CV PROJECT REPORT: Pedestrian Detection

Aravind Challa
S20190010169, CSE
Email: srisaiaravind.c19@iiits.in
Contributions: Model Training, Darknet53
Sai Satwik
S20190010092, CSE
Email: saisatwik.k19@iiits.in
Contributions: Model Training, Feature Extraction
Sathwik Rao
S20190010189, CSE
Email: sathwik.v19@iiits.in
Contributions: Building Model, Parameter Tuning
Lokesh Mortha
S20190010167, CSE
Email: sreesailokesh.m19@iiits.in
Contributions: Building Model, Class Prediction

Indian Institute of Information Technology Sri City, India

Abstract. Nowadays, the use of visual information technology is growing exponentially. Most of the big IT companies like Google, Microsoft, Amazon, Facebook, etc. are working on visual data analysis. Many startups also came in recent years in the Computer Vision area. Computer Vision also has very strong relevance in Robotics and Industrial Automation. It can be utilized very effectively in smart manufacturing, medical field, biometrics area, etc. It is really necessary for this generation engineers to know how to detect objects, and the traditional approaches used in the past for computer vision. It is necessary to familiarize ourselves with fundamental algorithms and concepts behind modern computer vision algorithms and to know principles of computer vision from the mathematical foundation to detecting objects in real time. Also, we need to know the stages in involved in the process to compute the properties of the three-dimensional world from digital images like reconstructing the 3D shape of an environment, determining how things are moving, and recognizing people and objects and their activities, all through analysis of images and videos, image formation, feature detection, motion estimation, image mosaics, 3D shape reconstruction, object/face detection and recognition, and deep learning. After learning all these from the computer vision course, we decided to create a project implementing all the procedures and algorithms we learnt from the course. We built a pedestrian detection algorithm using YOLOv3 which gave us an average precision of 59.7.

1 Literature Survey

There are several techniques which can be applied for detecting objects and pedestrians and there are several research works going on to improve the accuracy of the models. Pedestrian detection is an essential, significant task in any intelligent video surveillance system. It has an obvious extension to automotive applications due to the potential for improving safety systems. Object detection is considered to be one of the most challenging tasks in the computer vision field.

While there are a handful of different object detection algorithms, in this project, we will have a closer look at YOLO v3 (You Only Look Once). So, first we went through one of the approaches which uses YOLOv3 by Redmon, J. and Farhadi, A. It tells us what's the deal is with YOLO v3. Then it tells how we do. It also trained a new classifier network that's better than the other ones. It will just take you through the whole system from scratch so you can understand it all. Following YOLO9000 our system predicts bounding boxes using dimension clusters as anchor boxes. YOLOv3 predicts an objectness score for each bounding box using logistic regression. New network is a hybrid approach between the network used in YOLO v3, Darknet-53, and that newfangled residual network stuff.

There is one other approach by Yi, Z. *et al.* which is about An improved tiny-yolo v3 pedestrian detection algorithm. It deal with the challenges of the existing real-time pedestrian detection method often loses part of the detection accuracy. The vision system of pilotless automobile technology has always been a difficult point in the field of computer vision, and a reliable vision system will drive the development of pilotless automobile. The low-level feature map contains more information, which is good for retaining details, returning training errors, and improving the accuracy of detection. Compared with the YOLO algorithm, the recognition accuracy for small targets is improved. In this paper, we extracted the pedestrian image from the VOC dataset to form the pedestrian dataset. K-means clustering method is used to compare the IOU scores with different k values. We finally choose the K value of 6 considering the complexity of the model.

2 Methodology

2.1 Bounding Box Prediction

Following YOLO9000 our system predicts bounding boxes using dimension clusters as anchor boxes. The network predicts 4 coordinates for each bounding box. During training we use sum of squared error loss. YOLOv3 predicts an objectness score for each bounding box using logistic regression. This should be 1 if the bounding box prior overlaps a ground truth object by more than any other bounding box prior. If the bounding box prior is not the best but does overlap a ground truth object by more than some threshold we ignore the prediction, following. We use the threshold of .5. Unlike [17] our system only assigns one

bounding box prior for each ground truth object. If a bounding box prior is not assigned to a ground truth object it incurs no loss for coordinate or class predictions, only objectness.

2.2 Class Prediction

Each box predicts the classes the bounding box may contain using multilabel classification. We use a softmax as we have found it is necessary for good performance, we also use independent logistic classifiers. During training we use binary cross-entropy loss for the class predictions. This formulation helps when we move to more complex domains like the Open Images Dataset. In this dataset there are many overlapping labels (i.e. Woman and Person). Using a softmax imposes the assumption that each box has exactly one class which is often not the case. A multilabel approach better models the data.

