OBJECT DETECTION WITH THE HIGH-FREQUENCY CHANGE OF OBJECTS CLASSES

Lizhi Lou [1], Shen Zhang [1] , Shaoming Zhang [1, *]

[1] College of Surveying and Geo-Informatics, TONGJI University, Siping Road, Shanghai, 200092 China - (llz, zhangshen2018, Sheva2003)@tongji.edu.cn

KEY WORDS: Object Detection, Classes Change, Two Stage Detection, Datasets Production, PVA-Net, GoogLeNet

ABSTRACT:

With the development of deep learning, object detection has a significantly improvement. But most of algorithms only focus on the detection accuracy and speed, they do not consider the difficulty of making training datasets and the time consumption of training detection models, which will have a bad influence on the performance of detection model when the class of objects change in high frequency. This paper proposes a method named double network detection (DN detection), it can improve the efficiency of making training datasets and shorten the time of training model. At the same time, the experiment shows that the DN detection have a good performance in accuracy and speed.

1. INTRODUCTION

In recent years, deep learning has achieved remarkable achievements in all aspects with its powerful learning ability. And the object detection algorithm based on deep learning has a satisfying performance on speed and accuracy. However, the powerful learning ability of deep learning needs to be supported by massive data. At present, the production of training datasets for neural networks mainly depends on manual annotation of regions of interest and class, such as PASAL VOC 2012 (Everingham, M et al., 2012) and COCO (Lin, Z. et al., 2014). What's more, training object detection network model takes a long time and occupies much computing resources. In the actual application, the performance of the object detection algorithm will be badly affected by inefficiency of manual data annotation, longer model training time and consumption of much computing resources when the object classes change rapidly.

At present, the main methods of object detection are divided into two categories, one-stage detection and two-stage detection (Zou, Z et al., 2019). One-stage detection algorithm directly acts on the image, and the position in image and the class of the object are obtained by regression. The main algorithms include a set of YOLO detectors proposed by Joseph Redmon (Redmon J. et al., 2016) (Redmon J. et al., 2017) (Redmon J. et al., 2018) and the SSD detector proposed by Wei Liu (Liu, W et al., 2015). Then T.-Y. Lin (Lin, T. Y., 2017) proposed the RetinaNet detector that gets 39.1% mAP on COCO at 13 fps with ResNet101-FPN on Nvidia M40 GPU. The two-stage detection method obtains the final detection result by the selection of candidate regions of interest, bounding box regression and class prediction. The main algorithms are R-CNN (Girshick, R. B. et al., 2013) and Fast-R-CNN (Girshick, R. B. et al., 2015) proposed by Ross Girshick, Faster R-CNN (Ren, S., He, K. et al., 2015) proposed by Shaoqing Ren. Faster R-CNN gets 70.4% mAP on PASAL VOC 2012 at 5 fps and 36.2% with ResNet101-FPN (Lin, T. Y., 2017). Also, PVA-Net (Kim, K. et al., 2016) improves Faster-RCNN that gets 82.5% mAP on PASAL VOC2012, while taking only 46ms/image on NVIDIA Titan X GPU. In the competition between the two methods, the one-stage detection algorithm is more efficient, the detection speed is faster, and the detection accuracy once caught up with the two-stage detection algorithm until Li (Li, Z. et al., 2018) proposed Light-Head R-CNN that gets 39.5% mAP on COCO. What's more, Zhiyang Yu (Zhiyang Yu et al., 2017) proposed a two-stage full convolutional neural network for industrial flaw detection, which adopted a detection network for rough detection and then a fine classification method to obtain stable detection results, further verifying the stability of the two-stage method. Up to now, the research in object detection has plenty of outcomes. However, many of them rely on open source object detection datasets without considering the difficulty of datasets production. When the number of object classes are fixed and known, training datasets can be collected and made by manual annotation in advance. However, when the object classes are in the state of growth or changes and the detection model needs to be quickly able to detect new class object, the production of datasets and the frequent update of models will become the bottleneck of object detection algorithms, as well as the consumption of the time and computing resources for model training.

