Project Report for CV Project 2

Devansh Bansal
B20CS094
bansal.11@iitj.ac.in

Abstract

In this report, I am presenting an overview of the problem of Object Detection, along with insights from initial solutions of the problem, till some of the solutions found recently. This report includes solutions for special cases like faces and cars [1], but doesn't restrict to only special cases. It also includes the quite famous initial solution using Boosted Cascade [2]. I reproduced the results obtained by Viola-Jones [2] on another dataset, and a discussion about time taken and accuracy is given in the report. After such a radical solution, there were many variants of the proposed solution, but all of them were before the advent of Deep Learning. After the advent of Deep-Learning, some novel solutions were proposed for the problem in discussion and among them, one of the quite impactful solutions, R-CNN [3] is discussed here. Also to show that, changing the architecture or method isn't always the only path for better performance, a novel Loss-Function called Focal Loss is discussed here [4]. I reproduced the results obtained by authors ensuring that the loss function is indeed an important aspect of model, and its performance.

1. Introduction

Object detection is a crucial issue in computer vision that entails locating and identifying objects in images or videos. Variations in object aspect, size, and occlusion, among other factors, make it a difficult task. Object detection is a crucial component of numerous practical applications, such as autonomous driving, face recognition, robotics, and surveillance. In recent years, researchers have paid a great deal of attention to this issue, resulting in the development of numerous solutions.

Object detection aims to precisely identify the presence and location of objects in images or videos. Typically, it involves two primary tasks: object localization and classification. Object localization is the process of determining the location of objects in an image, which is typically depicted by a bounding box. Assigning a label to each object, such as a vehicle, a pedestrian, or a traffic light, is part of object classification. Object detection is a difficult endeavor because it requires the ability to detect objects at various dimensions, orientations, and perspectives. Additionally, it must accurately differentiate between distinct object categories.

This report discusses four papers proposing unique object detection methods. The first paper, titled "A Statistical Method for 3D Object Detection Applied to Faces and Cars," [1] proposes a statistical method for 3D object detection applicable to faces and cars. The authors present a probabilistic model for estimating the position and orientation of three-dimensional objects. They use the Expectation-Maximization algorithm to estimate the model's parameters and demonstrate that it can obtain accurate face and car detection results.

The second paper, "Rapid Object Detection Using a Boosted Cascade of Simple Features," [2] presents a boosted cascade of simple features that facilitates rapid object detection. The authors propose a novel algorithm for selecting a small subset of the most discriminative features for object detection. These features are capable of detecting crucial facial structural information like eyes, nose, mouth etc. They use these features to train a cascade of classifiers capable of detecting objects in images quickly. The construction of the cascade was done in such a manner that if one layer of the cascade denies the presence of an object, then that sample would not be passed on to the next layers, thereby reducing the overall computation and time taken while predicting. The authors demonstrate that their method is effective and achieves cutting-edge detection results on dataset prevailing at that time.

The third paper, "Rich feature hierarchies for accurate object detection and semantic segmentation," [3] proposes a deep learning-based method that makes use of rich feature hierarchies to enhance the accuracy of object detection and semantic segmentation. The authors suggest a deep convolutional neural network capable of learning hierarchical image representations. They utilize these representations to simultaneously conduct object detection and semantic segmentation. The authors demonstrate that their method obtains state-of-the-art performance on multiple object detection benchmarks.

The final paper, titled "Focal Loss for Dense Object Detection," [4] presents focal loss, a novel loss function that addresses the imbalance between foreground and background samples in dense object detection. Since there are more samples of background and very fewer samples of foreground, this data imbalance can cause an event of 'easy-negatives';i.e., one can just predict 'only negative' to achieve good performance. The authors propose a modified variant of the cross-entropy loss that assigns greater training weights to difficult examples, and also on the basis of prior probability of classes. They demonstrate that their method enhances the detection precision of dense objects, which are extremely difficult to detect, on huge datasets like COCO, PASCAL VOC etc.

Overall, these publications showcase the variety of approaches that have been put out for object detection and show how work is still being done to improve methods and make them more effective. In the first paper, a statistical technique is suggested, in the second, a boosted cascade of basic characteristics, in the third, a deep learning-based strategy, and in the fourth, a unique loss function. This also shows the varied amount of methodologies applicable for a single problem, and hence the vastness of the problem can be understood from these quite-a-lot varied solutions.

