Learning Efficient Detector with Semi-supervised Adaptive Distillation

Shitao Tang[1]
shitaot@gmail.com
Litong Feng[2]
fenglitong@sensetime.com
Wenqi Shao[3]
weqish@link.cuhk.edu.hk
Zhanghui Kuang[2]
kuangzhanghui@sensetime.com
Wayne Zhang[2]
wayne.zhang@sensetime.com
Zheng Lu[1]
zheng.lu@nottingham.edu.cn

[1] The University of Nottingham Ningbo, China
[2] SenseTime Research Shenzhen, China
[3] The Chinese Univerisity of Hong Kong Hong Kong

Abstract

Convolutional Neural Networks based object detection techniques produce accurate results but often time consuming. Knowledge distillation has been popular for model compression to speed up. In this paper, we propose a Semi-supervised Adaptive Distillation (SAD) framework to accelerate single-stage detectors while still improving the overall accuracy. We introduce our Adaptive Distillation Loss (ADL) that enables student model to mimic teacher's logits adaptively with more attention paid on two types of hard samples, hard-to-learn samples predicted by teacher model with low certainty and hard-to-mimic samples with a large gap between the teacher's and the student's prediction. We then show that student model can be improved further in the semi-supervised setting with the help of ADL. Our experiments validate that for distillation on unlabeled data. ADL achieves better performance than existing data distillation using both soft and hard targets. On the COCO database, SAD makes a student detector with a backbone of ResNet-50 out-perform its teacher with a backbone of ResNet-101, while the student has half of the teacher's computation complexity.

1 Introduction

Boosted by the development of deep Convolutional Neural Networks (CNN), the accuracy of object detection has been improved greatly over the years [15, 17, 21]. Despite satisfying detection accuracy, CNN based object detection techniques suffer from long computational time making them unusable in time-demanding applications such as mobile apps, self-driving cars, etc. Various efforts have been focused on speeding up the process including detection pipeline optimization [15, 17, 21], architecture design [10, 28], pruning [7], quantization [29], decomposition [12, 25], and knowledge distillation [9].

Among these works, Knowledge Distillation (KD) shows its advantage in terms of model acceleration by making use of two networks, namely student and teacher, during the training to improve the overall accuracy while reducing computational time in testing (student network only). This is done by encouraging the student network to converge to a better solution through mimicking the teacher network's feature maps or soften logits. KD has achieved great success on image classification [9, 22, 27]. However, in the area of object detection, due to the "small" capacity of the student network, it is very hard to mimic all feature maps or logits directly. To solve this problem knowledge transfer has been applied in various object detectors. Chen *et al*. [2] proposed a weighted cross-entropy loss to underweight matching errors in background regions. Li *et al*. [13] mimicked feature maps between the student and the teacher pooled from the same region proposal and discarded those from uninterested regions. All these efforts attempt to focus on mimicking informative neurons of the teacher network that contains two stages, namely region proposal network and classification network, despite the obvious efficiency of single-stage detectors. In other words, how KD can speedup object detectors that has a single network (single-stage) is yet to be exploited.

Compared with two-stage detectors, single-stage detectors needs to use much more samples due to dense anchors. Without region proposal network, sample imbalance between easy and hard samples is very challenging for single-stage detectors. Directly applying KD to single-stage detectors leads to a large number of easy samples dominating the KD loss. As a result, the lack of guidance from hard samples hinders the performance of single-stage detectors even with KD. We can categorize all the important samples in the distillation process into two types: 1) hard-to-mimic samples whose gaps between the student's prediction and the teacher's prediction are large; 2) hard-to-learn samples whose uncertainties defined by teacher's prediction are large. Both hard-to-mimic and hard-to-learn samples are generated from teacher model and should be paid more attention for an effective distillation in single-stage detectors. Based on this observation, an adaptive distillation knowledge loss (ADL) is proposed, which pays more attention to teacher-defined hard samples and adaptively adjusts the distillation weights between easy-to-mimic/easy-to-learn and hard-to-mimic/hard-to-learn samples in the distillation process. Furthermore, existing techniques mainly focus on supervised approach that requires labeling object bounding box that is extremely time-consuming. This indirectly hinders the overall performance of such techniques due to limited labeled data. Other lines of work such as [18, 23] have demonstrated that unlabeled data can potentially help image classification and object detection. However, it is still unclear how to extract the knowledge of unlabeled data to guide the student network training with approaches like KD.