The two-stage detection algorithm is divided into object localization and classification, which is as same as the process of training datasets production. It motivates the author of this paper. This paper proposes a method that will reduce the model training time, speed up the model iteration and maintain high accuracy in the context of high frequency classes updates. We completely separate the object detection into two steps. The first step is object detection, which mainly extracts the object from the background. The second step is object classification based on the detection result. The detection model is focused on the major category, so it has a powerful generalization ability to locate objects. When the object class changes (increase or complete change) but still belong to the same major category, the object can still be accurately detected and we only update classification models. And, in making detection model training datasets, the object detection can replace the manual annotation of the bounding box. In making classification training datasets, we can visually discriminate and classify the object image from detection training datasets. By this way, we improve the efficiency of the entire process including datasets production, model training and object detection with less human participation. As far as we know all, nobody does this work in this area until now.

* Corresponding author

2. METHODOLOGY

In this paper, the objects with similar appearance features are divided into a same major category (such as dogs), and then subdivided into types (such as Corgi and Shepherd Dogs) under the major category. And the number of the major categories are remained basically stable even if object classes change. In the first step of detection, the objects in the image are located and coarsely classified into a major category. In the second step of classification, the results of the coarse classification of the detection model are input into the corresponding classification network models for fine classification. Under the premise of the high detection accuracy, this method strips classes change from the detection to the classification with reducing the cost of datasets production and the difficulty of model training. Now, we will introduce the DN detection in details. The detection algorithm framework in this paper is shown in Figure1.

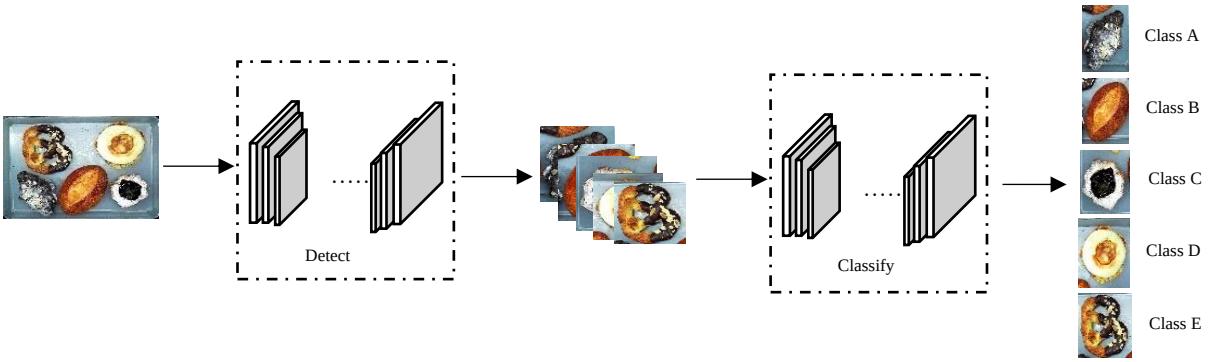

Figure 1. DN detection architecture. The image is input to the detection network to obtain the object location and the major category firstly, and then the obtained object is cropped and input to the classification network to obtain the final result.

2.1 Coarse Detection

The purpose of this stage is to extract the object from the background and roughly classify it into a major category. We adopt PVA-Net for object detection. Faster-RCNN uses neural networks to perform feature extraction, region proposal and ROI classification, and achieves the state of art in terms of accuracy. However, it consumes large computing resources in feature extraction, resulting in a slower detection with only 5fps. Based on Faster-RCNN, PVA-Net optimizes the feature extraction part and proposes a new feature extraction network. The specific method is that combining C.ReLU, Inception and HyperNet forms a basic feature extraction network. Also, based on the observation of feature extracted from convolutional networks, C.ReLU achieves twice the speed without loss of accuracy by symmetric parameter inversion. And the Inception structure is combined with convolution kernels of multiple receptive fields to make it multi-scale object detection. While HyperNet provides multi-scale information by combining fine-grained features with coarse-grained features. The framework of the algorithm is shown in the Figure 2. The final result is obtained by combining feature extraction, RPN network and classification regression.