2.3 Feature Extractor

We use a new network for performing feature extraction. Our new network is a hybrid approach between the network used in YOLOv2, Darknet-19, and that newfangled residual network stuff. Our network uses successive 3×3 and 1×1 convolutional layers but now has some shortcut connections as well and is significantly larger. It has 53 convolutional layers so we call it, Darknet-53. This new network is much more powerful than Darknet19 but still more efficient than ResNet-101 or ResNet-152.

Each network is trained with identical settings and tested at 256×256, single crop accuracy. Run times are measured on a Titan X at 256×256. Thus Darknet-53 performs on par with state-of-the-art classifiers but with fewer floating point operations and more speed. Darknet-53 is better than ResNet-101 and $1.5\times$ faster. Darknet-53 has similar performance to ResNet-152 and is $2\times$ faster. Darknet-53 also achieves the highest measured floating point operations per second. This means the network structure better utilizes the GPU, making it more efficient to evaluate and thus faster. That's mostly because ResNets have just way too many layers and aren't very efficient.

2.4 Model Weights and Training

Weights and cfg (or configuration) files can be downloaded from the website of the original creator of YOLOv3: https://pjreddie.com/darknet/yolo You can also (more easily) use YOLO's COCO pretrained weights by initializing the model with model = YOLOv3(). Using COCO's pre-trained weights means that you can only use YOLO for object detection with any of the 80 pretrained classes that come with the COCO dataset. This is a good option for beginners because it requires the least amount of new code and customization. There are 80 classes that are available using COCO's pretrained weights.

We still train on full images with no hard negative mining. We use multi-scale training, lots of data augmentation, batch normalization, etc. We use the Darknet neural network framework for training and testing.

3 Experimental Results

The YOLOv3 AP does indicate a trade-off between speed and accuracy for using YOLO when compared to RetinaNet since RetinaNet training time is greater than YOLOv3. However, the accuracy of detecting objects with YOLOv3 can be made equal to the accuracy when using RetinaNet by having a larger dataset, making it an ideal option for models that can be trained with large datasets.

The training experimental environment of the improved Tiny YOLOv3 is implemented in a Python library which called Keras, and the Keras is running on the top of TensorFlow. The experiment is trained in the environment of CPU. The total iteration number is 50, the initial learning rate is 0.001, the batch input quantity is 1, the weight attenuation coefficient is 0.0001, and the patience is 3. To prevent the overfitting, 70% of the dataset are used for training, 10% for validation, and 20% for testing. Uses K-means clustering to estimate the size of the anchor boxes for dataset. It achieves 57.9 AP_{50} in 51 ms on a Titan X, compared to 57.5 AP_{50} in 198 ms by RetinaNet, similar performance but 3.8× faster.

	backbone	AP	AP_{50}	AP_{75}
Two-stage methods				
Faster R-CNN+++ [5]	ResNet-101-C4	34.9	55.7	37.4
Faster R-CNN w FPN [8]	ResNet-101-FPN	36.2	59.1	39.0
Faster R-CNN by G-RMI [6]	Inception-ResNet-v2 [21]	34.7	55.5	36.7
Faster R-CNN w TDM [20]	Inception-ResNet-v2-TDM	36.8	57.7	39.2
One-stage methods				
YOLOv2 [15]	DarkNet-19 [15]	21.6	44.0	19.2
SSD513 [11, 3]	ResNet-101-SSD	31.2	50.4	33.3
DSSD513 [3]	ResNet-101-DSSD	33.2	53.3	35.2
RetinaNet [9]	ResNet-101-FPN	39.1	59.1	42.3
RetinaNet [9]	ResNeXt-101-FPN	**40.8**	**61.1**	**44.1**
YOLOv3 608 × 608	Darknet-53	33.0	57.9	34.4

Fig. 1. Result

4 Conclusion

The YOLOv3 enhances the extraction network, and the feature fusion also is strengthened by upsampling and downsampling in two feature map layers.

What's more, the YOLOv3 revised the IoU to CIoU in the loss function. All about this improves the detection accuracy. Although the network model of the improved YOLOv3 is grown, the accuracy has improved.

5 References

Yi, Z., Yongliang, S. and Jun, Z., 2019. An improved tiny-yolo v3 pedestrian detection algorithm. Optik, 183, pp.17-23.

Redmon, J. and Farhadi, A., 2018. Yolo v3: An incremental improvement. arXiv preprint arXiv:1804.02767.

Pathak, D. and Raju, U.S.N., 2021. Content-based image retrieval using feature-fusion of Group Normalized-Inception-Darknet-53 features and handcraft features. Optik, 246, p.167754.

Kim, K.J., Kim, P.K., Chung, Y.S. and Choi, D.H., 2018, November. Performance enhancement of yolov3 by adding prediction layers with spatial pyramid pooling for vehicle detection. In 2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS) (pp. 1-6). IEEE.