In this paper, we propose an Semi-supervise Adaptive Distillation (SAD) framework to learn an efficient object detector. Provided with potentially unlimited unlabeled data from the Internet, the teacher model in our framework can effectively guide the student model with significant detection accuracy improvement via the augmented transfer set. Instead of making use of unlabeled data with labels directly predicted by teacher model (hard targets) similar to [18], our framework makes use of both hard targets and soft targets provided by teacher model for better performance. This is because hard targets predicted by teacher model are often with very high confidence scores and hence can be easily classified in student model. In contrast, soft targets provided by ADL from teacher model often have a good balance of samples.

2 Related work

Semi-supervised learning and self training: Semi-supervised learning has been studied for years. [1, 11, 18, 23, 30] The goal is to make use of partially labeled data in training stage. In [23], experiments show that object detectors can gain extra improvement using semi-supervised instead of fully supervised training. Similarly, data distillation[18] improves the overall performance by first training a model with labeled data and then using the model to make predictions on unlabeled data through multi-transform inference and data transformations. While also using semi-supervise approach, our work focuses on knowledge transfer from strong teacher to weak student.

Object detection: Object detection techniques can be categorized into single-stage and two-stage approaches. Two-stage approaches usually consist of two parts, a region proposal network that generates a sparse set of candidate object proposals and a network to further refine and classify those proposals. [5, 6, 15, 21] Single-stage approaches directly forward raw pixel values through a Convolution Neural Network and obtain classification and location results. [4, 17, 20] Although working well, both approaches suffer from the problem of class imbalance. To address this problem, Abhinav *et al*. [24] introduce Online Hard Example Mining (OHEM) by selecting the top k samples according to loss values per mini-batch. In contrast to two-stage detectors whose region proposal network can reduce the candidate locations significantly, single-stage detectors suffer more severely from the problem of class imbalance. Different from OHEM, focal loss [16] aims to pay more attention to hard examples than easy examples by multiplying a focal term with common cross entropy loss. Our distillation loss design follows the same spirit of focal loss to solve the problem.

Deep network compression and acceleration: Among many works accelerating Convolution Neural Networks for practical applications, knowledge transfer aims to transfer the knowledge learned by teacher model to student model. Previous work exploit knowledge transfer by representing knowledge in various forms. For example, FitNet [22] makes student model mimic the full feature maps of teacher model. Knowledge Distillation (KD) [9] supervises student model by using soft targets predicted by teacher model. In this way, the probability distribution from teacher model provides extra information in contrast to using one-hot target encoding directly in conventional approaches. Naturally network model compression is applied to object detection as well. Chen *et al*. [2] utilize soft targets to guide student model in both region proposal network and region convolution network, and then balance positive and negative examples by re-weighting losses of positive and negative samples. Instead of directly addressing the problem of class imbalance, Li *et al*. [13] propose to match feature maps after ROI-pooling layer in which the number of candidate regions is significantly reduced. These methods are designed for two-stage detectors and cannot be easily applied to single-stage detectors directly. In this work, our approach is based on the same idea as KD for acceleration and introduces a uniquely designed loss to address the problem of class imbalance from a different perspective.

3 Semi-supervised Adaptive Distillation

The overall framework of our adaptive distillation for object detection in an semi-supervised learning setting is shown in Figure 1. During training, the labeled images are first used to train the teacher network. Then the labeled and unlabeled images are fed to both the teacher and student networks to use teacher model to guide student model with our adaptive

distillation loss with soft targets. If an image does not have ground truths, the predicted label by teacher model (hard target) is used as ground truth for focal loss computation. During testing, only the student model is used for acceleration with high detection accuracy thanks to adaptive distillation and semi-supervised setting.

