IJARIIT INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X

Impact Factor: 6.078

(Volume 10, Issue 5 - V10I5-1320)

Available online at: https://www.ijariit.com

A Bird's Eye View of Neural Networks and Object Detection Models

Chaitya Upadhyay
mypublishedpaper@gmail.com
Army Public School, Khadakwasla, Pune

ABSTRACT

This paper explores the field of Object Detection and the advancements in the same. We delve deep into different neural network-based object detection models, with emphasis on their application and address some of the challenges faced in this field. Object detection has a growing importance in fields like agriculture, manufacturing, security surveillance, autonomous vehicles etc. This paper compares different models: Convolution Neural Networks (CNN) Region-based Convolutional Neural Networks (R-CNN), Fast R-CNN, Faster R-CNNs, You Only Look Once (YOLO) and Single Shot Detectors (SSD) based on robustness, adaptability, and real-time processing capabilities. We see which models are suited for real-time applications and which are suited for feature extraction. Despite significant progress, there are still challenges that are faced in this field.

Keywords: *Object Detection, Artificial Intelligence, Machine learning, RCNN, Fast RCNN, Faster RCNN, YOLO, SSD, Applications of Object detection.*

Introduction

In recent times, advancements in Artificial Intelligence and Machine Learning (AI/ML) have impacted various domains of technology and commerce. One such application of AI/ML is ChatGPT-4 which utilises a large language model (LLM) and can converse with humans by creating content based on context. Another successful AI/ML is speech/text/pattern recognition. In the field of Computer Vision (CV) object detection and tracking utilises AI/ML algorithms extensively and has improved significantly over the past few years. However, there are numerous challenges and research problems which need to be addressed before the technology can be made viable. Connect AI and Object detection and tracking.

Research Objective

In this paper, neural network-based algorithms for object detection are explored. Algorithms are compared based on their limitations and performance, using the methodology listed below.

Research Methodology

To rank the algorithms, we will use the following performance metrics. Compendium of available literature/ literature review.

- **Robustness:** Robustness means that the tracking system can track the target even in complicated conditions such as background clutters, occlusion and illumination variation.
- **Adaptability:** In addition to the environment changes, the target is to changes, such as the complex and sudden movement of the target. To solve this problem, the tracking system must be able to detect and track the current apparent characteristics of the target.
- **Real-time processing of information:** A system that deals with image sequences should have high processing speeds. So there is a need to implement a high-performance algorithm [(Soleimanitaleb et al., n.d.)].

Object tracking can be classified into two main parts: Single Object Tracking and Multi-Object Tracking (MOT). Both are computer vision techniques used for detecting objects and determining their location. Object detection and tracking have numerous applications in varied fields as enumerated below.
- **Agriculture:** Although the techniques of small object detection are in the preliminary stages of development, they can be used to detect pests and other harmful organisms that can negatively affect the harvest.

- Sports: Detection and tracking of sporting objects like shuttles, balls, and players can present the coaches with significant insights for developing newer strategies and tactics for the game. Detailed analysis of the sport, inuding golf, football, badminton, tennis, table tennis etc.
-Security and Surveillance: can be used to monitor and track unauthorised personnel/vehicles in restricted areas.

Autonomous Vehicles: Real-time object detection and tracking is one of the core technologies in autonomous vehicles. Multi-Object tracking (MOT) is not just a combination of many single-object tracking. Albeit, MOT uses different approaches which involve advanced algorithms to predict the trajectories of objects in video frames. Two broad approaches such as the batch method and the online method can be used to track objects based on future frames and current/past frames respectively [1]. Even though substantial research has been done on the topic there are still many challenges in this field such as Illumination variation, the system needs to be smart enough to give consistent results even in different lighting conditions. Background clutter, the system needs to be intelligent enough to differentiate between the subject and the background. For example, in the footage of a football match, the system needs to track the player and reject the audience in the background. Low resolution, the system needs to put out consistent outputs even when the hardware is not up to standard. Occlusion, the system has to track the object even if it is partially covered. [(Soleimanitaleb et al., n.d.)]

Object Detection

In this section of the paper, we would like to discuss some of the commonly used methods for Object Detection. We would also like to discuss some of the flaws with these models.

CNN

Convolution Neural Networks (CNNs) are one of the most famous and utilised architectures for object detection. A CNN model consists of the following layers- input, two convolution layers, two pooling layers and two fully connected layers. Let's take a deeper look into each of these-

Input: Information in an image is stored in the pixels and pixels have a set of values assigned to them.
Pixels have 3 channels red, green and blue (RGB).

