DFSS: Distil FCOS for Security Scanning

Ziyang Xie, Yikun Wang, Shaowen Wang
{ziyangxie19, yikunwang19, wangsw19}@fudan.edu.cn
Fudan University

Abstract

We propose DFSS, a distilled FCOS for Security Scanning, which performs well in heavy computing scenes like crowded airports. We use YOLOX and our custom loss to teach the FCOS model and utilize the class weighted loss to make the model pay more attention to potentially dangerous objects. In this report, we find out that the distillation can greatly improve the performance in this finetune scene, and class weighted loss also can improve the performance of a specific class. Our approach allows security departments to deploy our model to devices with low computing power. We also release our code at https://github.com/Outsider565/DFSS

1. Introduction

Security scanning is becoming increasingly important in modern life. For example, airports and underground use X-rays as security scanning methods to detect potentially dangerous objects. Normally, specially trained security staff is needed to determine whether the dangerous object exists, which object it is, and where, which is pretty costly.

With the development of computer vision and the emergence of the X-ray dataset, automatic detection becomes possible using the object detection algorithm. Recently proposed algorithms like DETR have strong performance, but they need high computational power that X-ray detection devices normally don't have. Moreover, the general object detection algorithm treats each class equally. But in our case, dangerous objects should be considered more seriously.

We propose the DFSS: a distilled FCOS for security scanning to tackle these two issues. Firstly, We use the FCOS as the base model for its lightweights. Then we train the YOLOX as the teacher model and use our custom-designed distillation loss [4] to teach the child FCOS model. This operation improves the performance without requiring too much computational resources. Secondly, we add the supercategories normal and dangerous, which divides the original dataset into two parts. And we assign different weights to each class in our classification loss so that our model pays more attention to the dangerous objects.

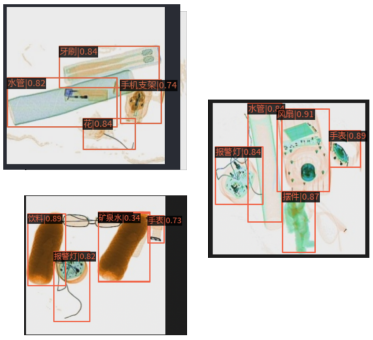

Figure 1. Instances of DFSS detection

1.1. Distillation

When we train from scratch, the model learns from the discrete ground truth label, which is hard for small model to generize. If we train a bigger model on the same data, the model can fit the distribution of the dataset and retrieve the internal information better. The key to distillation is how to pass the information from teacher model to student model. In our approach, we use the FCOS [7] as the student model and YOLOX [2] as the teacher model.

A common approach is to use a model of the same structure and then perform knowledge distillation on each layer of the model separately. However, in our case, fcos is the only one in its class. So we need to do the heterogeneous distillation. Because we uses the YOLOX [7] as the teacher model, we also inherite the crossentropy loss as the classification loss. In order to do the heterogeneous distillation, we make the student model also learn from the output logits from the teacher model using the KL loss. The idea behind this is logits contains richer information than the label itself. For instance, if the teacher model predicts one object with score [0.6, 0.4], the student model could infer that this object is similar to both class. Through this method, we achieve up to 6.5 % of performance improvement.

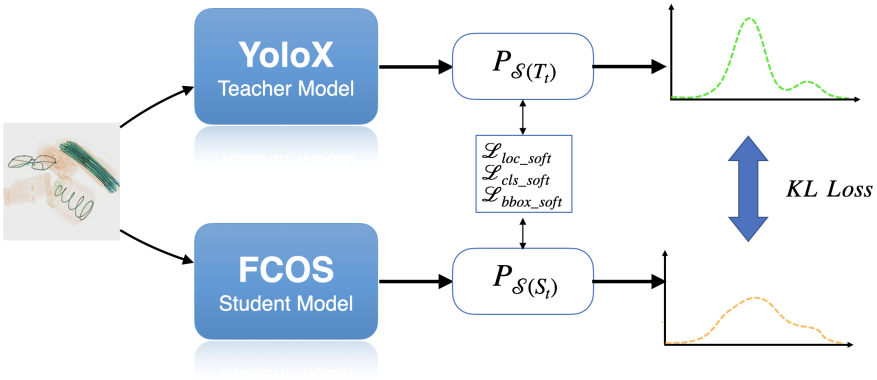

Figure 2. DFSS Model Architecture. $P_{\mathcal{S}(S_t)}$ and $P_{\mathcal{S}(T_t)}$ represent the distribution of the output logits from the student and teacher model. KL Loss is operated on all of the three dense head losses

1.2. Class Weighted Loss

As the DFSS is designed to do the security scanning, the precision and recall of detecting dangerous object is more important than general mAP. So we modify our dataset and the optimization goal, i.e. the loss. On the one hand, we convert the standard dataset to COCO [6] form, assign class labels to each image along with the (x,y,w,h) coordinates. Then we assign the supercategory of each class by hand. Every class can either be normal or dangerous. As a result, we assigned 8 dangerous classes, 75 normal classes and 1 unknown class.