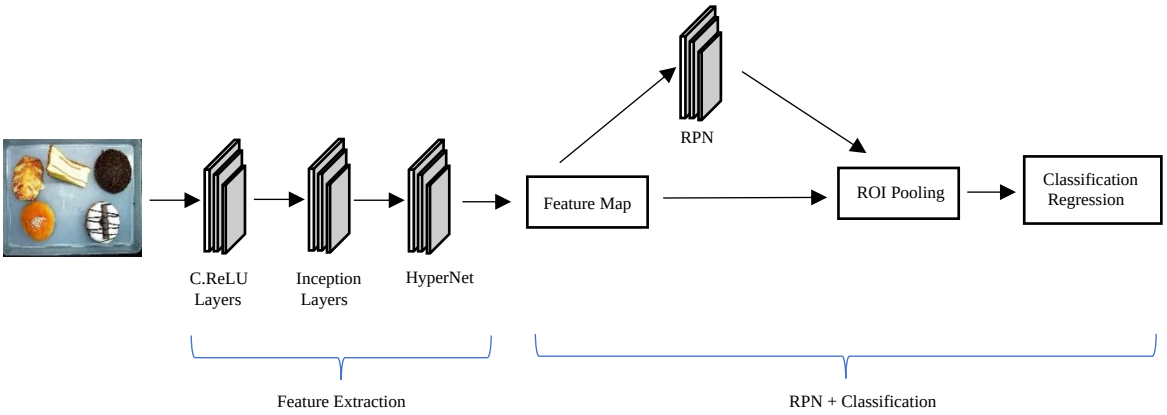

Figure 2. PVA-Net detection framework

2.2 Fine Classification

The purpose of this stage is to finely classify the output from the detection network in the previous stage. This stage uses the GoogLeNet (Szegedy, C. et al., 2015) classification network. Its unique network structure enables GoogLeNet's classification result in the ImageNet (Li, L.-J. et al., 2009) dataset (including 1000 classes objects) to achieve an accuracy rate of 44.5%., defending champion in the competition. Different from the previous methods to increase the network accuracy by increasing the network width and depth, GoogLeNet is designed with the inception structure (Figure 3), which aggregates sparse matrices into dense sub-matrices to improve computing performance, which avoid sparse connections that the computing power of the computing hardware cannot play the best performance. At the

same time, two auxiliary loss function branches are added to avoid the problem of gradient disappearance caused by the network being too deep. The network reaches the best classification with only 5 million parameters at that time. In the scenario that the object classes are update at a high frequency, the network can provide sufficient classification capabilities for the classification stage to ensure that even when the number of object classes accumulate to a large value, it still has high accuracy for fine classification.