Convolution Layer: Convolution Layer is a mathematical combination or in other words a dot product of two functions to produce a third function. In CNN this operation is implemented by a Kernel. The Kernel moves across the input taking the dot product of the matrices and then saving the values to a new matrix, dubbed the feature map. This highlights the features of the input. Each convolution can have multiple kernels producing feature maps of their own.

Pooling Layer: Pooling layers take the most important parts of the image and discard the rest. This helps to speed up calculations. The pooling layer uses a kernel of its own to extract information. For example, a convolution layer gives a feature map of 28x28 pixels, now the job of the pooling layer is to take this 28x28 pixel input and produce an output of half the size (14x14). This reduces computational power and saves time.
This cycle of convolution and pooling layer continues till we get an abstract image a fraction of the size of the original input. This whole cycle is known as feature extraction. Now, with features extracted we still need to classify the object. Here comes the last and final part of the CNN

Fully connected layer: They understand the abstract max pooled layers and classify the object in the input. But even all this is not enough, because the pooling layer whilst they try to reduce computational time they lose a lot of valuable information. It also comes as no surprise to run such a complicated process that requires significant time which is not feasible for real-time applications. Here by real-time applications, we understand applications that require a quick response, for example: autonomous vehicles require millisecond decision-making, and they are required to identify pedestrians, and other automobiles in a split second. That is why an object detection model needs to be efficient and fast. [(Skalski, n.d.)][(Tch, 2017)][(Tch, 2017)][(Fergus, 2016)][(Pröve, 2017)]

Region-based - Convolutional Neural network (R-CNN)

This model does not explore every part of the input, instead, it uses an external algorithm to isolate and propose some regions. R-CNN uses selective search as its external algorithm. The proposals given by the selective search are simply called region proposals. Let's understand the flow of R-CNN. The input image goes through the selective search algorithm and the algorithm suggests some region proposals, now the model extracts this proposal making it suitable to run through a CNN model. This is then classified. However, the bounding box suggested by the selective search is not perfect, For example: in an image of a person the selective search may suggest a bounding box that only covers the upper body, in this case, another branch is introduced which tweaks the bounding box to be perfect.

Let's see how this branch tweaks the bounding box.
Let the bounding box suggested by selective search be **(Px, Py, Ph, Pw)**
And Let the transformed bounding box by the branch be **(Tx, Ty, Th, Tw)**
Now to obtain the final bounding box **(Bx, By, Bh, Bw)**.

We have two ways to achieve this.

The first method is known as Translation.

In this method Bx and By are generated by moving original pigs by Pw*Tw, if the product is negative then the point moves to the left, and if it's positive the point moves towards the right.

$$Bx = Px + PwT \qquad \textbf{(1.1)}$$

Similarly, By is generated by the product of Ph and Th.
Moving on to the second method which is called the log-space scale transform. Here the operation rear scales the width by multiplying the original Pw and exponential of Tw.

$$Bw = Pw\, exp(tw) \qquad \textbf{(1.2)}$$

And similarly, the height can be computed. But even this method was limited by its efficiency, as it was computationally heavy and time-consuming. This method was faster than CNNs but not fast enough for real-time applications [(Grishick et al., 2013, pg 1-3)]

Fast R-CNN

In a R-CNN module, the image is divided into multiple smaller images by selective search and then passed through a CNN module. The idea behind this module is that, in an R-CNN model the image is passed through a selective search algorithm and then the region proposals are passed through a CNN module. In this module the image is passed through a CNN module called the backbone CNN which extracts every feature in the image and then selective search is run providing region proposals and then these region proposals are passed through another CNN module called Pre-Region CNN. Both of these CNN models are not computationally heavy and require minimum effort to run. [(Girshick, 2015)]

Faster R-CNN

It was discovered that a model with a selective search and CNN took 2.3 seconds to compute, but a model with only CNN models only took 0.32 seconds. From this, we conclude that the time-consuming function is the selective search. We have to understand that at the time Fast - R-CNN there was no other option than selective search, but now in this model, the selective search is replaced with another CNN model called the Region Proposal Network (RPN) this in turn replaced the old Fast R-CNN and was simply called Faster- R-CNN. [(Ren et al., 2015)]

SSD

SSD only needs an input image and ground truth boxes for each object during training. In a convolutional fashion, we evaluate a small set of default boxes of different aspect ratios at each location in several feature maps with different scales. For each default box, we predict both the shape offsets and the confidence for all object categories. The SSD approach is based on a feed-forward convolutional network that produces a fixed-size collection of bounding boxes and scores for the presence of object class instances in those boxes, followed by a non-maximum suppression step to produce the final detections. The early network layers are based on a standard architecture used for high-quality image classification (truncated before any classification layers), which we will call the base network2. We then add an auxiliary structure to the network to produce detections with the following key features: Multi-scale feature maps for detection We add convolutional feature layers to the end of the truncated base network. These layers decrease in size progressively and allow predictions of detections at multiple scales. The convolutional model for predicting detections is different for each feature layer (cf Overfeat [4] and YOLO [5] that operate on a single scale feature map) [(Liu et al., 2015)]

