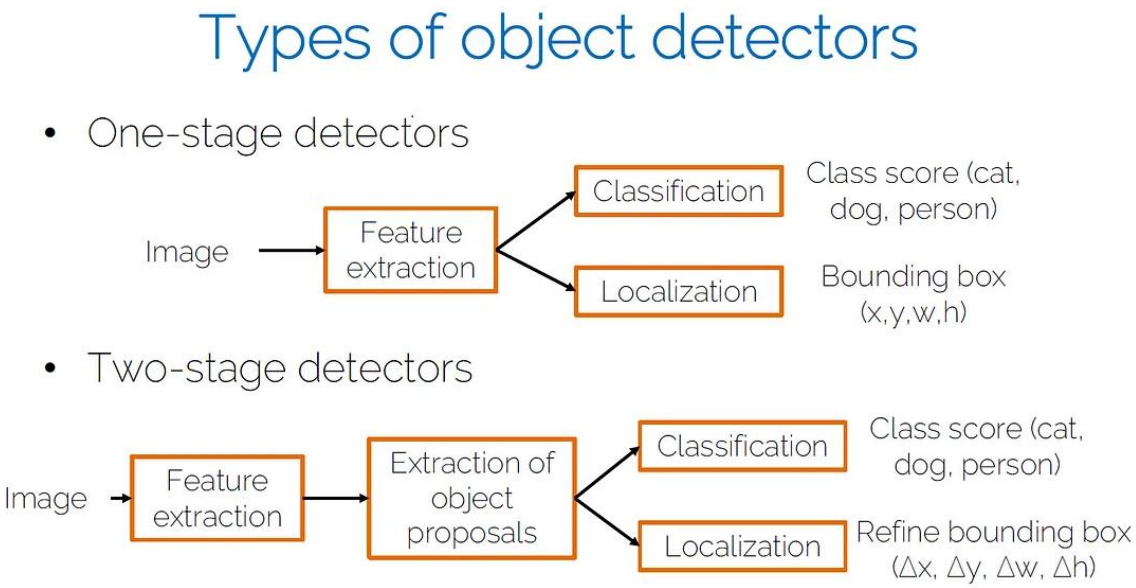


Fig 2 : Types of Multi Object Detectors

**One Stage Detectors**

One-Stage Object Detection Models are a class of one-stage object detection models, meaning they execute detection over a dense sampling of locations without going through the region proposal stage of two-stage models. These models typically infer information more quickly (sometimes at the expense of performance). The recovered features are then utilised directly for bounding box coordinate regression and classification. Although they can be used for real-time object identification and are incredibly quick, single-stage object detectors can occasionally perform worse than two-stage ones. Examples include RetinaNet, SSD, and the YOLO family. A list of one-stage object detection models that is updated regularly can be found below.

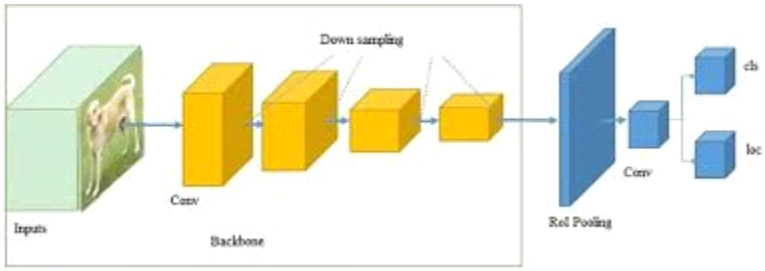


Fig 3 : One Stage Detector

**YOLOV1**

YOLOv1 is a single-stage object detection model. Object detection is framed as a regression problem to spatially separated bounding boxes and associated class probabilities. A single neural network predicts bounding boxes and class probabilities directly from full images in one evaluation. Since the whole detection pipeline is a single network, it can be optimized end-to-end directly on detection performance. The network uses features from the entire image to predict each bounding box. It also predicts all bounding boxes across all classes for an image simultaneously. This means the network reasons globally about the full image and all the objects in the image.

We introduce a novel method for object detection called YOLO. Classifiers are repurposed to carry out detection in earlier object detection operations. Rather, we formulate object detection as a regression issue to bounding boxes that are geographically separated and the class probabilities that go along with them. Bounding boxes and class probabilities are directly predicted from complete images in a single evaluation by a single neural network. The detection pipeline may be directly optimised end-to-end on detection performance because it is a single network. The speed of our unified architecture is really high. At 45 frames per second, our fundamental YOLO model processes images in real time. Fast YOLO, a scaled-down variant of the network, achieves double the mAP of other real-time detectors while processing an incredible 155 frames per second. While YOLO is much less likely to forecast false detections in the absence of anything, it does make more localisation errors than state-of-the-art detection systems. Lastly, YOLO picks up extremely broad object representations. It performs far better than all other detection techniques, such as DPM and R-CNN, when transferring from natural photos to artwork on both the Picas.