On the other hand, apart from traditional crossentropy loss, out loss assigns different weight on different class based on its supercategory. If the ground truth label is in the dangerous supercategory, the loss of the item should be increased. In this way, our model pays more attention to the dangerous objects.

2. DFSS

2.1. Knowledge Distillation [3]

We proposed distill FCOS for security scanning. we choose YoloX as the teacher model with Kullback-Leibler divergence loss to instruct the student FCOS model.

YoloX teacher model

First, we trained yolox on our X-Ray dataset. we adopts the yolox_s model as our baseline, with CSPDarknet Backbone and YOLOX Feature Pyramid Network. [2] The classes number of dense head is modified to 84 alining with or dataset.

Distillation Loss

In this subsection, we introduce the proposed distillation method. Instead of general knowledge distillation, which distills the hybrid knowledge from the feature map of teacher model. we proposed a distillation strategy that distills the category and localization knowledge from the dense head loss. The knowledge distillation is operated on the output logits of the teacher model rather than the inner feature which allows us to distill the knowledge from a heterogeneous model but still achieves a decent performance.

The distillation loss is composed of three parts, the centerness(objectness) loss L_{loc_soft}, the category loss L_{cls_soft} and the loss of bounding box L_{bbox_soft}. These losses are added on the logits between the output logits from the student and teacher model. The KL distillation loss can guide the student model to mimic the logits distribution of the teacher model and converge.

$$\mathcal{L}_{t_{soft}} = \mathcal{L}_{KL}(\hat{P}_{S_t}, P_{T_t}), \quad t \in loc,\ cls,\ bbox$$

$\hat{P}_{S_t}$ and , P_{T_t} represent the distribution of the output logits from the student model and the teacher model. $\mathcal{L}_{KL}$ represents the Kullback-Leibler divergence loss.

The total loss for training the student S can be represented as:

$$\mathcal{L}_{W_s} = \lambda_0 L_{loc\_soft} + \lambda_1 L_{cls\_soft} + \lambda_2 L_{bbox\_soft}$$

We simple set the $\lambda_0 = \lambda_1 = \lambda_2 = \frac{1}{3}$, which is the same weight factor for each loss.

2.2. Class Weighted Loss

Then we introduce the class weighted loss. As the name suggests, we directly assign the weight so that the loss of dangerous objects is doubled. Original crossentropy loss has the following form:

$$\mathcal{L}_{CE} = -\frac{1}{N}\sum_{i}\sum_{c=1}^{M} y_{ic} log(p_{ic})$$

And after adding the weight:

$$\mathcal{L}_{CW} = -\frac{1}{N}\sum_{i}\sum_{c=1}^{M}\alpha_c y_{ic} log(p_{ic})$$

We assign $\alpha_c = 2$ when class C is dangerous, otherwise we assign $\alpha_c = 1$

2.3. Unknown Label Threshold

We noticed there are unknown labeled samples in the test set, but not in the training set. To deal with this issue, we present unknown lable threshold in the final NMS procedure. We introduce a score threshold $\mathcal{T}$, if the max category score of the sample is lower than the score, $Max(S_1,\ S_2, \ldots, S_N) < \mathcal{T}$, which means the model has lower confidence about which category this sample belongs to and we tend to classify this sample into unknown.

3. Experiments

3.1. Pretrain

For faster convergence, we use pretrained [5] models including teacher model (YOLOX-s) and student model (FCOS). The pretrained paramenters could accessed from internet, they are paramenters of detection models pretrained on famous detection dataset COCO. In our work, open mmlab https://openmmlab.com/ is main platform for coding and main source for pretrain parameters.

3.2. Fine tune

Era eyewitnessed Pretrain + Fine tune has become a popular AI application paradigm in post-ImageNet [1] era. In such a paradigm, models are often designed and trained for large dataset like ImageNet [1] (classification dataset) and COCO (detection dataset), and then trained on small size dataset for specified purposes such as detect a specified object like turtles. The second step often takes obviously less time.

We train models for x-ray detection by fine tuning them on the assigned x-ray dataset (24001 figures in train split and 16001 figures in test split). For the teacher model, (a.k.a YOLOX-small [2]), we fine tune with 100 epochs for more decent performance; considering the effectiveness of distillation, and for student model, we fine tune with only 12 opochs. For both teacher model and student model, we initialize learning rate with 0.01.

3.3. Classwise evaluation

We compare our result on dangerous and normal objects when using the class weighted loss and when not using the class weighted loss. We found out that while the performance of normal objects has dropped slightly, the performance of detecting dangerous objects increases notably.

