Deep Convolutional Neural Networks for Real Time Object Recognition

[1]Jaiswal Shruti R, [2]Keyur Shah

[1]PG Student,Computer Engineering Department ,Silver Oak College of Engineering & Technology, Gujarat, India
2Assistant Professor, Computer Engineering Department ,Silver Oak College of Engineering & Technology, Gujarat, India

ABSTRACT

This paper focuses on Object recognition using YOLO, a real-time object detection model developed to run on portable devices such as a laptop or web camera. The model was first trained on the dataset then on the COCO dataset, when we look at the old .5 IOU mAP detection metric YOLOv3 is quite good. This paper presents a new Object detection method based on transfer learning and sample enhancement under the framework of YOLO network. This method takes full advantages of the real-time feature of YOLO, as well as the enhancement of the generalization ability brought by transfer learning and sample enhancements.

Keyword : - *Object detection, YOLO, Convolution neural networks, deep learning, non-GPU, Camera.*

1. INTRODUCTION

To develop a fast deep neural network for real-time video object Recognition by exploring the ideas of knowledge-guided training and predicted regions of interest. Intelligent video surveillance has become an important research domain in machine learning. System can be applied to identify and Recognition of object, detect abnormal circumstances, and automatic identification in dynamic public environments.[1] Object identification and recognition has been one of the important concepts which are the application of deep learning. Deep learning is the extension of Machine learning which utilizes the representation of data for learning. Recognition of objects using Deep Neural Networks is an area of research that is actively growing. Along with the wide deployment of video object recognition system, it is becoming increasingly important to extract the useful information from the video system automatically. However, the existing system do not supply the target identification function especially for the given scenario.

In this paper, a deep learning based framework is proposed to Recognition the given objects from video. In the video system, the targets change easily in different scenes with different lighting or shadow. The traditional object detection method in the case of complex scenes, cannot correctly identified the targets especially multiply types in real time [2]. Humans glance at an image and instantly know what objects are in the image, where they are, and how they interact. The human visual system is fast and accurate, allowing us to perform complex tasks like driving with little conscious thought. Fast, accurate, algorithms for object detection would allow computers to drive cars in any weather without specialized sensors, enable assistive devices to convey real-time scene information to human users, and unlock the potential for general purpose, responsive robotic systems. Current detection systems repurpose classifiers to perform detection. To detect an object, these systems take a classifier for that object and evaluate it at various locations and scales in a test image. Systems like deformable parts models use a sliding window approach where the classifier is run at evenly spaced locations over the entire image [10]. More recent approaches like R-CNN use region proposal methods to first generate potential bounding box After classification, post-processing is used to refine the bounding box, eliminate duplicate detections, and rescore the box based on other objects in the scene[13]. These complex pipelines are slow and hard to optimize because each individual component must be trained separately. We reframe object detection as a single regression problem, straight from image pixels to bounding box coordinates and class probabilities.

Using our system, you only look once (YOLO) at an image to predict what objects are present and where they are. A single convolutional network simultaneously predicts multiple bounding boxes and class probabilities for

those boxes. YOLO trains on full images and directly optimizes detection performance. This unified model has several benefits over traditional methods of object detection. First, YOLO is extremely fast. Since we frame detection as a regression problem we don't need a complex pipeline. We simply run our neural network on a new image at test time to predict detections. Previous methods, such as You-Only-Look-Once (YOLO) [10] and Regional-based Convolutional Neural Networks (R- CNN) [11], have successfully achieved an efficient and accurate model with high mean average precision (mAP), however, their frames per second (FPS) on non-GPU computers render them useless for real-time use. In this paper, YOLO-LITE is presented to address this problem Using the You Only Look Once (YOLO) [10] algorithm as a starting point, YOLO is an attempt to get a real time object detection algorithm on a standard non-GPU computer.

1.1 INTRODUCTION OF CONVOLUTIONAL NEURAL NETWORK

A convolutional neural network is an architectural extension of the feed forward artificial neural networks with multiple layers. A neural network generally consists of 3 basic layers which are the input layer, hidden layer and the output layer. The nodes in these layers are described as neurons. However, this has not been effective because the weights of the neurons have to update frequently. We implement a convolutional neural network based on VGG net architecture to identify objects in real-time[5] We skip frames intelligently to prevent output lag while maintaining precision. Region proposal techniques are used to localize object region and pass it to the CNN to reduce computation overhead.