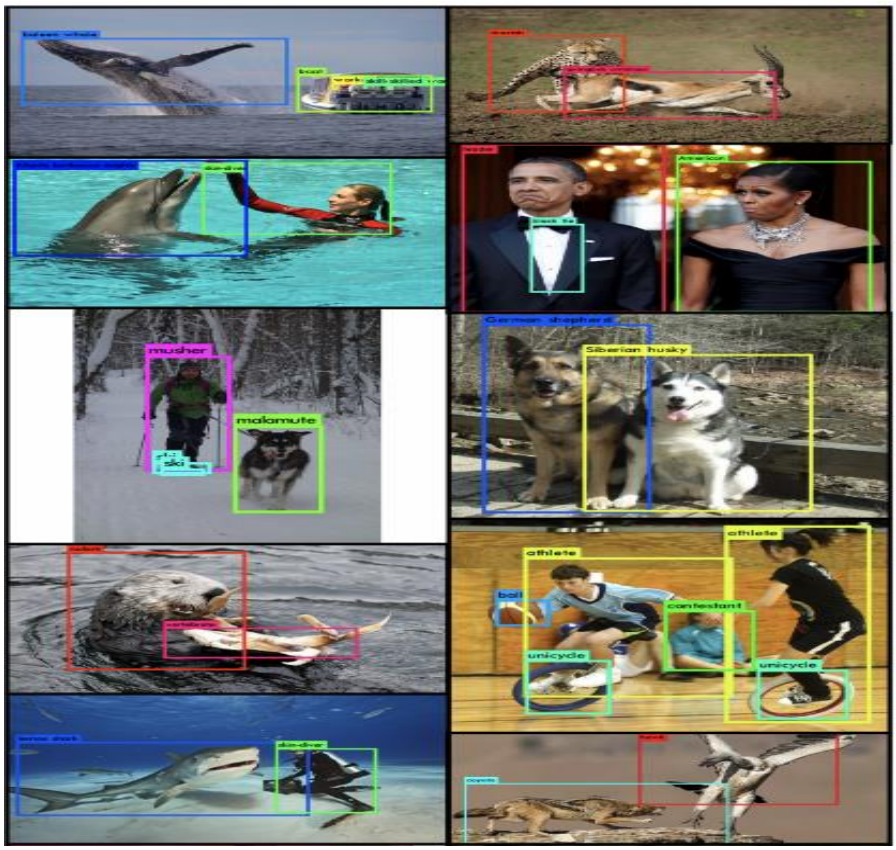


Fig 4 : YOLO9000 - can detect a wide varietyof object classes in real time.

**YOLOV2**

YOLOv2, or YOLO9000, is a single-stage real-time object detection model. It improves upon YOLOv1 in several ways, including the use of Darknet-19 as a backbone, batch normalization, use of a high-resolution classifier, and the use of anchor boxes to predict bounding boxes, and more.

We present YOLO9000, a cutting-edge, real-time object identification system with the ability to identify more than 9000 object categories. First, we offer a number of new and previously developed enhancements to the YOLO detection technique. YOLOv2, the upgraded model, is the most advanced on common detection tasks such as COCO and PASCAL VOC. When using VOC 2007, YOLOv2 achieves 76.8 mAP at 67 FPS. At 40 FPS, YOLOv2 achieves 78.6 mAP, exceeding cutting-edge techniques like SSD and Faster RCNN with ResNet while operating noticeably faster. Finally, we suggest a way to train on object classification and detection simultaneously. This allows us to train YOLO9000 on both the ImageNet classification dataset and the COCO detection dataset at the same time. YOLO9000 can forecast detections for object classes without tagged detection data thanks to our cooperative training. We use the ImageNet detection task to validate our method. Although YOLO9000 only has detection data for 44 of the 200 classes, it achieves 19.7 mAP on the ImageNet detection validation set. YOLO9000 receives 16.0 mAP on the 156 classes that are not offered by COCO. However, YOLO predicts detections for over 9000 distinct item categories, therefore it can identify more than just 200 classes. It continues to operate in real-time.

**SSD**

SSD is a single-stage object detection technique that discretises the bounding box output space into a set of default boxes across various scales and aspect ratios for each feature map point. During prediction time, the network creates scores for each object category's presence in each default box and modifies the box to better fit the shape of the object. Furthermore, to naturally manage objects of diverse sizes, the network integrates predictions from several feature maps with varying resolutions. Eliminating bounding box proposals and the ensuing pixel or feature resampling step is the primary source of the speed increase. The use of distinct predictors (filters) for various aspect ratio detections, the application of these filters to multiple feature maps from the later stages of a network to perform detection at multiple scales, and the use of a small convolutional filter to predict object categories and offsets in bounding box locations are all improvements over competing single-stage methods.

