Imitating What You Need: An Adaptive Framework for Detector Distillation

Ruoyu Sun and Hongkai Xiong(✉)

Shanghai Jiao Tong University, 800 Dongchuan Road, Shanghai, China
{sunruoyu1,xionghongkai}@sjtu.edu.cn

Abstract. Object detection models with favorable performances usually suffer from high computational costs. Knowledge distillation, a simple model compression method, aims at training a light-weight student network by transferring knowledge from a cumbersome teacher model. In this paper, we investigate different components of typical two-stage and single-stage detector in details, and propose a detector distillation framework that adaptively transfers knowledge from teacher to student according to task specific priors. The knowledge is transferred adaptively at three levels, *i.e.*, feature backbone, classification head, and bounding box regression head, according to which model performs more reasonably. Furthermore, considering that it would introduce optimization dilemma when minimizing distillation loss and detection loss simultaneously, we propose a distillation decay strategy to help improve model generalization via gradually reducing the distillation penalty. Experiments on widely used detection benchmarks demonstrate the effectiveness of our method. Particularly, taking Faster R-CNN as an example, we achieve an accuracy of 39.4% with Resnet-50 on MS COCO 2017 dataset, which surpasses its baseline 37.5% by 1.9% points, and even better than the teacher model with 39.3% mAP.

Keywords: Object detection · Knowledge distillation · Gaussian masking · Adaptive regularization

1 Introduction

Object detection is a fundamental and challenging problem in computer vision. Typical detection models, varying from single-stage [17,19,23] to two-stage [5,6,20], achieves significant improvement in performances. However, these detectors are usually equipped with cumbersome models and suffer expensive computation cost. Hence, designing light-weight neural networks with high performance has attracted much attention in real-world applications.

Knowledge Distillation (KD), introduced by Hinton [11], has received much attention due to its simplicity and efficiency. The distilled knowledge is defined as soft label outputs from a large teacher model, which contain structural information among different classes. Following KD, many methods are proposed to either utilize softmax outputs [4,16] or mimic the feature layer of teacher models

Y. Peng et al. (Eds.): ICIG 2021, LNCS 12888, pp. 89–101, 2021.
https://doi.org/10.1007/978-3-030-87355-4_8

[21,27,28]. However, detection requires reliable localization in addition to classification, while the interleaved relationships among various modules in detector make it difficult to transfer knowledge directly.

To address the above issues, this paper proposes an adaptive distillation framework for typical object detectors, of which we deliberately design specific distillation strategy for each module according to their intrinsic properties. The highlight is that what we want to borrow from the teacher model is its generalization ability. In particular, our method adaptively mimics responses of teacher model in three aspects: 1) At feature backbone level, we highlight foreground regions by Gaussian masking operation for feature distillation. 2) At classification level, benefiting from a region proposal sharing mechanism, teacher model outputs soft labels within regions that student provides. 3) At bounding box regression level, regressed bounding box locations from teacher model's regression head are used as extra regressed targets for student model.

In addition, minimizing distillation loss and original training loss simultaneously would introduce optimization dilemma, where the optimal state suitable for distillation may not be acceptable for detection. To solve this issue, we further propose a distillation decay strategy to improve student's generalization via gradually reducing distillation penalty. In this way, the distillation term can be treated as regularization to help student model converge to a better optimization point.

To sum up, this paper makes following contributions:

- We propose an adaptive distillation framework for object detection, which transfers knowledge from teacher to student according to task specific priors. This is achieved by imitating knowledge at three levels, *i.e.*, feature backbone, classification head, and bounding box regression head, according to which model performs more reasonably.
- We propose a distillation decay strategy to gradually reduce teacher's interference to student, and help improve model generalization.

2 Related Works

2.1 Object Detection

Current CNN-based object detectors mainly include single-stage, two-stage and anchor-free detectors, where the former two are well-developed. Single-stage object detectors such as YOLO [19] directly perform object classification and bounding box regression on the feature maps. RetinaNet [14] proposes focal loss to mitigate the unbalanced positive and negative samples. Two-stage detectors [2,5,6,9,20] treat detection as a coarse-to-fine process via firstly generating candidate regions of interests and followed by a region refinement procedure. In particular, Faster R-CNN [20] introduces Region Proposal Network (RPN) to produce region proposals.