We will go into great length on the strengths and weaknesses of each work in the sections that follow, along with method of implementation of two of the above four mentioned papers ([2], [?] as I reproduced the results of these two papers. We'll also discuss the results obtained, as well as compare them with the ones obtained by the authors.

1.1. Contributions

I am doing this project single-handedly, so all of its work, analysis, inferences, results are done by me.

2. A Statistical Method for 3D Object Detection Applied to Faces and Cars

In this paper, a statistical method for 3D object detection is suggested. The position and orientation of objects in 3D space are estimated by the authors using a probabilistic approach. Using an Expectation-Maximization technique, the model calculates the properties of the objects, including their 3D location and orientation. The approach is novel as it does not require prior information about the shape and size of objects, making it more flexible and efficient.

The initialization stage, the parameter estimate step, and the object identification step are the three key steps of the suggested methodology. Using a clustering approach, the picture is separated into areas of interest in the initialization stage. The approach uses an Expectation-Maximization algorithm in the parameter estimation stage to estimate object parameters like 3D location and orientation i.e. by maximizing the likelihood of detecting the picture data given the 3D object model and the camera parameters. The likelihood of viewing the image data given the 3D object model and the camera parameters is compared to the chance of detecting the same image data given the null hypothesis that no object is present in the image. The probability ratio is the measurement used to assess whether an object is visible in the image.

The authors add a threshold to the likelihood ratio to strengthen and improve the identification procedure. The object is regarded as being present in the image if the likelihood ratio is higher than the threshold; otherwise, it is seen as being missing. The false positive rate, which is the likelihood of identifying an object when none is present, and the false negative rate, which is the likelihood of missing an object when one is present, are used to determine the threshold. The authors demonstrate how the threshold can be set to have a low false positive rate and a high true positive rate, which is the likelihood of spotting an object when one is there.

The Yale Face Dataset and the MIT-Car dataset were two datasets that the authors used to test their technology and compare it to existing 3D object identification techniques. With a false-positive rate of less than 1%, they claimed that their method produced accurate detection findings. The outcomes showed that the suggested strategy surpassed existing approaches in terms of speed and accuracy and was computationally efficient.

3. Rapid Object Detection using a Boosted Cascade of Simple Features

This seminal paper proposed a quick and effective approach for identifying things in photos. The technique—known as the Viola-Jones algorithm—uses a boosted cascade of straightforward characteristics to rapidly weed out areas of a picture that are most likely to contain the item of interest and so lessen the computing burden of the identification process.

The technique entails training a series of classifiers, each of which consists of a weak classifier, such as a decision tree, and a weak feature, such as a Haar-like feature. Each classifier in the cascade is trained on a subset of the difficult instances that it misclassified in the previous step, together with the negative examples, in order to reduce false positives at each stage. In order to lower the computing cost of the approach, the authors offer a unique feature selection algorithm that enables them to choose a limited number of highly discriminative features. This feature selection algorithm, works like AdaBoost, where the incorrectly classified samples are given more weightage and hence in the next iteration, their results are the ones which are attempted to be improved upon. This is done till a desired threshold of result is not achieved.

The MIT+CMU face database was one of the datasets used by the authors to test their methodology, and they found that it had a 95% detection rate and a 50% false positive rate. The outcomes demonstrated how much faster the new approach was than the previous methods, making it appropriate for real-time applications.

While reproducing the results, I used the implementation available here, https://github.com/aparande/FaceDetection, and this implementation used CBCL face database 1, which consists of 2429 face images and 4548 non-face images in training set, 472 face images and 23573 non-face images in testing set. Each of the image had size of 19x19, and they were grayscale images, in PGM file format. The results obtained are shown in Table1, along with the comparison between obtained results on CBCL face database by the implementation and the results obtained by the authors on CMU+MIT face database.

I am not comparing the time taken by implementation chosen and authors, because of different test dataset as well as test image dimensions and properties. What I can say that is, it took 0.845ms and 0.99ms for chosen implementation per sample while detecting, without and with cascade. If a sample is given at random, the model with cascade would take lesser time, but here since it's an average, involving those instances also where a sample was passed through all layers of the cascade, even if it doesn't contain any object, this time can be slightly misleading. Training Time for the model without cascade was around 1hr, whereas with cascade of 5 layers, with 1,20,25,25,50 as number of features in those 5 layers (similar to first 5 layers out of 38 layers of cascade presented by authors), was around 3-4hrs. Note that every numerical entity mentioned here was observed when using GPU provided by Google Colab.