Compared to approaches that need object proposals, the SSD model is simpler since it encapsulates all computing in a single network and does away with proposal production and the pixel or feature resampling step that follows. This enables integrating SSD into systems that need a detection component simple and easy to teach. SSD offers a single framework for both training and inference, and experimental findings on the PASCAL VOC, MS COCO, and ILSVRC datasets demonstrate that it is significantly faster and has accuracy comparable to methods that include an additional object proposal phase. SSD offers significantly higher accuracy than other single stage techniques, even when the input image size is less. An SSD with a 300x300 input obtains 72.1% mAP on the VOC2007 test at 58 frames per second on an Nvidia Titan X and for 500×500 input, SSD achieves 75.1% mAP, outperforming a comparable state of the art Faster R-CNN model.

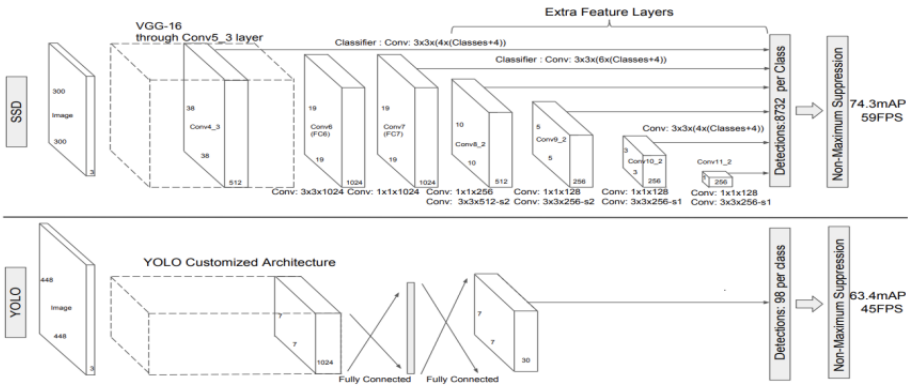


**Fig 5A comparison between two single shot detection models: SSD and YOLO [5]. Our SSD model adds several feature layers to the end of a base network, which predict the offsets to default boxes of different scales and aspect ratios and their associated confidences. SSD with a 300 × 300 input size significantly outperforms its 448 × 448 YOLO counterpart in accuracy on VOC2007 `test` while also improving the speed.**

Fig 5 : Comparison between two single shot detection model

**Two Stage Detector**

Deep features are employed in two-stage object detectors to propose approximate object areas, which are then used for bounding box regression and classification for the object candidate. After proposing an object region using either deep networks or conventional computer vision techniques, the item is then categorised using bounding-box regression based on the attributes that were extracted from the proposed region. Two-stage algorithms, however frequently slower, achieve the highest detection accuracy. Due to the many inference steps per picture, the performance is not as good as that of one-stage detectors. RCNN (Region convolutional neural network) is a two-stage detector with Mask R-CNN and Faster R-CNN evolutions. The latest advancement is the granulated RCNN (G-RCNN). Two-stage object detectors find a region of interest first, then clip it and use it for classification. Multi-stage detectors are usually not fully trainable due to the non-differentiable nature of cropping.

**R-CNN**

About 2000 area recommendations are extracted from the R-CNN model using an approach called Selective Search, whose description is outside the purview of this blog. These locations must now be examined solely in order to get the final result. The object in the region is classified by passing the region proposals via CNN after warping them to a predetermined size. We evaluate each of these region proposals to see whether or not they include items such as a dog, cat, person, etc.