3.1 Adaptive Distillation

Our adaptive distillation loss is uniquely designed for single-stage detectors. Compared to two-stage detectors, the distinguishing feature of single-stage detectors is their dense sampling of possible object locations. In a single-stage detector, dense anchors are set on multiple feature maps in the backbone network, which leads to computational inefficiency. For example, in KD, distillation loss needs to be calculated on a large number of output logits between teacher and student models. In RetinaNet [16], a typical number of anchors is $\sim$100K and most of those anchors correspond to easy-to-mimic or easy-to-learn samples. Although an easy sample contributes very little, the distillation loss are dominated by easy samples during training due to their sheer amounts. As a result, those hard-to-mimic or hard-to-learn samples are not learned well hence restricting the capacity of KD on a single-stage detector.

Without loss of generality, we study the case of cross entropy for binary classification to further demonstrate the problem. The original focal loss is defined as

$$\begin{aligned} FL(p_t) &= -(1-p_t)^{\gamma} log(p_t) \\ p_t &= \begin{cases} p, & \text{if y=1} \\ 1-p, & \text{otherwise} \end{cases} \end{aligned} \tag{1}$$

In above, $y \in \{\pm 1\}$ specified the ground-truth class and $p \in [0,1]$ is the model's estimated probability for the class with label $y = 1$.

In the following, We represent q as the soft probability value predicted by teacher model and p as the one predicted by student model. Kullback-Leibler divergence (KL) inspiring the original knowledge distillation, which measures the similarity between two distributions, is defined as:

$$KL(T||S) = q\log(\frac{q}{p}) + (1-q)\log(\frac{1-q}{1-p}) \tag{2}$$

Student model tries to mimic the soft class probability distribution predicted by teacher model by making use of this equation. We abbreviate $KL(T||S)$ as KL for rest of the paper.

Focal Distillation Loss: The common way of adopting focal loss to knowledge distillation is to multiply KL by a Focal Term (FT). If the focal term utilized by the classification loss (hard targets) is shared by KL (soft targets), the Joint Focal Loss of classifications and KL can be defined as

$$\begin{aligned} JFL &= FT(p)(-\log(p_t) + KL) \\ FT(p) &= (1-p_t)^{\gamma} \end{aligned} \tag{3}$$

where $FT(p)$ is the focal term. Thus, the focal distillation loss is:

$$FDL = (1-p_t)^{\gamma} KL \tag{4}$$

where FDL is a simple modification from FL and KL and used as a baseline in our experiments.

Adaptive Distillation Loss: However, as our experiments show, FDL is usually dominated by the focal term FT. KL contributes little to the total loss. In order to address the problem, we propose the adaptive distillation loss. Our idea is that that KD on a single-stage detector should focus on measuring the distance of the probability distributions between student and teacher models. We use a modulating factor between 0 to 1 learn the features adaptively. Inspired by KL-divergence, we use the following to accomplish the purpose:

$$DW = (1 - e^{-KL})^{\gamma} \tag{5}$$

where KL is defined in Equation (2). DW is abbreviated for distillation weight. Similar to the focal loss, hyper-parameter γ controls the rate at which easy examples are down-weighted. Term $(1 - e^{-KL})$ controls the weight of each sample. Note that DW only adjusts the weights between the student and teacher during the training process. Given that hard-to-learn samples are extremely important for distillation, we propose ADW to adjust the weights of hard-to-learn samples, defined as:

$$\begin{aligned} ADW &= (1 - e^{-(KL + \beta T(q))})^{\gamma} \\ T(q) &= -(q \log(q) + (1-q)\log(1-q)) \end{aligned} \tag{6}$$

T(q), the entropy of the teacher model, reaches maximum when q is 0.5 and minimum when q approaches 0 or 1. The teacher's probability q reflects the uncertainty of classification. When q approaches to 0.5, the corresponding sample is treated as a hard-to-learn sample. And a sample with a high KL is treated as a hard-to-mimic sample. Intuitively, the weights of the hard-to-learn samples increase when β becomes larger. Thus, KL controls the weights of hard-to-mimic samples that are adjustable in the training process, while $T(q)$ controls the weights of hard-to-learn samples initially defined by the teacher model. The combination of these adaptively adjusts the distillation weights. So our adaptive distillation loss is defined as:

$$ADL = ADW \cdot KL \tag{7}$$

And for student model, we optimize the following function:

$$L = FL + ADL + L_{loc} \tag{8}$$

FL is the original focal loss and L_{loc} is the bounding box loss. ADL is the proposed adaptive distillation loss.