2.2 Knowledge Distillation

In order to accelerate network training, various model compression strategies have been proposed, such as weight quantization [7,26,29], network pruning [8,18,24], and low-rank factorization [12,22]. However, these methods either change network structure or contains large complexity, even hurting performance significantly.

Knowledge distillation is proposed to transfer knowledge from a high performance teacher model to a compact student model, aiming at improving latter's performance. It is first proposed by Hinton *et al.* [11] on image classification models, utilizing teacher's class probability vectors as soft labels to guide student's training. Hint learning [21] distills a deeper and thinner student model by imitating both soft outputs and intermediate feature representations of teacher model. Similar works are presented in [4,16,28].

In detection domains, knowledge distillation also shows its potential. Wang *et al.* [25] only distill backbone features within local regions near object. Chen *et al.* [1] distill two-stage detectors on all components including task heads. Li *et al.* [13] transfer knowledge from both positive and negative proposals on high-level features and corresponding task heads. However, these methods either have accuracy limited by redundant selected background features or suffer from drowning background proposals. Misguidance may also be provided when confronting proposals teacher model performs poorly on.

In summary, current distillation frameworks either lack specific design for detectors or fail to effectively select the most informative parts for distillation. Different from these works, our adaptive distillation method designs specific distillation strategy for each module according to their intrinsic properties and utilizes distillation decay strategy to further improve generalization.

3 Method

3.1 Network Overview

In this section, we describe our proposed adaptive distillation framework in detail. Without losing generality, we take the typical two-stage object detector Faster R-CNN [20] for instance, and the whole framework is shown in Fig. 1. Our proposed distillation method effects mainly on three parts: distillation of feature backbone, classification head, and bounding box regression head, respectively. We adaptively transfer knowledge from teacher to student model with different imitation strategies.

Based on responses of each components: 1) For backbone features, the foreground regions are modeled by a two-dimensional Gaussian mask inside ground truth bounding boxes to enhance objects while suppressing backgrounds. 2) For classification head, by region proposal sharing, teacher model utilizes student's positive samples to outputs soft labels and student's classification head is supervised by both one-hot labels and soft ones. 3) For bounding box regression head, a selective distillation scheme takes regressed bounding box locations from teacher

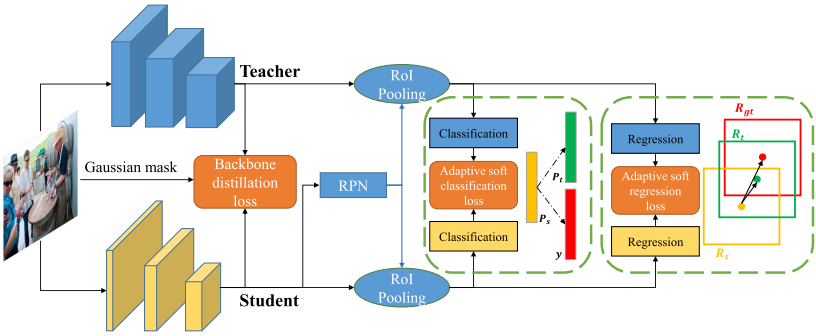

Fig. 1. Overview of the proposed task adaptive distillation framework. The proposed distillation method consists of three modules. Blue parts remain unchanged during distillation, while yellow parts are adaptively supervised by the teacher (Color figure online).

model as extra regressed targets to guide student model. As such, regression targets are progressively approaching to ground truths, which is more robust for bounding box regression. Each component would be elaborated as follows.

3.2 Backbone Features

For CNN-based detectors, the performance gain mainly credits to better backbone features. However, previous work [25] has found that directly imitating features at all locations would not promote the performance much. The reason is that it is the foreground regions that counts for a successful detector. Since drowning background regions usually occupy the majority area of a feature map, merely perform distillation at all the locations would inevitably introduce large amount of noise from those uncared negative regions. To address this issue, we introduce a Gaussian mask to highlight the centric foreground pixels and suppress those boundary regions around the objects.

Specifically, given a bounding box B of an object, with size of $w \times h$ and centered at (x_0, y_0), the two-dimensional Gaussian mask is defined as:

$$M = \begin{cases} e^{-\frac{(x-x_0)^2}{\sigma_x^2(w/2)^2} - \frac{(y-y_0)^2}{\sigma_y^2(h/2)^2}}, & (x,y) \in B \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

where σ_x^2 and σ_y^2 are decay factors along the two directions, and we set $\sigma_x^2 = \sigma_y^2$ for simplicity. The mask is only effective within ground truth bounding boxes and equals to zero everywhere else. In particular, when several Gaussian masks overlap over a single pixel (x, y), the mask value is simply set as point-wise maximum. Figure 2 illustrates two images, as well as their corresponding Gaussian masks.