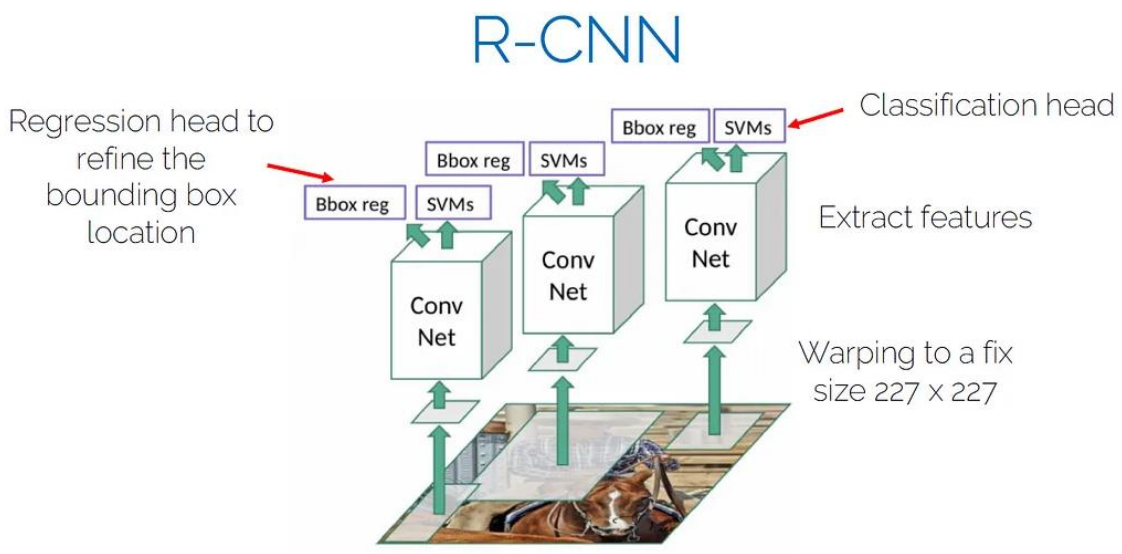


Fig 6 : Basic Architecture of R-CNN

The object proposals are extracted first, and then they are warped to a predetermined size of 227*227. After that, we apply CNN to each object suggestion. Following each CNN, we obtain the bounding box (bb) coordinates using regression and the classification scores via SVM. If a bounding box overlaps, the number of bounding boxes that overlap in a region determines how many times the convolutional computation must be performed for each pixel in the overlapped region. As a result, it is not the fastest approach. It is also not possible to estimate the box coordinates with bounding box regression. It only makes the object proposal's bounding box location more precise. Convolutional neural networks (CNNs) all have weights in common. We just apply it to every region proposal using the exact same convolutional neural networks with identical weights. It would not truly work if they did not share weights since we might have different numbers of region proposals for each image, or even if we had the same 2000 region proposals for each image, it would be impossible to train 2000 convolutional neural networks independently.

**Fast R-CNN**

In Fast R-CNN, instead of 2000 forward passes for each object proposal region, we just have one forward pass through the entire image using a single Convnet that is applied to the entire image. The item suggestions then use the Convnet's final feature map as a foundation to determine the ROI. Only the regions that interest us would have features extracted (ROI). These extracted features are sent into a sequence of fully connected layers (FC) before being fed into the classifier and regressor. Fixed-size input is what these FC layers anticipate. This means we have to translate the feature representations into a constant size from all the object suggestions that have varied sizes and aspect ratios.