YOLO

You Only Look Once is an extremely fast model that doesn't require any fancy region models, as it only uses a single convolution network and simultaneously predicts multiple bounding boxes and class probabilities for those boxes. YOLO trains on full images and directly optimises detection performance. It has multiple benefits over traditional models.

First, YOLO is extremely fast, since it only uses single-frame detection we don't need a complex pipeline.
Second, YOLO reasons globally about the image when making predictions, unlike sliding window and region-based techniques. YOLO sees the entire image while training.
But the only area this model lags is accuracy. It still lags behind some of the state-of-the-art direction systems in accuracy. While it is fast, it struggles to precisely localise some objects, especially small ones. [(Redmon et al., 2015)]

Ranking (Table)

Comparisons	R-CNN	Fast R-CNN	Faster R-CNN	YOLO(You Only Look Once)	SSD (Single Shot Detector)
Score (VOC2007mAp)	66%	70%	73%	66%	74%
Speed (fps)	0.02 fps	0.4 fps	7 fps	21 fps	46 fps
Regional Proposal algorithms	Selective Search	Selective Search	Region Proposal Network (RPN)	N.A.	N.A.

From this, we infer that both YOLO and SSD are arguably the fastest of all the models but the CNN-based models are more accurate. However, SSD excels at both the matrices predicting with an accuracy of 74% in 46fps.[(Grishick et al., 2013, #)][(Girshick, 2015)][(Verma, n.d.)]

Applications Detail

Object detection is a fundamental technology, and its development affects many industries such as

Autonomous Vehicles-

When we talk about object detection the first application that comes to mind is Autonomous vehicles. The best model to use for this application would be YOLO or SSD as they are fairly accurate (table) and work in real-time. It would help us detect pedestrians, cyclists and obstacles on the road, help segment roads into lanes by recognising the markings on the road, and the system would also detect signs (stop signs, speed limit etc) and act accordingly.

Surveillance and Security-

For this application, an R-CNN model is used because R-CNN models have rich feature extraction, are able to handle complex backgrounds, high accuracy and scalability. Intruder detection: surveillance cameras match the features of a blacklisted member in a crowd and alert the authorities. Crowd Monitoring: analyse crowd patterns for unusual behaviour. Facial recognition.

Agriculture

Crop Monitoring: drones observe plants and inform the farmer about plant health pests. Livestock Monitoring: Detects signs of distress and analyses movements and behaviour. Automated Harvesting: Robots equipped with cameras detect the crops and harvest them with minimal human intervention. For these models like YOLO or R-CNN can be used.

Manufacturing

Quality control: Detects defective products or inconsistencies in the assembly line. Safety: detects humans and informs the authorities if certain safety protocols are not being followed. Models used: R-CNN. [(Ren et al., 2015)][(Girshick, 2015)]

Conclusion

In this paper, we saw the advancements in AI, specifically the field of Computer Vision. The field of Object Detection has come far and has many applications in different fields like agriculture, industrial regions, autonomous vehicles etc. This paper explored multiple Neural Network object detection models like Convolution Neural Networks (CNN) Region-based Convolutional Neural Networks (R-CNN), Fast R-CNN, Faster R-CNNs, You Only Look Once (YOLO) and Single Shot Detectors (SSD) highlighting their strengths, applications and limitations. Through analysis and available literature, it is seen that R-CNN-based models are more accurate and better at handling complex backgrounds, it is also seen that R-CNN-based models are used when extreme feature extraction is needed these models are also extremely robust. On the other hand, YOLO and SSD models are extremely fast, and ideal for real-time situations, but may be slightly less accurate and may even struggle in low lighting. These models are paramount to real-time processing and speed. Experiments conducted in past literature prove this as we also see how well SSD and YOLO perform in VOC2007mAp tests coming out as the better alternative to R-CNN models in some cases. Object Detection has had substantial growth in the past decade. However even "state-of the art" models struggle with Illumination variation, Background clutter Low resolution input etc. In conclusion, the future of object detection lies in the advancement of these detection models, balancing speed with accuracy to feed the growing demand for this technology in diverse fields. As these technologies evolve they will no doubt make their way into every household for security and overall improvement in quality of life.