With the delicately designed Gaussian mask, the backbone features are distilled via minimizing the following loss:

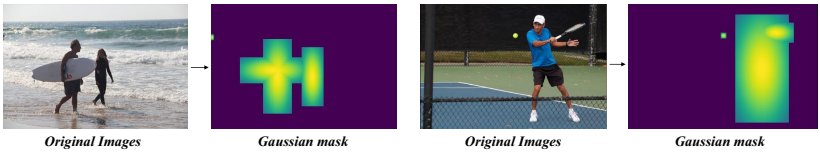

Fig. 2. An illustration of the generated Gaussian masks over samples from MS COCO [15] dataset.

$$L_{bk} = \frac{1}{2N_a} \sum_{i=1}^{W} \sum_{j=1}^{H} \sum_{c=1}^{C} M^{ijc} (F_s^{ijc} - F_t^{ijc})^2, \tag{2}$$

where $N_a = \sum_{i=1}^{W} \sum_{j=1}^{H} \sum_{c=1}^{C} M^{ijc}$, W, H, C are the width, height, and channels of the feature map, F_s^{ijc} and F_t^{ijc} denote the backbone features of student and teacher models, respectively.

3.3 Classification Head

Knowledge distillation is widely used in classification tasks [11,21,28], where soft labels provide structural information among different categories. However, directly transferring soft labels in a detection system is not applicable: First, background regions occupy the majority of proposals, which introduce a large amount of noise to their structural information; Second, proposals output by two models are inevitably different, which makes them not comparable for knowledge distillation.

Our method effectively address the above issues: 1) We only focus on positive samples which are beneficial when modeling inter-class structural priors and neglect negative ones. 2) We share student's RPN proposals to teacher model, and use its corresponding outputs as soft labels. Specifically, given N region proposals from student RPN's output, we compute soft labels of N_p positive samples over teacher model $\{y_t^i\}_{i=1}^{N_p} \in \mathbb{R}^{C'}$, where C' denotes the number of classes. Accompanying with all the N proposals and their ground truth labels $\{y^j\}_{j=1}^{N} \in \mathbb{R}^{C'}$, total loss for classification head is reformulated as

$$L_{cls} = \sum_{j=1}^{N} L_{CE}(y_s^j, y^j) + \beta_1 \sum_{i=1}^{N_p} L_{BCE}(y_s^i, y_t^i), \tag{3}$$

where L_{CE} and L_{BCE} denote cross-entropy and binary cross-entropy, respectively. y_s is prediction of student model, and β_1 is a balancing factor that controls the two loss terms.

3.4 Bounding Box Regression Head

Regression head in detectors adjusts locations and sizes of candidate region proposals. For bounding box regression distillation, we expect teacher model's

output to offer a reasonable mild target for student model to regress, relieving the forced abrupt regression targets from current proposal to ground truth.

However, teacher's regression output may provide wrong guidance for student model and even contradicts to ground truth's direction. Therefore, we propose a distillation strategy that selectively relies on teacher's outputs. Similar to classification head distillation, student's positive proposals are shared to teacher model. Specifically, given N_p positive region proposals from student's RPN output, we denote $\{r_p^i\}_{i=1}^{N_p}$, $\{r_t^i\}_{i=1}^{N_p}$, $\{r_s^i\}_{i=1}^{N_p}$, $\{r_{gt}^i\}_{i=1}^{N_p}$ as proposal locations before regression, teacher's regression output, student's regression output, and corresponding ground truth, respectively. We first calculate IoU (Intersection-Over-Union) between r_p^i and r_{gt}^i, and IoU between r_t^i and r_{gt}^i. If $IoU(r_t^i, r_{gt}^i) > IoU(r_p^i, r_{gt}^i)$, it indicates that teacher's regression output is a reliable indicator to provide correct guidance for student. Otherwise, this proposal is abandoned in distillation. The final regression loss is formulated as follows:

$$L_{reg} = \sum_{j=1}^{N} L(r_s^j, r_{gt}^j) + \beta_2 \sum_{i=1}^{N_p} L_{dist}(r_s^i, r_t^i, r_{gt}^i), \tag{4}$$

where L is $L1$ loss defined as $L1$ distance between two vectors, β_2 is a balance factor, and L_{dist} is the selective distillation loss:

$$L_{dist}(r_s^i, r_t^i, r_{gt}^i) = \begin{cases} L(r_s^i, r_t^i) & \text{if } IoU(r_t^i, r_{gt}^i) > IoU(r_p^i, r_{gt}^i) \\ 0. & \text{otherwise} \end{cases} \tag{5}$$