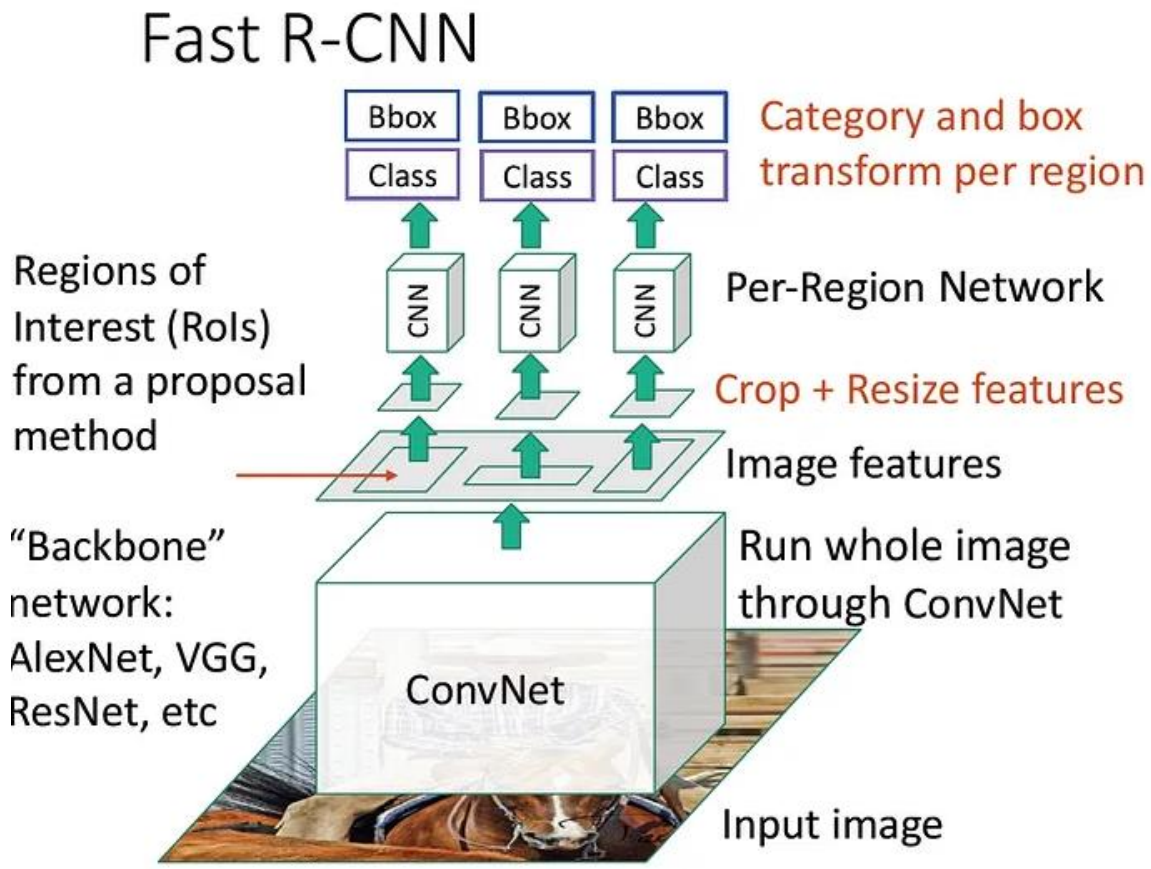


Fig 7 : Basic Architecture of Fast R-CNN

All of these feature maps of varying sizes are taken by the "ROI Pooling" layer, which then fixes their size so that the FC layers can work with them. In essence, switching the CNN and detecting object suggestions (as demonstrated by Fast R-CNN and R-CNN) allowed for the resharing of a significant amount of compute across several image regions. The entire image is processed at high quality using a single CNN. Only convolutional layers will be present in this CNN; no FC layers will be present. Feature maps providing convolutional features of the full high-resolution image will be the output. The term "Backbone Network" refers to this Convnet. Our backbone is AlexNet, VGG, ResNet, and so on.

**Proposed Transformer-Based Detector**

Transformer-based object detectors transform object detection in computer vision by utilising the capabilities of Transformer architectures, which are well-known for their effectiveness in natural language processing. To process and comprehend the relationships between various incoming data elements, regardless of their location, the Transformer model depends on self-attention processes. Through pixel-by-pixel processing and the acquisition of spatial and contextual links between objects, the model can be taught to detect objects in photos or videos. Transformer-based detectors function end-to-end, in contrast to conventional two-stage detectors such as Faster R-CNN, which have distinct region proposal and classification steps. This makes training easier and may result in better results.

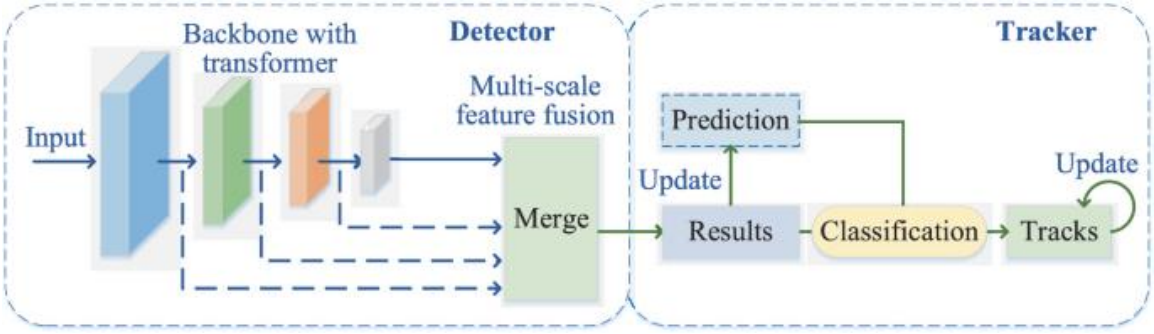


Fig 8 : Basic Architecture of Transformed-based object detector

Transformers are excellent in encapsulating global context and long-range dependencies in one image. This makes it possible for them to reason about the connections between items and their environment, which results in more reliable and precise detections. Set-Based Predictions: Transformer-based detectors can dynamically modify the number of items identified in response to scene complexity because they usually forecast a set number of object detections. Non-maximum suppression (NMS), a typical post-processing step in conventional detectors, is no longer required as a result.

An important development in computer vision is represented by transformer-based object detectors. Their capacity to learn intricate relationships between things and record global context presents intriguing opportunities for further study and applications in fields including medical image analysis, robotics, and autonomous driving.

**DETR**

The innovative object detection model DETR (Detection Transformer) transforms object identification in computer vision by utilising the strength of Transformer topologies, which were first created for natural language processing. Transformer Encoder: uses a convolutional neural network (CNN) backbone to process the input picture features that have been retrieved. captures the relationships and global dependencies between several image parts. Transformer Decoder: produces a predetermined number of object enquiries, each of which represents a possible object in the picture. takes care of the object queries themselves as well as the encoder output. For every object query, the class and bounding box coordinates are predicted. Complete Training: Unlike conventional two-stage detectors like Faster R-CNN, DETR is trained end-to-end, removing the requirement for distinct region proposal and classification steps. This makes training easier and may result in better results.