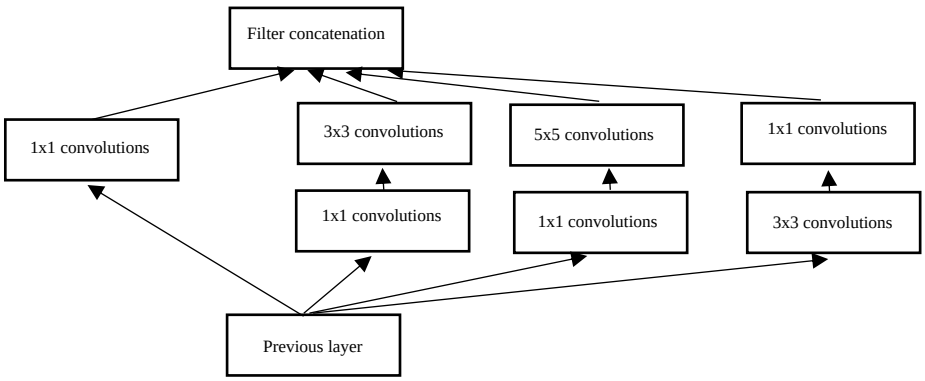

Figure 3. GoogLeNet inception architecture

2.3 Training and Datasets Production

At the first, we will train the detection network and the classification network separately, and make training datasets for both. Of course, the detection model only detects the object in a major category (such as the car only), and the classification model is able to get object's type (trucks, vans, etc in cars) under the major category. After the training of the initial models are both completed, because the detection model has certain generalization ability to detect object in the major category, low-frequency updating can be achieved as the classes of detected object continue to increase (such as pickup trucks, bus, and commercial vehicles). What's more, when making new class training datasets, we only need to use the initial detection model to locate and crop the object image from new picture, and merge them with the previous classification datasets, finally update the classification model. In this way, we can have an ability to detect new class objects rapidly. For traditional detection, if the classes of objects increase, in order to ensure that the model has sufficient discrimination ability between new and old object, the new objects and the old objects need to be mixed and labelled in the same picture when making training datasets. When there are many new and old objects with similar features, the datasets production process will be extremely tedious and time-consuming. The method in this paper transfers the workload of increasing objects types to the classification, and the training datasets of the classification model only needs to be superimposed.

The object detection model has a high utilization rate due to the major categories relatively stable. The high frequency changes in the objects type is transferred to the object classification, so the classification model is in a state of high frequency replacement. Compared with the classification model, the object detection model's cost in time and manpower for making the training datasets increase multiples. At the same time, the classification model is much faster than the detection model in the convergence speed and shorter in iteration time of model training (see Table 2). The DN detection method in this paper improves the efficiency of the entire process so that the algorithm can meet the actual production requirements in a cost-effective manner, which is the biggest advantage of this method.

3. EXPERIMENT

3.1 Datasets

In this experiment, the bread is selected as the major category, because the bread types will change frequently in a short time in a bakery. We have 65 types under the major category of bread (Figure 4). In the scenario of high-frequency update in the classes, the update can be divided into two cases: the object class changes completely and increases only. So, we divide the datasets into three groups, the type number of which are 24, 41, and 65 (24 + 41). The average number of images for each type in the bread is 150. The ratio between test datasets and training datasets is 1:9. Firstly, in order to obtain the ground truth of the experimental datasets, we manually label the image to obtain their true class and location (Figure 5), and crop the image by the annotation to get classification datasets of each class for subsequent experiments.

Figure 4. All types of bread

Figure 5. Data Annotation

3.2 Experiment Platform and Parameters

The operating system of the hardware platform in this experiment is Ubuntu16.04, the GPU is NVIDIA GTX Titan X, the CPU is Intel Core i7-8700k CPU @ 3.70GHz. Caffe is used as the deep learning framework, and CUDA8.0 and cuDNN5.0 are used for model acceleration training.

The model training in this paper include detection network model training and classification network model training. The training parameters for the detection network model are set as follows, batch-size is 256, max-iteration is 50,000, base learning rate is 0.0005, learning rate adjustment strategy is "step", gamma coefficient is 0.1, weight of last gradient update (momentum) is 0.9, and the weight decay coefficient (weight decay) is 0.0002. For the classification model, the batch-size is 256, the max-iteration is 5000, the base learning rate is 0.001, the learning rate adjustment strategy is "step", the gamma coefficient is 0.96, the last gradient update weight (momentum) is 0.9, and the weight decay coefficient is 0.0002