3.2 Semi-supervised Adaptive Distillation Framework

Labeling samples is labor-intensive, especially for the task of object detection. It is natural to try to make use of unlabeled samples vastly available on the Internet. In our work, we make use of both labeled and unlabeled samples in a semi-supervised way. Previous work [18] introduces semi-supervised data distillation (DD), in which the learner exploits all available labeled data plus unlabeled data from the Internet. The work reveals a strong connection between the improvement of student model and the amount of unlabeled data used. DD proposes to use final output from the teacher model as the distillation model. These labels can be generated from the soft targets by selecting high-confident bounding boxes. However, the low-confident samples dropped by non maximum suppression (NMS) are of great importance in knowledge transferring as well because these samples are often the ones the student

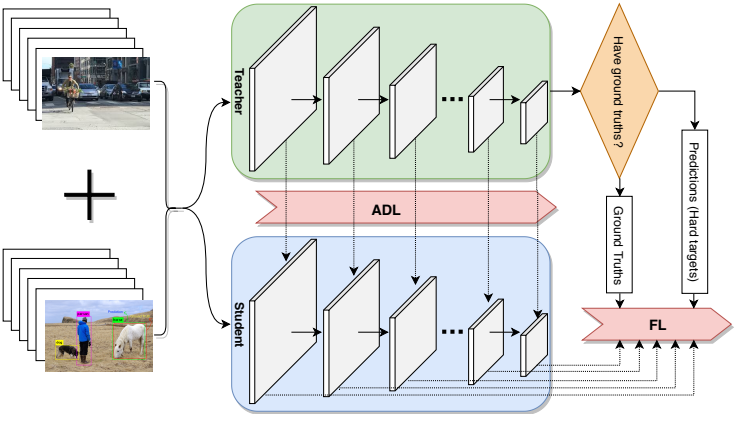

Figure 1: Semi-supervised adaptive distillation (SAD) framework. To begin with, the teacher model selects and annotates samples with at least one annotation and then combines those selected samples with labeled ones. The student is then trained using $ADL+FL+L_{loc}$ guided by the teacher.

model has troubles with. Thus, we propose to use the combination of both hard and soft targets in our framework. The steps are as follows: 1) train the teacher model with labeled data; 2) generate hard targets for unlabeled data using the teacher model; 3) train the student model with both labeled data and unlabeled data using soft targets and hard targets.

4 Experiments

We evaluate our work on the detection task of COCO benchmark [12]. We report all results by evaluating methods on the *mini-val* (5k images) or *test-dev* (41k images) split with the standard metrics of average precision following the COCO definitions, including AP, AP_{50}, AP_{75}.

Notation: In the following experiments, 115k COCO labeled images are represented as co-115 while 120k COCO unlabeled images are represented as un-120

Optimization and loss: We evaluate our technique using RetinaNet, one of the state-of-the-art single stage detectors. All the hyper-parameters are the same as [16]. All the models are trained with synchronized SGD over 8 GPUs with a total of 16 images per mini-batch (2 images per GPU). For training models only using co-115, we set the number of iterations to 90000. For training models on both co-115 and un-120, a iteration size of 270000 is used. The learning rate is divided by 10 at 70% and 90% of the total number of iterations. The hyper-parameter γ is set as the same for soft target loss and hard target loss, which is 2.