Architecture of Convolutional Neural Nets (CNNs) are inspired by biological visual cortex of mammals. Visual cortex processes images differently than we think, it first captures low level features of an image or frames and then goes for whole picture. But is processes so fast that we can't observe the process. Here, some step to defined CNN: Step 1 :Prepare dataset of image, Step 2:Convolution, Step 3: Pooling, Step 4 :Normalization (ReLu): Rectified Linear Unit, Step 5: Probability Conversion, Step 6: Choose Most label.

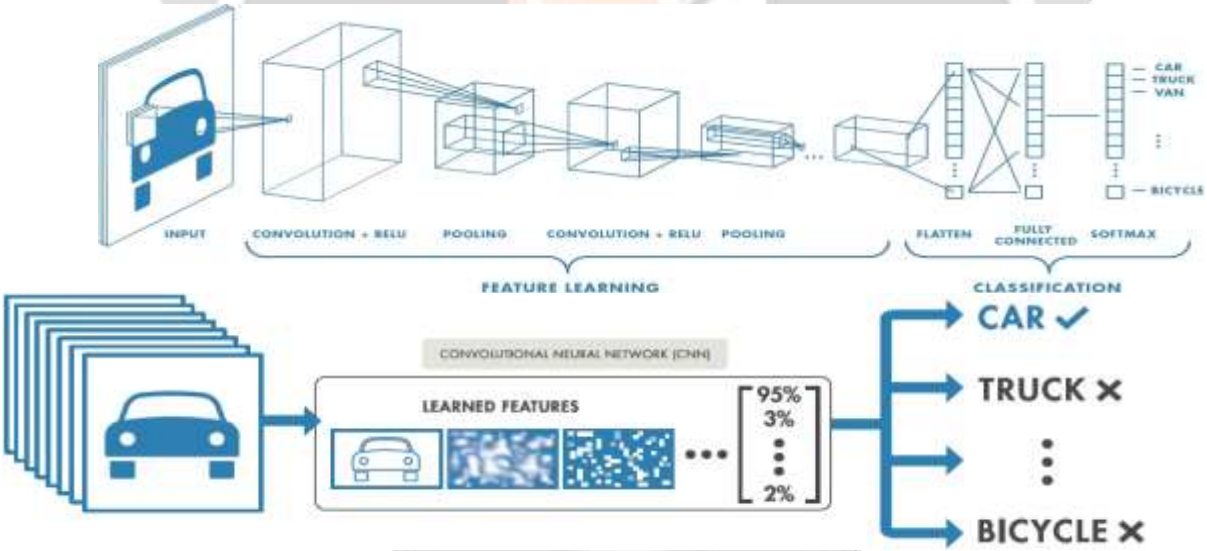

Fig.1. http://www.pyimagesearch.com/2018/11/12/yolo-object-detection-with

1.2 YOLO OVERVIEW

So here's the deal with YOLOv3: We mostly took good ideas from other people. We also trained a new classifier network that's better than the other ones. We'll just take you through the whole system from scratch so you can understand it all.

Bounding Box Prediction : Following YOLO our system predicts bounding boxes using dimension clusters as anchor boxes [15]. The network predicts 4 coordinates for each bounding box, tx, ty, tw, th. If the cell is offset from the top left corner of the image by (cx, cy) and the bounding box prior has width and height pw, ph, then the predictions correspond to:

$$bx = \sigma(tx) + cx \quad (1)$$

$$by = \sigma(ty) + cy \quad (2)$$

$$b_w = p_w e^{t_w} \quad (3)$$

$$b_h = p_h e^{t_h} \quad (4)$$

During training we use sum of squared error loss. If the ground truth for some coordinate prediction is ˆ t* our gradient is the ground truth value (computed from the ground truth box) minus our prediction: ˆ t*−t*. This ground truth value can be easily computed by inverting the equations above. YOLOv3 predicts an object 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.

Class Prediction: Each box predicts the classes the bounding box may contain using multi label classification. We don't use a softmax as we have found it is unnecessary for good performance, instead we simply 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 [7]. 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 multi label approach better models the data.

Predictions Across Scales: YOLOv3 predicts boxes at 3 different scales. Our system extracts features from those scales using a similar concept to feature pyramid networks [8]. From our base feature extractor we add several convolutional layers. The last of these predicts a 3-d tensor encoding bounding box, objectness, and class predictions. In our experiments with COCO[10].we predict 3 boxes at each scale so the tensoris N×N ×[3∗(4+1+80)] for the 4 bounding box offsets, 1 Objectness prediction, and 80 class predictions.