Loss	Overall	Dangerous	Normal
CE loss	0.589	0.485	0.600
weighted loss	0.594	0.582	0.595
		+0.097	-0.005

Table 1. Compare between different loss. Results are shown in mmAP. Overall means the mAP of all classes. Dangerous means the mmAP50 of classes whose supercategory is dangerous. Normal means the mmAP50 of classes whose supercategory is normal. Weighted loss is trained based on the result of distilation checkpoint.

We also discover that our method isn't fully optimized for the unknown class, which only has 0.03 mAP. Further study can be conducted to improve the zero shot performance.

3.4. Result

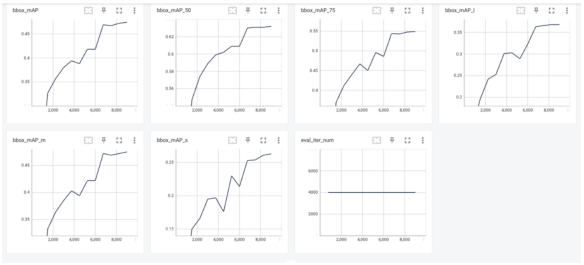

Figure 3. The mAP of DFSS during distillation process

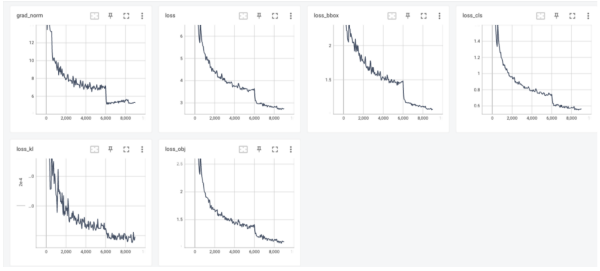

Figure 4. The loss descent of DFSS

Even considering less load in fine tune compared to pretrained on COCO dataset, fine tuning with 100 as max epoch number is still computing-demanding, eventually we fine tuned YOLOX-small and FCOS [7] on eight 2080-ti GPUS Linux platform.

During fine tuning, DFSS shows better data effectiveness compared to its counterpart (FCOS without distillation), within 12 epoches DFSS converged.

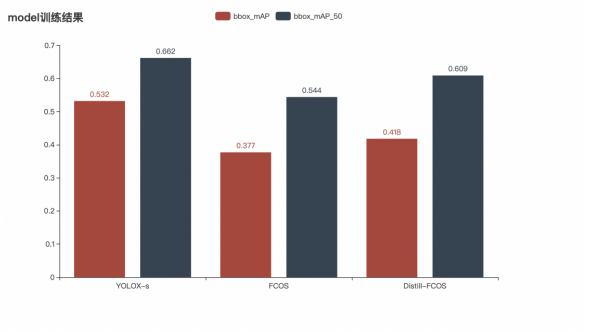

Figure 5. Comparation of teacher model (YOLOX-small), fine tuned student model (FCOS) without distillation, fine tuned student model (DFSS) with distillation

As the above histogram shows, teacher model performs best among the three models (0.662 mAP50, 0.532 mmAP), DFSS (0.609 mAP50, 0.418 mmAP) outperforms its counterpart (0.544 mAP50, 0.377 mmAP) without distillation. DFSS overall shows faster convergence during training, and perform better accuracy compared to undistil-FCOS (FCOS without distillation).

4. Conclusion

This work describes a light but powerful distillation model for x-ray detection. On the assigned x-ray dataset, DFSS reach mAP_50 of 0.609, and mmAP of 0.418. We show the effectiveness of distillation in the field of detection, using distillation FCOS well perform on par with large detection model like YOLOX, and outperforms fine-tuned FCOS, such a light-weighted model work better in computing demanding scenes like busy airport security check. Also, we provide costumed characteristics for security demand. With a highly man-designed loss, dangerous objects could be detected with now higher probability. Through such a work of distillation fully show its potential in the security scanning field. Future work should focus on further performance improvement by using a larger teacher model and how to better detect the unknown objects.

References

[1] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 3

[2] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. Yolox: Exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*, 2021. 1, 2, 3

[3] Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129(6):1789–1819, 2021. 2

[4] Jianyuan Guo, Kai Han, Yunhe Wang, Han Wu, Xinghao Chen, Chunjing Xu, and Chang Xu. Distilling object detectors via decoupled features, 2021. 1

[5] Geoffrey E Hinton and Russ R Salakhutdinov. A better way to pretrain deep boltzmann machines. *Advances in Neural Information Processing Systems*, 25:2447–2455, 2012. 3

[6] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2014. cite arxiv:1405.0312Comment: 1) updated annotation pipeline description and figures; 2) added new section describing datasets splits; 3) updated author list. 2

[7] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9627–9636, 2019. 1, 3