3.3 Metrics

mAP: The evaluation metric of the detection model is the mean AP(average precision) value of all classes, and the mAP formula is

$$mAP = \frac{1}{|Q_R|}\sum_{q \in Q_R} AP(q) \tag{1}$$

where AP = average precision of a certain class
Q_R = the number of test datasets classes
q = test dataset of a certain class

and P (precision) formula is

$$P = \frac{TP}{TP+TN} \tag{2}$$

where TP = the number of true positive detection results
TN = the number of true negative detection results

Accuracy: The evaluation metric of the classification model is the top1 accuracy, and the true classification is that the highest score classification result is true. The top1-accuracy formula is

$$acurrcy = \frac{\sum_{i=0}^{n} TP_i}{\sum_{i=0}^{n} P_i} \tag{3}$$

where P_i = the number of class i test dataset
TP_i = the number of the true classification in class i

3.4 Accuracy and Speed

In this experiment, we compare the traditional two-step detection algorithm with DN detection in this paper in the detection accuracy, speed and training time. To control variable, we use PVA-Net as the traditional two-stage detection algorithm. The three groups datasets which the number of types is 24,41,65 are compared with each other by the two methods. The result is shown at Table1.

Method	mAP(24)	mAP(41)	mAP(65)	Time(FPS)
Traditional detection	99.56%	99.95%	99.44%	30
DN Detection	96.66%	94.78%	96.66%	13

Table 1. The accuracy and speed of traditional detection and DN detection

Model	Class 24	Class 41	Class 65
Detection	5.5h	4.5h	4.5h
Classification	0.10h	0.35h	0.37h

Table 2. The training time of detection model and classification model in DN detection

According Table 1, we can see that the accuracy of the DN detection method is weaker than that of the traditional detection method. Because the traditional two-stage detection method shares features between the detection and classification stage, it ensures the consistency of detection and classification. At the same time, because the dual networks need to repeatedly extract image features in performing detection and classification separately, the speed decreases. However, the detection accuracy is only reduced by 3% to 4%, the average detection accuracy is about 95%, and the speed is 13 fps when just one object is detected. Therefore, under the requirement to be able detect new types objects rapidly, the method in this paper has a certain practicality and feasibility. As can be seen from the comparison of the model training time in Table 2, the training time of the classification model is far lower than the training time of the detection model, and the time is shortened by more than 10 times. What's more, considering the difference between making the training datasets of the two models, it can be concluded that the method of this paper has significant advantages in the object detection of various classes and high-frequency update, and achieves good detection results at lower human-power and time costs.

Table 3 shows the accuracy of the detection model and the classification model in the three groups of experiments for DN detection. For the detection model, it can be seen that the accuracy of the detection model suddenly decreases from the first group experiments to the second. After analysing, we find the main reason is that the detection model has more false detections on the test datasets (Figure 6 and Figure 7). This problem will be studied in subsequent studies. However, the performance of the classification model is relatively stable with the update of objects types. Therefore, the stability of classification performance will ensure the detection stability when there are too many objects types with frequent updates.

Model	Class 24	Class 41	Class 65
Detection (mAP)	99.87%	90.11%	90.88%
Classification (Top1-Accuracy)	99.58%	99.76%	99.31%

Table 3. The accuracy of detection model and classification model in DN detection

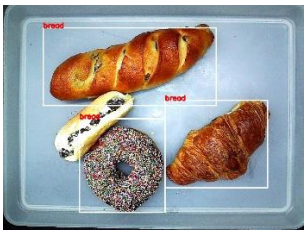

Figure 6. The missed detection image in detection stage

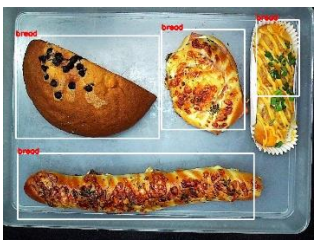

Figure 7. The false positive detection image in detection stage

3.5 Generalization Ability

In order to verify the generalization ability of the detection model, which the detection model can still achieves a good performance when the object types completely change or increase after the first initialization completed, the following experiment is carried out. In this experiment, the detection model of first group including 24 types bread is tested with the classification models of other two group including 41 and 65 types bread, and the experiment results are shown in Table 4. As can be seen from the table, in the second and third groups experiments, the accuracy of the DN detection method in this paper has reached more than 80% without training model detection. So, the detection model has a certain generalization ability.