Integrating the above three distillation terms produces our overall training targets of the student model, which can be formulated as:

$$L = \lambda L_{bk} + L_{cls} + L_{reg} + L_{rpn}, \tag{6}$$

where λ is the balance parameter for backbone distillation and L_{rpn} is the RPN training loss in two-stage detector as described in [20].

3.5 Adaptive Distillation Decay

The overall loss function in Eq. (6) simultaneously minimizes distillation loss and detection loss. It formulates a multi-task learning issue and makes training process hard to converge. To solve this issue, we propose a distillation decay strategy to help improve model generalization via gradually reducing distillation penalty, hoping that the model focuses more on detection task in the training process. This is achieved by introducing a time decay variable $\gamma(t)$, which decreases to 0 as the training proceeds. In our implementation, we simply set $\gamma(t) = 1 - t/T$ at the tth training iteration, where T is the total training iterations. The time decay variable is imposed to balance parameters β_1, β_2, and λ in Eq. (3), Eq. (4), Eq. (6) to control the intensity of distillation loss, *i.e.,*

$$\tilde{\beta_1} = \gamma(t)\beta_1, \quad \tilde{\beta_2} = \gamma(t)\beta_2, \quad \tilde{\lambda} = \gamma(t)\lambda. \tag{7}$$

4 Experiments

In this section, we evaluate our adaptive distillation framework for object detection, providing extensive designed evaluation and making comparison with previous works.

4.1 Experimental Setup

Datasets and Evaluation Metrics. We evaluate our approach on two widely used detection datasets: 1) PASCAL VOC 2007 [3], containing totally 9,963 images of 20 object classes, of which 5,011 images are included in *trainval* and the rest 4,952 in *test*; 2) Microsoft COCO 2017 [15], a large scale dataset that contains over 135k images spanning 80 categories, of which over 120k images are used for *train* and around 5k for *val*. Following the default settings, for PASCAL VOC, we choose the *trainval* split for training and the *test* split for test, while for MS COCO, we choose the *train* split for training and the *val* split for test. For performance evaluation, the average precision(AP) is used. We report the COCO style (AP [0.5:0.95]) detection accuracy for MS COCO, and PASCAL style (AP [0.5]) accuracy for PASCAL VOC.

Baseline Models. We evaluate our method based on both two-stage and single-stage detection frameworks. For two-stage detectors, we choose widely used Faster R-CNN [20] detector and for single-stage detectors, RetinaNet [14] is selected. Since there are no RPN layers for RetinaNet and anchors generated by teacher and student are exactly the same, we directly utilize positive anchors for task heads' distillation. Other operations and parameters are the same with those in two-stage detectors. Resnet [10] series networks are used as backbones of detectors, depending on their model sizes. For ease of narration, if Resnet-101 is used as teacher model and Resnet-50 as student, the distilled model is simply denoted as R-101-50.

Implementation Details. All experiments are performed on NVIDIA Tesla V100 8 GPUs with parallel acceleration. With Stochastic Gradient Descend (SGD) as optimization method, we set batch size to 16, allocating 2 images per GPU. The Gaussian parameters σ_x^2 and σ_y^2 in Eq. (1) are set to 2. The balance factors β_1, β_2, and λ are set to 10, 3, 0.6, respectively, via diagnosing the initial loss of each branch and ensuring that all losses are within the same scale. We find that these parameters are robust in our method, and do not affect the results too much as long as they are in similar scale. Unless specified, all experiments choose 1x schedule for training 12 epochs. The resolutions for images in COCO and PASCAL VOC are set as (1333,800) and (1000,600), respectively, following traditional implementation of each dataset.

Table 1. Effects of various distillation modules on PASCAL VOC 2007.