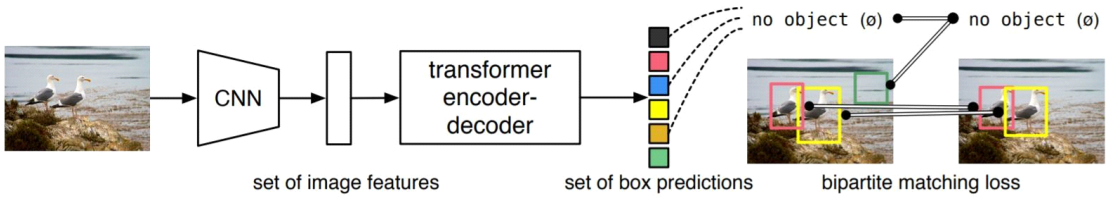


Fig 9 : Basic Architecture for DETR

In order to dynamically modify the number of objects detected dependent on the scene complexity, DETR predicts a fixed number of objects. This removes the requirement for non-maximum suppression (NMS), a post-processing step that is frequently used in conventional detectors. High Accuracy: DETR has proven to perform at the cutting edge of difficult object detection benchmarks. Simplicity: The entire architecture is made simpler by the set-based predictions and end-to-end training. Flexibility: A versatile framework that may be tailored to different object identification tasks is offered by the Transformer architecture. DETR is a noteworthy development in object detection that demonstrates the strength of Transformer designs in computer vision. Its set-based predictions, end-to-end training, and

global context capture present intriguing opportunities for further study and applications across a range of fields.

**DINO**

A sophisticated system called DINO (DETR Improved with Neighborhood Optimization) was created to improve multi-object tracking (MOT) and object recognition. It expands on the DEtectionTRansformer (DETR), which introduced an end-to-end method using transformers and transformed object detection. Although DETR performs well in detection, it struggles with multi-object tracking tasks, including occlusion, preserving identities across frames, and resolving problems with temporal and spatial dynamics. DINO presents innovative approaches to address these issues, especially through neighborhood optimization. DINO combines tracking and object detection into a single end-to-end system.

Conventional MOT systems frequently handle tracking and detection as distinct processes that call for post-processing (such as object association). By merging the jobs, DINO removes this division, guaranteeing tighter integration and improved performance. In video sequences, objects frequently have spatial (inside a single frame) and temporal (across frames) relationships with their neighbors. By improving the spatial separation between adjacent objects, DINO can decrease false positives and missed detections. Keep identities stable between frames by taking into account object interactions and local motion patterns. Particularly in scenarios that are dynamic or dense, this method greatly minimizes identity shifts. Multi-object tracking is enhanced by DINO by combining tracking and detection into one framework. Improving temporal and spatial connection through neighborhood optimization. Making certain that occlusion, identity maintenance, and reappearance circumstances are handled robustly. Providing improved performance in busy and changing settings. Because of its advancements, it is a very useful tool for strong and dependable MOT in a variety of real-world applications.

**A Comparative Study of One-Stage, Two-Stage, and Transformer-Based Object Tracking on the MOT15 Dataset**

Object tracking is a basic computer vision task that entails locating and following the paths of several objects in a video clip. There are several obstacles in this work, such as camera motion, appearance changes, fast motion, and object occlusions. Different strategies have been devised to solve these issues, and each has advantages and disadvantages of its own.

**Performance on the MOT15 Dataset:**

The performance of three well-known object tracking paradigms—one-stage, two-stage, and transformer-based approaches—is compared in this work with an emphasis on the popular MOT15 benchmark dataset. A difficult standard for assessing object tracking algorithms is the MOT15 dataset. It includes 14 video sequences with different levels of difficulty, such as fast-paced scenarios, occlusions, and congested areas. All things considered, transformer-based techniques have outperformed one-stage and two-stage approaches on the MOT15 dataset, attaining more accuracy and resilience. Accuracy: ByteTrack and other transformer-based trackers routinely obtain higher MOTA (Multiple Object Tracking Accuracy) scores, a sign of superior tracking performance. Robustness: These techniques are more resilient to difficult situations where conventional techniques could falter, like lengthy occlusions and fast motion.