Method	mAP(24)	mAP(41)	mAP(65)
DN Detection	96.87%	80.19%	83.26%

Table 4. The generalization ability test result for DN Detection

4. CONCLUSIONS

In this paper, we propose a DN detection method to solve the problem that the traditional detection model cannot be updated rapidly under the scenario of high-frequency change in the object classes. It transfers the pressure of high frequency classes update to the update of the classification network model, which reduces

the training time of the model and the difficulty of making training datasets. Our method improves the efficiency of datasets production and model training by at least 10 times. In addition, although the detection model in this paper has certain generalization capabilities, the detection accuracy is still reduced. Future research will be conducted on this issue.

REFERENCES

Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., Zisserman, A., 2012. The pascal visual object classes (VOC) challenge 2012. *International Journal of Computer Vision, 88*(2), 303–338.

Kim, K.-H., Cheon, Y., Hong, S., Roh, B.-S., & Park, M. 2016. PVANET : Deep but Lightweight Neural Networks for Real-time Object Detection. *ArXiv , abs/1608.08021v3*, 1–7.

Li, L.-J., Li, K., Li, F. F., Deng, J., Dong, W., Socher, R., Fei-Fei, L., 2009. ImageNet: A Large-Scale Hierarchical Image Database. *The IEEE Conference on Computer Vision and Pattern Recognition*, 248–255.

Lin, T. Y., Goyal, P., Girshick, R., He, K., Dollar, P., 2017. Focal Loss for Dense Object Detection. *The IEEE International Conference on Computer Vision*, 2999–3007.

Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D. C. L., 2014. Microsoft COCO: Common objects in context. *Lecture Notes in Computer Science*, 8693 LNCS(PART 5), 740–755.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., Berg, A. C. 2016. SSD: Single shot multibox detector. *Lecture Notes in Computer Scien*ce, 9905 LNCS, 21–37.

Li, Z., Peng, C., Yu, G., Zhang, X., Deng, Y., Sun, J. 2017. Light-Head R-CNN: In Defense of Two-Stage Object Detector. *CoRR*, abs/1711.0, 1–9.

Girshick, R. B. 2015. Fast R-CNN. *CoRR*, abs/1504.08083.

Girshick, R. B., Donahue, J., Darrell, T.,Malik, J., 2013. Rich feature hierarchies for accurate object detection and semantic segmentation. *CoRR*, abs/1311.2.

Redmon, J., Divvala, S. K., Girshick, R. B., Farhadi, A. 2015. You Only Look Once: Unified, Real-Time Object Detection. *CoRR*, abs/1506.0.

Redmon, J., Farhadi, A. 2018. YOLOv3: An Incremental Improvement. *CoRR*, abs/1804.02767.

Redmon, J., Farhadi, A. 2017. YOLO9000: Better, faster, stronger. *The 30th IEEE Conference on Computer Vision and Pattern Recognition*,6517–6525. doi.org/10.1109/CVPR.2017.690.

Ren, S., He, K., Girshick, R. B., Sun, J., 2015. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *CoRR*, abs/1506.0, 91-99.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Rabinovich, A., 2015. Going deeper with convolutions. *The IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1–9.

Yu, Z., Wu, X., Gu, X. 2017. Fully Convolutional Networks for Surface Defect Inspection in Industrial Environment, *Computer Vision Systems*, 417–426.

Zou, Z., Shi, Z., Guo, Y., Ye, J. 2019. Object Detection in 20 Years: A Survey. *CoRR*, abs/1905.0, 1–39.