Student-teacher pairs: We validate our technique in different student-teacher pairs. We first investigate the performance improvement through distillation when the input size is reduced. In our implementation of KD, the teacher and student models have the same number of output logits, although input sizes are different. We then simply add a deconvolutional layer on top of the final feature map of the student model to match the size of the teacher's final feature map. We refer this model as ResNet-50-up. In addition to distillation over different input sizes, we also examine distillation over detectors with different capacities, in which a strong teacher model distills a weak student model. For example, ResNet-50 distills

β	AP	AP50	AP75
Baseline (Student)	28.8	45.8	30.6
0	28.9	45.9	30.6
0.5	29.4	46.3	31.2
1.0	30.5	48.5	32.7
1.5	30.7	48.8	32.7

Method	AP	AP50	AP75
Baseline (Student)	28.8	45.8	30.6
Feature map mimic	28.8	45.8	30.6
FDL	28.5	45.5	30.2
ADL	**30.7**	**48.8**	**32.7**

Table 1: The left table show results on varying β of ADL with half ResNet as student and ResNet as teacher. The performance increases as β becomes larger. The right table show results for different distillation methods with half ResNet as student and ResNet as teacher. Feature map mimic is to minimize L2 loss between the student and the teacher. *FDL* is introduced in Equation (4)

.

half ResNet-50. We refer half ResNet-50 as ResNet-50-half in the following experiments. Experiments with several pairs of teacher and student models are also conducted.

4.1 Adaptive Distillation Study

We compare different settings of our technique. We use ResNeXt-50 as the teacher and ResNet-50-half as the student if not specified. The scale is 800*1333.

Feature map mimic: First we evaluate the method of naive logits mimic using L2 loss. The entire feature map regression is implemented through the mimic mentioned in [13]. Results of logits mimic and entire feature map mimic are shown in Table 1. The mimicked models do not obtain any improvement compared to the baseline.

Focal distillation loss: We evaluate the loss function that adopts the same focal term between hard targets and soft targets. Surprisingly, the performance of the student model drops from 28.8 to 28.5. We attribute the performance decrease to the reason that the supervision of the ground truth in the focal term is so strong that it neglects the effect of the soft targets. The gradient is 0 when p is equal to the ground truth. In other words, *FDL* is not minimum when p is equal to q.

Adaptive distillation loss: Results using the proposed *ADL* with varying β are shown in Table. 1. When β is 0, which is equivalent to *DW*, *ADL* does not work well, since the weights of hard-to-mimic samples are very small. The performance improves as β becomes larger. With $\beta = 1.5$, ADL yields nearly 2 AP improvement over the student. Compared with *FDL*, our proposed *ADL* has the property that the loss is minimum when the output p produced by the student is equal to q produced by the teacher. Hence, we use $\beta = 1.5$ for all the following experiments.

ADL under different student-teacher pairs: In Table 2, we show distillation results using co-115 over different student-teacher pairs, with *ADL*. The performance of the student models improves significantly with distillation, despite architectural differences between the teacher and student. In general, the weak student model achieves over 1% improvement in mAP and 2% in AP50.

4.2 Semi-supervised Adaptive Distillation Study

We also conduct different experiments to evaluate our technique using un-120. The results are summarized in Table 3.

Model	scales	AP	AP50	AP75
ResNet-50 (T)	800	35.4	54.6	37.9
ResNet-50-up (S)	400	29.8	48.8	30.9
ADL	400	**31.2**	**50.9**	**32.5**
ResNet-50 (T)	800	35.4	54.6	37.9
ResNet-50-half (S)	800	28.8	45.8	30.6
ADL	800	**30.7**	**48.8**	**32.7**
ResNeXt-101 (T)	600	37.9	57.2	40.6
ResNet-50 (S)	600	34.3	53.2	36.9
ADL	600	**35.2**	**54.1**	**37.7**

Table 2: Distillation with co-115k using *ADL* over different student-teacher pairs. ResNet-50-up refers to the network with deconvolutional layer on the top and ResNet-50-half refers to ResNet-50 with half channel numbers. S stands for student and T stands for teacher. The notation is the same in the following table.