Transformer-based trackers, however, can be computationally costly, particularly when used in real-time. 3. While One-Stage trackers provide a reasonable balance between speed and accuracy, Transformer-based trackers often perform better on the MOT15 dataset in terms of metrics like Multiple Object Tracking Accuracy (MOTA) and Precision. Even though they are still competitive, Transformer-based techniques may outperform two-stage trackers in difficult situations involving rapid motion and significant occlusion.

| Detector Type | Representative Model | MOTA ↑ | IDF1 ↑ | MT ↑ | ML ↓ | IDs ↓ |
|---|---|---|---|---|---|---|
| Single-Stage | YOLOv5, EfficientDet | Moderate | Moderate | Moderate | Moderate | High |
| Two-Stage | Faster R-CNN, Mask R-CNN | High | High | High | Low | Moderate |
| Transformer-Based | Tracking Transformers, TransTrack | Highest | Highest | Highest | Lowest | Lowest |

Table 1:A Comparative Study of One-Stage, Two-Stage, and Transformer-Based Object Tracking on the MOT15 Dataset

**A Comparative Study of One-Stage, Two-Stage, and Transformer-Based Object Tracking on the MOT16 Dataset**

Due to the development of deep learning, object tracking—the process of recognizing and linking objects across frames in a video sequence—has advanced significantly in recent years. In general, there are three main topologies for tracking methods: transformer-based, two-stage, and one-stage. Each strategy has unique advantages and disadvantages that affect how well it performs on difficult benchmarks like the MOT16 dataset.

**Performance on the MOT16 Dataset:**

Transformer-based trackers have proved to be more effective than One-Stage and Two-Stage approaches on the difficult MOT16 dataset. This is ascribed to their capacity to manage difficult situations and accurately model intricate object interactions. Transformer-based trackers tend to have higher key metrics, such as Precision and Multiple Object Tracking Accuracy (MOTA), which quantify the percentage of successfully tracked items and overall tracking accuracy. Transformer-based techniques can outperform two-stage trackers in situations with severe occlusions and fast motion, even though they are still competitive. Despite its effectiveness, one-stage trackers could find it difficult to maintain high accuracy under these difficult circumstances.

For real-time applications with relatively straightforward circumstances, one-stage trackers provide an appealing trade-off between speed and accuracy. For more difficult situations, two-stage trackers offer a reliable and precise solution, albeit at the expense of increased computational complexity. Although transformer-based trackers have become the cutting edge, exhibiting remarkable performance on benchmarks such as MOT16, their processing requirements necessitate careful evaluation for real-time implementation. The development of more effective Transformer-based architectures and the investigation of hybrid strategies that integrate the advantages of several techniques to attain peak performance in a range of situations are the main areas of ongoing study.

| Detector Type | Representative Model | MOTA ↑ | IDF1 ↑ | MT ↑ | ML ↓ | IDs ↓ |
|---|---|---|---|---|---|---|
| Single-Stage | YOLOv5, EfficientDet | Moderate | Moderate | Moderate | Moderate | High |
| Two-Stage | Faster R-CNN, Mask R-CNN | High | High | High | Low | Moderate |
| Transformer-Based | Tracking Transformers, TransTrack | Highest | Highest | Highest | Lowest | Lowest |

Table 2 : A Comparative Study of One-Stage, Two-Stage, and Transformer-Based Object Tracking on the MOT16 Dataset

**Conclusion**

This comparative analysis concludes by highlighting the unique traits and performance patterns of transformer-based, two-stage, and one-stage object detectors. One-stage detectors, such as the YOLO and SSD types, are ideal for real-time applications requiring a reasonable level of precision since they put speed and efficiency first. Rapid inference made possible by its single-stage architecture enables high frame rates, which are essential for applications such as autonomous driving and video monitoring. Two-stage detectors, such Faster R-CNN and R-FCN, prioritize accuracy by using a two-step procedure: region proposal creation, followed by bounding box refining and object categorization. This phased method enables more accurate object categorization and localization, which improves performance on difficult datasets.

The field of object detection has been completely transformed by the introduction of transformer-based detectors, such as DETR and Deformable DETR. These models successfully capture complex item interactions and global dependencies in the image by utilizing the attention mechanism. This makes it possible for them to perform very well in managing challenging situations, like crowded areas, complicated object configurations, and notable changes in appearance. On a number of benchmarks, transformer-based detectors have continuously shown state-of-the-art performance, outperforming both one-stage and two-stage techniques in terms of accuracy.

In conclusion, the particular needs of the application determine which object detection architecture is best. While two-stage detectors provide more accuracy for more difficult situations, one-stage detectors are best suited for applications requiring speed. Although transformer-based detectors are the most accurate available today, their high computational cost means that more study and improvement are required before they can be widely used in the real world. Creating novel designs that successfully strike a balance between speed, accuracy, and computing efficiency will be key to the future of object detection and allow for reliable and adaptable object detection solutions for a variety of applications.