References

[1] C. Eggert, S. Brehm, A. Winschel, D. Zecha and R. Lienhart, "A closer look: Small object detection in faster R-CNN," 2017 IEEE International Conference on Multimedia and Expo (ICME), Hong Kong, China, 2017, pp. 421-426, doi: 10.1109/ICME.2017.8019550. keywords: {Proposals;Companies;Object detection;Feature extraction;Pipelines;Image resolution;Barium;Small objects;Faster R-CNN;RPM;Feature map resolution;Company logos},

[2] Fergus, R. (2016, August 11). An Intuitive Explanation of Convolutional Neural Networks – Ujjwal Karn's blog. Ujjwal Karn's blog. Retrieved August 29, 2024, from https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/

[3] Girshick, R. (2015, April 30). [1504.08083] Fast R-CNN. arXiv. Retrieved August 29, 2024, from https://arxiv.org/abs/1504.08083?so

[4] Grishick, R., Donahue, J., Darrell, T., & Malik, J. (2013, November 11). Rich feature hierarchies for accurate object detection and semantic segmentation.

[5] L. Chen, H. Ai, C. Shang, Z. Zhuang and B. Bai, "Online multi-object tracking with convolutional neural networks," 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 2017, pp. 645-649, doi: 10.1109/ICIP.2017.8296360. keywords: {Target tracking;Feature extraction;Detectors;Training;Computational modeling;Convolutional neural networks;Multi-object Tracking;Convolutional Neural Network;Appearance Model},

[6] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2015, December 8). [1512.02325] SSD: Single Shot MultiBox Detector. arXiv. Retrieved August 29, 2024, from https://arxiv.org/abs/1512.02325

[7] M. M. Faisal et al., "Object Detection and Distance Measurement Using AI," 2021 14th International Conference on Developments in eSystems Engineering (DeSE), Sharjah, United Arab Emirates, 2021, pp. 559-565, doi: 10.1109/DeSE54285.2021.9719469. keywords: {Meters;Roads;Neural networks;Object detection;Cameras;Real-time systems;Safety},

[8] Pröve, P. L. (2017, July 22). An Introduction to Different Types of Convolutions in Deep Learning. Towards Data Science. Retrieved August 29, 2024, from https://towardsdatascience.com/types-of-convolutions-in-deep-learning-717013397f4d

[9] Redmon, J., Divvala, S., Grishick, R., & Farhadi, A. (2015, June 8). [1506.02640] You Only Look Once: Unified, Real-Time Object Detection. arXiv. Retrieved August 29, 2024, from https://arxiv.org/abs/1506.02640

[10] Ren, S., He, K., Grishick, R., & Sun, J. (2015, June 4). [1506.01497] Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. arXiv. Retrieved August 29, 2024, from https://arxiv.org/abs/1506.01497

[11] Shafkat, I. (2018, June 1). Intuitively Understanding Convolutions for Deep Learning | by Irhum Shafkat. Towards Data Science. Retrieved August 29, 2024, from https://towardsdatascience.com/intuitively-understanding-convolutions-for-deep-learning-1f6f42faee1

[12] Skalski, P. (n.d.). Gentle Dive into Math Behind Convolutional Neural Networks. Towards Data Science. Retrieved August 29, 2024, from https://towardsdatascience.com/gentle-dive-into-math-behind-convolutional-neural-networks-79a07dd44cf9

[13] Soleimani Taleb, Z., Keyvanrad, M. A., & Jafari, A. (n.d.). Object Tracking Methods: A Review. 2019 9th International Conference on Computer and Knowledge Engineering (ICCKE), pp. 282-288. 10.1109/ICCKE48569.2019.8964761

[14] Tch, A. (2017, August 4). The mostly complete chart of Neural Networks is explained. Towards Data Science. Retrieved August 29, 2024, from https://towardsdatascience.com/the-mostly-complete-chart-of-neural-networks-explained-3fb6f2367464

[15] Verma, Y. (n.d.). Compare R CNN Models - Fast Vs Faster R CNN. Analytics India Magazine. Retrieved August 29, 2024, from https://analyticsindiamag.com/topics/compare-r-cnn-models/

[16] Y. Wei, N. Song, L. Ke, M. -C. Chang and S. Lyu, "Street object detection/tracking for AI city traffic analysis," 2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), San Francisco, CA, USA, 2017, pp. 1-5, doi: 10.1109/UIC-ATC.2017.8397669. keywords: {Videos;Urban areas;Artificial intelligence;Object detection;Detectors;Trajectory;Cameras;object detection;multi-object tracking;traffic analysis;smart transportation;AI City},