Student R-50	✓						
Teacher R-101							✓
Backbone?		✓			✓	✓	
Classification Head?			✓		✓	✓	
Regression Head?				✓	✓	✓	
Distillation Decay?						✓	
mAP (%)	70.0	72.8	73.2	73.4	73.8	**74.5**	74.3

Table 2. Hyperparameter analysis of Gaussian mask's variances on PASCAL VOC 2007. 'Rectangle' denotes using rectangle mask while 'All features' denotes distilling the whole feature map.

$\sigma_x^2 = \sigma_y^2$	1	2	4	Rectangle	All features
mAP	72.7	**72.8**	72.7	72.4	72.1

4.2 Ablation Study

Component Analysis. We first conduct experiments to understand how each distillation module contributes to the final performance, as well as the robustness of our method to different parameters. Without loss of generality, all experiments in this section are based on PASCAL VOC 2007 with Resnet-101 as teacher and Resnet-50 as student, which produces accuracy of 74.3% and 70.0%, respectively.

As shown in Table 1, different distillation components includes 1) backbone with Gaussian masking, 2) classification head, 3) regression head, 4) distillation decay. From the table we make following observations:

- *Backbone Distillation:* The backbone distillation brings about 2.8% mAP gain. The results demonstrates that our masking strategy enables student model to learn highlighted foreground information.
- *Classification and Regression Head Distillation:* The independent distillation strategies on classification head and regression head improve the student model by 3.2% and 3.4% points, respectively. It indicates that classification head distillation can provide effective soft labels, while regression head offers correct guidance.
- *Combination:* Combination of the above three distillation targets achieves better results, which brings about another 0.4% points gain compared with the best single distillation module. The combination strategy obtains marginal improvement compared with the individual components, partially due to the difficulty of joint optimization.
- *Distillation Decay:* The distillation decay strategy can further improve the results from 73.8% to 74.5%. This demonstrates effectiveness of the proposed distillation decay strategy. In this way, teacher model can be treated as a guider that leads student to a better optimization point gradually, which improves its generalization ability.

Table 3. Per category evaluation results on PASCAL VOC 2007.

Network	mAP	aero	bike	bird	boat	bott.	bus	car	cat	chai.	cow	tabl.	dog	hors.	mbike	pern	plnt	sheep	sofa	train	tv
R-101	74.3	73.3	83.6	78.2	60.2	61.5	75.5	84.6	85.7	58.6	80.1	61.1	85.7	82.5	80.0	82.9	50.2	74.7	72.0	81.1	75.2
R-50	70.0	67.1	79.1	73.9	56.3	54.4	74.9	81.5	82.7	51.7	76.4	53.2	82.3	81.4	75.7	80.5	45.0	71.9	64.0	77.7	70.9
R-101-50	74.5	74.0	82.4	75.7	60.9	62.0	79.9	84.7	86.1	58.2	80.3	64.0	84.7	83.7	81.0	83.2	51.2	77.9	70.2	78.3	72.1

Table 4. Evaluation results for different teacher and student models for Faster R-CNN on MS COCO 2017.

Network	Model info	mAP	AP50	AP75	APs	APm	APl
R-152	76.5M/10.8 fps	41.3	61.9	45.1	24.2	45.8	53.3
R-101	60.9M/11.9 fps	39.3	60.0	42.7	22.8	43.7	51.0
R-152-101		**41.8**	**61.8**	**45.5**	**23.6**	**46.1**	**54.5**
R-50	41.8M/13.6 fps	37.5	58.3	40.8	21.8	40.9	48.5
R-101-50		39.4	59.8	43.0	22.5	43.5	52.0
R-152-50		40.2	60.4	43.9	23.4	44.3	53.1
R-152-101-50		**40.6**	**60.9**	**44.2**	**23.4**	**44.5**	**53.2**

Table 5. Distillation results of RetinaNet on MS COCO 2017, together with the model size and inference speed.