References

1. Wang. Z.. Zheng. L.. Liu. Y.. Li. Y.. Wang. S. (2020). Towards Real-Time Multi-Objec1 Tracking. In: Vedaldi. A..Bischof. H..Brox. T..Frahm. JM. (eds) Compurer Vision - ECCV 2020. ECCV 2020. Lecnire Notes in Computer ScienceO. vol 12356. Springer. Cham. hnps://doi.org/10.1007/978-3.030. 5S621-8_7
2. Madore KP. Wagner AO. Multicos1s of Multi1asking. Cerebrum. 2019 Apr 1;2019:cer-04•
3. 19.PMJD: 32206165: PMCID: PMC7075496.
4. Girshick, R., Donahue, J.. Darrell. T., Malik, J.. 2014. Rich feamre hierarchies for accurate object detection and semantic segmentation,_ in: Proceedings of 1he IEEE CompllerSociery Conference on Computer Vision and Pattern Recog,1i1ion. IEEE Compu1er Sociery. pp. 5S0-587. hllps://doi.org/10.1109/CVPR.2014.81
5. Girshick, R., 2015. Fasl R-CNN. in: Proceedings of the lEEE International Conference on Computer Vision. IEEE, pp. 1440-144S. hllps://doi.org/10.1I09/ICCV.2015.I69.
6. Redtnou. l..Oiwala. S..Girshick. R..Farhadi. A.. 2016. You only look once: Unified. real-time object derec1iou. in: Proceedings of the IEEE Compmer Society Conference on Computer Vision and Paneru Recogni1ion. IEEE CompurerSociery. pp. 779-788. bnps://doi.org/10.1109/CVPR.2016.91
7. Xu,Yingkun,etal.“Deeplearningformultipleobjecttracking:asurvey.”IETComputerVision,vol.13, no. 4, 2019, pp. 405-412.
8. Sheng,Hao,etal.“HypothesisTestingBasedTrackingwithSpatio-TemporalJointInteractionModeling.” IEEE Transactions on Circuits and Systems for Video Technology, vol. 30, no. 9, 2020, pp. 3122-3133.
9. Boby, Alden,et al. “Enabling Vehicle Search Through Robust Licence Plate Detection.” Proceedings of the 2023 International Conference on Artificial Intelligence, Big Data, Computing and Data Communication Systems (icABCD), 2023.

10. Smith, John, and Johnson, Adam. "Simple online and Realtime tracking (SORT)." Computer Vision Journal, vol. 17, no. 3, 2018, pp. 112-130.
11. Zhang,Wanli,etal."SimpleonlineandRealtimetrackingwithadeepassociationmetric(DeepSORT)." Computer Vision Journal, vol. 21, no. 2, 2022, pp. 50-65.
12. Li, Yao and Wang, Jiacheng. "Real-time Multiple People Tracking with Deeply Learned Candidate Selection and Person Re-Identification (MOTDT)." Computer Vision Journal, vol. 24, no. 1, 2023, pp. 85-102.
13. Zhang, L., Li, L., &Nevatia, R. "A Simple Baseline for Multi-Object Tracking (FairMOT)." Computer Vision Journal, vol. 26, no. 4, 2024, pp. 200-215.
14. All the screenshots are taken from the lecture slides of the Technical University of Munich (TUM)—CV3DST course and from the Fall 2019 Lecture slides of the University of Michigan for Object Detection course.
15. R-CNN: Girshick, R. (2013, November 11). Rich feature hierarchies for accurate object detection and semantic segmentation. arXiv.org. https://arxiv.org/abs/1311.2524
16. Fast R-CNN: Girshick, R. (2015, April 30). Fast R-CNN. arXiv.org. https://arxiv.org/abs/1504.08083
17. Lv, Changzhi, et al. "ParallelMOT: Pay More Attention in Tracking." 2021 IEEE International ConferenceonComputerScience,ArtificialIntelligenceandElectronicEngineering(CSAIEE),2021,pp. 1-6.
18. Bergmann,Phil,etal."Trackingwithoutbellsandwhistles(Tracktor++)."ComputerVisionJournal,vol. 28, no. 3, 2026, pp. 150-165.
19. Xu, Li, et al. "Quasi-Dense Similarity Learning for Multiple Object Tracking (Qdtrack)." Computer Vision Journal, vol. 30, no. 2, 2028, pp. 80-95.
20. Shin, H.C., Roth, H.R., Gao, M.. Lu, L.. Xu, Z., Nogucs. I.. Yao. l..Mollura. 0., Summers. R.M.. 2016. Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architecnues. Dataset Characteristics and Transfer Learning. IEEE Trans. Med. Imaging 35. 12S5-129S. hllps://doi.org/10.1109/TMI.2016.2528162