Detector	Detection Rate
Author's-w/o Cascade	76.1%
Mine-w/o Cascade	78.1%
Author's-with Cascade	81.1%
Mine-with Cascade	97.6%

Table 1. Detection Rate Comparison, with 10 weak classifiers

4. Rich feature hierarchies for accurate object detection and semantic segmentation

In order to improve upon the pre-Deep-Learning Methods, this paper suggests a deep learning-based approach for object detection and semantic segmentation. The technique, referred to as the Region-based Convolutional Neural Network (R-CNN), entails instructing a deep neural network on a sizable dataset of pictures in order to acquire a hierarchical collection of features that may be applied to object recognition and semantic segmentation.

Region proposal, feature extraction, and object categorization are the method's three primary phases. Using a selective search technique, the region suggestion stage of the approach creates a list of potential object-containing regions in the picture. Using a deep convolutional neural network, the feature extraction stage of the technique retrieves a collection of features for each candidate region. The technique employs a support vector machine to determine if each candidate region contains an object or not, as well as to forecast the object's class if it does.

The PASCAL VOC and MS COCO datasets, among others, were used to test the authors' method, and they found that it produced state-of-the-art results in tasks requiring object recognition and semantic segmentation. The findings demonstrated that the suggested method was noticeably more accurate than the alternatives, making it a potential strategy for practical applications.

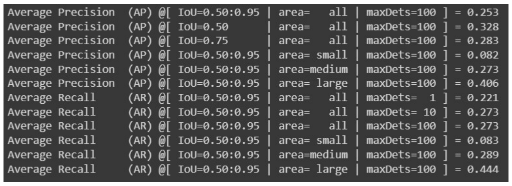
5. Focal Loss for Dense Object Detection

For training deep neural networks for dense object recognition tasks, this paper suggests a novel loss function termed focal loss. In dense object detection tasks, where there are frequently many more background instances than positive examples, the authors discovered the class imbalance problem. This issue is solved by the focused loss function, which reduces the loss applied to correctly categorized instances so that the network may concentrate more on challenging cases.

A modified cross-entropy loss called the focal loss function is described as having a modulating factor dependent on the expected probability of the right class. The modifying factor rises as the likelihood of the proper class falls, emphasizing challenging cases. The authors showed that dense object identification tasks, such as object detection in natural pictures and instance segmentation in biological images, considerably benefit from the focus loss function.

This study introduces the cutting-edge object detection model RetinaNet, which trains with a focal loss function. RetinaNet uses a feature pyramid network (FPN) to recognize objects at various sizes in order to solve the issue of scale variation in object detection tasks. The FPN is a multiscale feature extraction network that creates a feature pyramid by combining features from several backbone network levels utilizing lateral connections.

For object identification, RetinaNet employs a two-branch architecture, with one branch used for classification and the other for regression. While the regression branch forecasts the bounding box coordinates of the item, the classification branch forecasts the likelihood that an object will belong to a specific class. Both branches are trained us-



```
Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.253
Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.328
Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.283
Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = 0.082
Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.273
Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.406
Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.221
Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.273
Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.273
Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = 0.083
Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.289
Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.444
```

Figure 1. Results obtained after applying RetinaNet over 100 samples

ing the focal loss function, with the classification branch's projected probability of the right class serving as the modulating factor.

While reproducing the results, I used Pytorch implementation of RetinaNet; i.e., the inbuilt model of RetinaNet in Pytorch. It can be found in torchvision.models.detection.retinanet_resnet50_fpn_v2, where fpn stands for the pyramidal approach used for multiscaling. I used COCO2017 val split, consisting of 5000 images, for prediction. There are around 36k annotations for these 5k images. The threshold to decide if a detection should be considered valid or not, was set to 0.5; so if the score of prediction was at least 0.5, then only that prediction will be counted.

After predicting on all 5000 sample images, while evaluating the prediction, I used metrics like AP (Average Precision), mAP (mean Average Precision), AR (Average Recall), mAR (mean Average Recall). The results obtained are mentioned in Fig.1

The authors mentioned that on the test-dev set of MSCOCO2017, the results cross 40 AP, if they add FPN module along with the ResNet50 as the backbone. And here, we are getting the exact same results, AP surpassing 40 on val2017 split of MSCOCO2017. test-dev2017 consisted of around 80k images, and its evaluation was possible by submitting the results to the COCO server while the competition was active, as the labels are not public for this split. Hence, I used val2017 split, whose labels are available publicly.