Network	Model info	mAP	AP50	AP75	APs	APm	APl
Retina-101	57.1M/10.9 fps	38.6	57.6	41.3	22.2	42.5	51.0
Retina-50	38.0M/12.1 fps	36.4	55.5	38.6	20.3	40.0	47.9
Retina-101-50		**38.6**	**57.8**	**41.2**	**21.9**	**42.1**	**50.9**

Hyperparameter of Gaussian Mask. We now investigate the influence of Gaussian mask for detection performance. For simplicity, we fixed $\sigma_x^2 = \sigma_y^2 = k$ and jointly change the two parameters. In principle, with larger k, Gaussian mask becomes more scattered, while with smaller k, the mask will concentrate more on center of the box. As an extreme condition, when $\sigma_x^2 = \sigma_y^2 = +\infty$, Gaussian mask degrades to a rectangle mask. To verify the effectiveness of Gaussian mask, we take experiments on rectangle mask and the whole feature map without any masking operation. The results are shown in Table 2, where we find that the performance is relatively robust to Gaussian mask and it is much better than simply using rectangle mask or the whole feature maps for distillation.

4.3 Experimental Results

PASCAL VOC. The detection results by category on PASCAL VOC are shown in Table 3. Our distillation model R-101-50 achieves a significant boost for each category, and brings 4.5% overall gain (from 70.0% to 74.5%) compared to student model, which demonstrates superior performance of our proposed distillation framework.

Table 6. Comparison with previous distillation methods for detectors on VOC 2007.

	Network	Mimic [13]	FGFI [25]	Ours
Teacher	R-101	74.3	74.4	74.3
Student	R-50	70.0	69.1	70.0
Distilled	R-101-50	72.7	72.0	**74.5**
Improvement	–	+2.7	+2.9	**+4.5**

MS COCO. Table 4 and Table 5 show overall distillation performances on COCO dataset for Faster R-CNN [20] and RetinaNet [14], respectively. As shown, R-101-50 exceeds its teacher R-101 and R-152-101 also surpasses R-152. Specifically, R-152-50 (40.2%) even exceeds R-101 (39.3%) by a large margin. The reason for the gap between R-152-50 and its teacher R-152 is the large differences in structures and features between them. The progressive distillation approach R-152-101-50 brings R-50 closer to R-152. In detail, R-101 is firstly utilized as teacher model to distill R-50, obtaining R-101-50. Then we use R-152 as teacher model to further distill R-101-50, resulting R-152-101-50 which achieves a further mAP gain compared to R-152-50. It is worth mentioning that although R-50 contains much less layers than those of R-152 (almost 1/3), our R-152-101-50 still has a comparable result to the R-152. Similarly, distillation on single-stage RetinaNet also obtains outstanding performance. Retina-101 improves Retina-50 by 2.2% mAP and shows equal performance with its teacher Retina-101. The experiment results demonstrate significant performance improvement brought by our distillation framework.

Compression and Acceleration. To better illustrate compression and acceleration effect of knowledge distillation in object detection, we provide parameters amount and inference speed of each model in Table 4 and Table 5. The two metrics are shown in 'model info' column, where 'M' and 'fps' denote 'million(s)' and 'frames per second', respectively. According to the results, our distillation framework effectively lightens network sizes and increases their inference speed. For Faster R-CNN, the distillation models (*e.g.*, R-101-50, R-152-50) are much faster, while offering comparable or better performance against their teacher models. For RetinaNet, Retina-101-50 compresses Retina-101 by about 68% with 1.2 fps gain, while its performance remains comparable to Retina-101.

4.4 Comparison with Previous Distillation Methods

To further explore effectiveness of the proposed distillation framework in object detection, we present comparative results between our method with previous works [13,25]. Since the devil is in experimental details, the results of teacher and student models may differ in our implementation. We simply re-implement results of [13] which distills all proposals, while for method in [25], we refer to results in the original paper. From Table 6, we make following observations:

Mimic [13] improves R-50 from 70.0% to 72.7% with 2.7% mAP gain, while FGFI [25] offers a little better result with 2.9% mAP gain. Apparently, our approach outperforms these distillation methods on absolute performance with a prominent mAP gain of 4.5%. The significant advantage of our method mainly comes from three aspects: 1) The Gaussian mask effectively suppresses the undesirable background noise while retaining informative foregrounds; 2) Adaptive distillation for task heads provides suitable guide for student model; 3) Distillation decay strategy helps model's optimization.

5 Conclusion

In this paper, we proposed an adaptive distillation framework for typical object detectors. The key contribution is that we deliberately design different imitating schemes according to the property of each distilled target. Based on the responses, we are able to successfully select crucial part of teacher's feature maps, classification structural priors, and bounding box regression results as supervision for distillation. Besides, a distillation decay strategy is deployed to help improve model generalization via gradually reducing the distillation penalty. Experiments conducted on widely used detection benchmarks demonstrate the effectiveness of the proposed method.