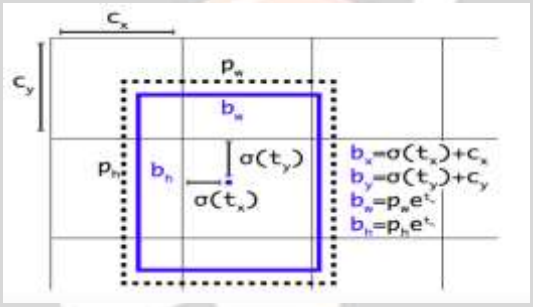

Fig. 2. [15]

Next we take the feature map from 2 layers previous and up sample it by 2×. We also take a feature map from earlier in the network and merge it with our upsampled features using concatenation. This method allows us to get more meaningful semantic information from the up sampled features and finer-grained information from the earlier feature map. We then add a few more convolutional layers to process this combined feature map, and eventually predict a similar tensor, although now twice the size. We perform the same design one more time to predict boxes for the final scale. Thus our predictions for the 3rd scale benefit from all the prior computation as well as fine grained features from early on in the network. We still use k-means clustering to determine our bounding box priors. We just sort of chose 9 clusters and 3 scales arbitrarily and then divide up the clusters evenly across scales. On the COCO dataset the 9 clusters were:
(10×13),(16×30),(33×23),(30×61),(62×45),(59× 119),(116×90),(156×198),(373×326).

Feature Extractor : We use a new network for performing feature extraction. Our new network is a hybrid approach between the network used inYOLOv2, 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. 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-101and1.5×faster. Darknet-53has similar performance to ResNet-152 and is 2×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. Training : We still train on full images with no hard negative mining or any of that stuff. We use multi-scale training, lots of data augmentation, batch normalization, all the standard stuff. Fig. 3

Fig. 4

2 . COMPARISON TO OTHER DETECTION SYSTEMS

Object detection is a core problem in computer vision. Detection pipelines generally start by extracting a set of robust features from input images (Haar [25], SIFT [23], HOG [4], convolutional features [6]). Then, classifiers [35, 21, 13, 10] or localizers [1, 31] are used to identify objects in the feature space. These classifiers or localizers are run either in sliding window fashion over the whole image or on some subset of regions in the image [34, 15, 38]. We compare the YOLO detection system to several top detection frameworks, highlighting key similarities and differences.

Deformable parts models. Deformable parts models (DPM) use a sliding window approach to object detection [10]. DPM uses a disjoint pipeline to extract static features, classify regions, predict bounding boxes for high scoring regions, etc. Our system replaces all of the disparate parts with a single convolutional neural network. The network performs feature extraction, bounding box prediction, non-maximal suppression, and contextual reasoning all concurrently. Instead of static features, the network trains the features in line and optimizes them for the detection task. Our unified architecture leads to a faster, more accurate model than DPM.

R-CNN. R-CNN and its variants use region proposals instead of sliding windows to find objects in images. Selective Search [34] generates potential bounding boxes, a convolutional network extracts features, an SVM scores the boxes, a linear model adjusts the bounding boxes, and non-max suppression eliminates duplicate detections. Each stage of this complex pipeline must be precisely tuned independently and the resulting system is very slow, taking more than 40 seconds per image at test time [14]. YOLO shares some similarities with R-CNN. Each grid cell proposes potential bounding boxes and scores those boxes using convolutional features. However, our system puts spatial constraints on the grid cell proposals which helps mitigate multiple detections of the same object. Our system

also proposes far fewer bounding boxes, only 98 per image compared to about 2000 from Selective Search. Finally, our system combines these individual components into a single, jointly optimized model.

Other Fast Detectors Fast and Faster R-CNN focus on speeding up the R-CNN framework by sharing computation and using neural networks to propose regions instead of Selective Search [14] . While they offer speed and accuracy improvements over R-CNN, both still fall short of real-time performance. Many research efforts focus on speeding up the DPM pipeline [5]. They speed up HOG computation, use cascades, and push computation to GPUs. However, only 30Hz DPM actually runs in real-time. Instead of trying to optimize individual components of a large detection pipeline, YOLO throws out the pipeline entirely and is fast by design. Detectors for single classes like faces or people can be highly optimized since they have to deal with much less variation. YOLO is a general purpose detector that learns to detect a variety of objects simultaneously.