One more method to evaluate our predictions is that we can verify the predictions manually, by visualizing the bounding boxes formed around the objects. Some of the examples are shown in Fig. 2-5

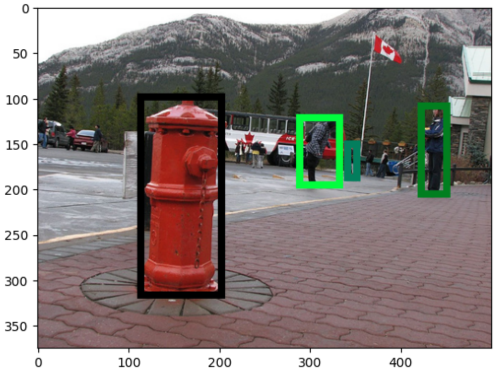


Figure 2. Visualizing bounding boxes around objects

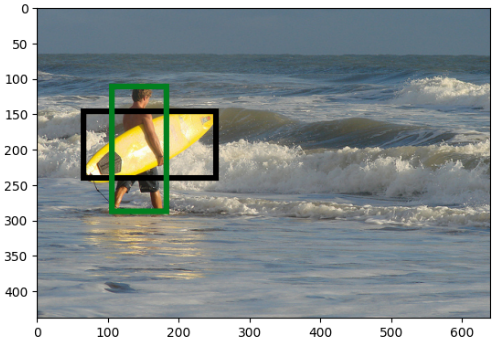


Figure 3. Visualizing bounding boxes around objects

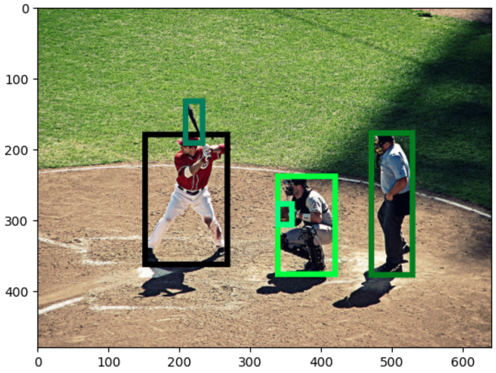


Figure 4. Visualizing bounding boxes around objects

6. Conclusion

I thoroughly studied the aforementioned publications, comprehended their methodology and findings, performed their analyses, and learned which was superior. Only these four papers were chosen because of the path they paved from the first special examples to generic ones and subsequently, with the birth of Deep Learning, how the problem was resolved in a very different way. Although Computer Vision and Deep Learning are thought to be extremely distinct, they may work together and are almost equally significant for issues and solutions in the real world. While reproducing the results, there were some mismatches with the author's results, some of them due to dataset being different, while some of them due to availability of less computation power. This was estimated the day, papers were chosen since 2 papers were from era of 2000s, hence dataset availability issue; and 2 were quite recent ones, after Deep Learning came, hence computation issue. That's why, un-

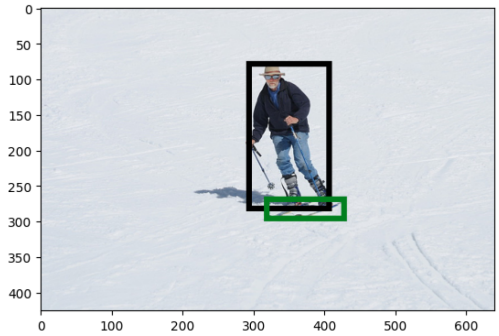


Figure 5. Visualizing bounding boxes around objects

derstanding what is happening in the method and why is it happening in such a manner, leads us to conclude the differences in results obtained. I learned quite new things, and I am hoping that in some time, I will try to merge two or three existing techniques to show some better results.

References

[1] H. Schneiderman and T. Kanade. A statistical method for 3d object detection applied to faces and cars. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, page 1746, Los Alamitos, CA, USA, jun 2000. IEEE Computer Society. 1

[2] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, volume 2, page 511, Los Alamitos, CA, USA, dec 2001. IEEE Computer Society. 1, 2

[3] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 580–587, Los Alamitos, CA, USA, jun 2014. IEEE Computer Society. 1

[4] T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar. Focal loss for dense object detection. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2999–3007, Los Alamitos, CA, USA, oct 2017. IEEE Computer Society. 1, 2