References

1. Chen, G., Choi, W., Yu, X., Han, T., Chandraker, M.: Learning efficient object detection models with knowledge distillation. In: Advances in Neural Information Processing Systems, pp. 742–751 (2017)
2. Dai, J., Li, Y., He, K., Sun, J.: R-FCN: object detection via region-based fully convolutional networks. In: Advances in Neural Information Processing Systems, pp. 379–387 (2016)
3. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (VOC) challenge. Int. J. Comput. Vis. **88**(2), 303–338 (2010)
4. Fukuda, T., Suzuki, M., Kurata, G., Thomas, S., Cui, J., Ramabhadran, B.: Efficient knowledge distillation from an ensemble of teachers. In: Interspeech, pp. 3697–3701 (2017)
5. Girshick, R.: Fast R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1440–1448 (2015)
6. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 580–587 (2014)
7. Han, S., Mao, H., Dally, W.J.: A deep neural network compression pipeline: pruning, quantization, Huffman encoding. arXiv preprint arXiv:1510.00149 (2015)
8. Han, S., Pool, J., Tran, J., Dally, W.: Learning both weights and connections for efficient neural network. In: Advances in Neural Information Processing Systems, pp. 1135–1143 (2015)
9. He, K., Zhang, X., Ren, S., Sun, J.: Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Trans. Pattern Anal. Mach. Intell. **37**(9), 1904–1916 (2015)

10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
11. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531 (2015)
12. Kim, Y.D., Park, E., Yoo, S., Choi, T., Yang, L., Shin, D.: Compression of deep convolutional neural networks for fast and low power mobile applications. arXiv preprint arXiv:1511.06530 (2015)
13. Li, Q., Jin, S., Yan, J.: Mimicking very efficient network for object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6356–6364 (2017)
14. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2980–2988 (2017)
15. Lin, T.Y., et al.: Microsoft COCO: common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8693, pp. 740–755. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10602-1_48
16. Liu, P., Liu, W., Ma, H., Mei, T., Seok, M.: KTAN: knowledge transfer adversarial network. In: Association for the Advance of Artificial Intelligence (2019)
17. Liu, W., et al.: SSD: single shot MultiBox detector. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9905, pp. 21–37. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_2
18. Park, J., et al.: Faster CNNs with direct sparse convolutions and guided pruning. arXiv preprint arXiv:1608.01409 (2016)
19. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: unified, real-time object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 779–788 (2016)
20. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. In: Advances in Neural Information Processing Systems, pp. 91–99 (2015)
21. Romero, A., Ballas, N., Kahou, S.E., Chassang, A., Gatta, C., Bengio, Y.: FitNets: hints for thin deep nets. In: International Conference on Learning Representations (2015)
22. Sainath, T.N., Kingsbury, B., Sindhwani, V., Arisoy, E., Ramabhadran, B.: Low-rank matrix factorization for deep neural network training with high-dimensional output targets. In: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 6655–6659. IEEE (2013)
23. Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., LeCun, Y.: OverFeat: integrated recognition, localization and detection using convolutional networks. arXiv preprint arXiv:1312.6229 (2013)
24. Tung, F., Mori, G.: Deep neural network compression by in-parallel pruning-quantization. IEEE Trans. Pattern Anal. Mach. Intell. **42**(3), 568–579 (2018)
25. Wang, T., Yuan, L., Zhang, X., Feng, J.: Distilling object detectors with fine-grained feature imitation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4933–4942 (2019)
26. Wu, J., Leng, C., Wang, Y., Hu, Q., Cheng, J.: Quantized convolutional neural networks for mobile devices. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4820–4828 (2016)
27. Yim, J., Joo, D., Bae, J., Kim, J.: A gift from knowledge distillation: fast optimization, network minimization and transfer learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4133–4141 (2017)

28. Zagoruyko, S., Komodakis, N.: Paying more attention to attention: improving the performance of convolutional neural networks via attention transfer. In International Conference on Learning Representations (2017)
29. Zhou, A., Yao, A., Guo, Y., Xu, L., Chen, Y.: Incremental network quantization: towards lossless CNNs with low-precision weights. arXiv preprint arXiv:1702.03044 (2017)