OverFeat. Sermanet et al. train a convolutional neural network to perform localization and adapt that localizer to perform detection [31]. Over-Feat efficiently performs sliding window detection but it is still a disjoint system. Over- Feat optimizes for localization, not detection performance. Like DPM, the localizer only sees local information when making a prediction. Over-Feat cannot reason about global context and thus requires significant post-processing to produce coherent detections.

3. CONCLUSION

In this paper, we proposed a new method to detect obstacles in the indoor environment that introduces deep learning technology and the camera to recognize the obstacle and perceive its information. According to the recognition result and depth map, the object filter is applied to remove the unconcern obstacle. To demonstrate the performance of our method, different types of scene, including pedestrian,chair, book and so on, are demonstrated. The experimental results prove the effectiveness of our detection algorithm. In the future, we will further investigate how to detection the object in real time base on the camera and the deep learning technology. This paper adopts the deep learning method to improve the network structure of YOLO v3 and obtains the YOLO network model. It has achieved good results in pedestrian detection. Experiments show that this algorithm improves the accuracy of pedestrian detection. The number of detection frames can reach 25 frames/s , basically meeting the requirements of real-time performance. In the future, we intend to integrate more pedestrian context features to improve the accuracy of pedestrian detection.

4. REFERENCES

[1] Analogy. Wikipedia, Mar 2018. 1

[2] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) chal- lenge.
International journal of computer vision, 88(2):303– 338, 2010. 6

[3] C.-Y. Fu, W. Liu, A. Ranga, A. Tyagi, and A. C. Berg. Dssd: Deconvolutional single shot detector. arXiv preprint arXiv:1701.06659,
2017. 3

[4]D.Gordon,A.Kembhavi,M.Rastegari,J. Redmon, D. Fox, andA.Farhadi. Iqa: Visual question answering in inteactive environments.arXiv preprint arXiv:1712.03316, 2017. 1

[5] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learn- ing for image recognition. In Proceedings of the IEEE con- ference on computer vision and pattern recognition, pages 770–778, 2016. 3

[6] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A.Fathi,I.Fischer,Z.Wojna,Y.Song,S.Guadarrama,etal. Speed/accuracy trade-offs for modern convolutional object detectors. 3

[7] I. Krasin, T. Duerig, N. Alldrin, V. Ferrari, S. Abu-El-Haija, A. Kuznetsova, H. Rom, J. Uijlings, S. Popov, A. Veit, S. Belongie, V. Gomes, A. Gupta, C. Sun, G. Chechik, D. Cai, Z. Feng, D. Narayanan, and K. Murphy. Open-images: A public dataset for large-scale multi-label and multi-class image classification. Dataset available from https://github.com/openimages, 2017.

[8] M.Scott. Smartcameragimbalbot scanlime:027,Dec2017. 4

[9] A. Shrivastava, R. Sukthankar, J. Malik, and A. Gupta. Be- yond skip connections: Top-down modulation for object de- tection. ArXiv preprint arXiv:1612.06851, 2016. 3

[10] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. 2017. 3
[11] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Doll´ar. Focal loss for dense object detection. arXiv preprint arXiv:1708.02002, 2017. 1, 3, 4 [10] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ra- manan, P. Doll´ar, and C. L. Zitnick. Microsoft coco: Com-monobjectsincontext. InEuropeanconferenceoncomputer vision, pages 740–755. Springer, 2014. 2 [11] W. Liu, D. Anguelov, D. Erhan, C.Szegedy, S. Reed, C.- Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector. In European conference on computer vision, pages 21–37.Springer, 2016. 3
[12] I. Newton. Philosophiae naturalis principia mathematica. William Dawson & Sons Ltd., London, 1687. 1
[13] J. Parham, J. Crall, C. Stewart, T. Berger-Wolf, and D. Rubenstein. Animal population censusing at scale with citizen science and
photographic identification. 2017. 4
[14] J. Redmon. Darknet: Open source neural networks in c. http://pjreddie.com/darknet/, 2013–2016. 3
[15] Dalal,N., Triggs, B.:Histograms of oriented gradients for human detection. In:CVPR.(2005).
[16] Felzenszwalb, P., McAllester,D., Ramanan, D.A.:discriminatively trained, multiscale,deformable part model.In:CVPR.(2008).
[17]Dollár P, Tu Z, Perona P, et al. Integral Channel Features[C]//British Machine Vision Conference, BMVC,2009:1-11.
[18] F. Utaminingrum et al.A laser-vision based obstacle detection and distance estimation for smart wheelchair navigation," 2016 IEEE Inter- national Conference on Signal and Image Processing (ICSIP), Beijing, 2016.