Student (scale)	Teacher (scale)	co-115 GT	co-115 ST	un-120 HT	un-120 ST	AP	AP50	AP75
ResNet-50-up (400)	ResNet-50 (800)	✓				28.8	45.8	30.6
		✓		✓		32.1	51.6	33.9
		✓	✓		✓	32.3	51.3	34.1
		✓	✓	✓	✓	**33.2**	**53.2**	**35.1**
ResNet-50-half (800)	ResNet-50 (800)	✓				28.8	45.8	30.6
		✓		✓		32.1	50.6	34.2
		✓	✓		✓	32.3	50.3	34.6
		✓	✓	✓	✓	**33.1**	**52.1**	**35.2**
ResNet-50 (600)	ResNeXt-101 (600)	✓				34.3	53.2	36.9
		✓		✓		35.6	54.7	37.9
		✓	✓		✓	35.9	54.9	38.5
		✓	✓	✓	✓	**36.6**	**55.8**	**38.9**

Table 3: Distillation results using un-120 under different settings. Ground truths are abbreviated as GT. Soft targets are abbreviated as ST. Hard targets are abbreviated as HT, representing hard targets produced by the teacher. The notation co-115 is the 115k COCO training set with annotations while un-120 is the 120k COCO unlabeled set.

Experiment setting: The notation co-115 ST or un-120 ST represents soft targets produced by the teacher. Ground truths are abbreviated as GT while hard targets are abbreviated as HT. Hard targets are predicted by the teacher using the method introduced in [18].

Effect of un-120: First, we investigate the method introduced in [18]. Following the protocol, we generate annotations for un-120k by selecting a threshold that makes "the average number of annotated instances per unlabeled image" roughly equal to "the average number of instances per labeled image". Compared with the student model only using co-115 GT, the use of un-120 yields significant improvement.

Effect of soft targets and hard targets in un-120: We use both the soft targets and hard targets of un-120 in the training stage. Note that the soft targets and hard targets are all from the teacher model. Compared with the distillation model, using only hard targets or only the soft targets of un-120k, the combination of them yields over 1 AP improvement in all student-teacher pairs. This shows both the more accurate samples (hard targets) and more informative samples (soft targets) is effective in knowledge transfer.

4.3 Overall Performance

Student Model Out-perform Teacher Model: We show that the student model can outperform the teacher model with our ADL under the semi-supervised setting. In this experi-

Model	Scale	AP	AP50	AP75
T (ResNet-101)	600	36.0	54.8	38.7
S (ResNet-50)	600	34.3	53.2	36.9
SAD	600	**36.3**	**55.2**	**38.9**
T (ResNeXt-101)	500	36.6	55.5	39.3
S (ResNet-101)	500	34.4	52.7	36.9
SAD	500	**36.8**	**55.7**	**39.4**

Table 4: Results on the union of co-115 and un-120 using the proposed semi-supervised adaptive distillation. We show that the student can out-perform its teacher.

	AP	time
YOLOv2 [19]	21.6	25
SSD321 [17]	28.0	61
DSSD321 [4]	28.0	85
R-FCN [3]	29.9	85
SSD513 [17]	31.2	125
DSSD513 [4]	33.2	156
FPN-FRCN [15]	36.2	172
RetinaNet-50-800	35.7	123
RetinaNet-101-500	34.4	90
RetinaNet-101-800	37.8	190
RetinaNet-50-600 (SAD)	**36.7**	90
RetinaNet-101-500 (SAD)	**36.9**	90

Table 5: Speed (ms) versus accuracy (AP) on COCO *test-dev* , which has no public labels and requires evaluation on servers. Our detector has achieves an AP of 36.8, running at 90 ms per image. The distilled detector is more accurate and faster than RetinaNet-50-800.

ment, it is shown that when the teacher is better than the student without knowledge transfer, the student can out-perform its teacher with knowledge transfer. We carry out our experiments with two student-teacher pairs: 1) (ResNet-50, ResNet-101) pair; 2) (ResNet-101, ResNeXt-101) pair. The teacher is trained on co-115 and the student is trained on the union of co-115 and un-120. The performance of the teachers model are 1.7 and 2.2 higher than that of the student respectively, but the student with knowledge transfer can still beat the teacher by some margin. In Table 4, the ResNet-50 (600) [8] student detector can reach to a mAP of 36.6 guided by the ResNeXt-101 [26] teacher, only slightly higher than the one guided by ResNet-101 in this experiment. We argue that this limitation of (ResNet-50, ResNeXt-101) pair is caused by the limited amount of data in the transfer set.

Comparison with other detectors: We compare our distilled detector with different detectors. As shown in Table 5, our detector is significantly better than any other detector except FPN-FRCN and RetinaNet-101-800. With a comparable mAP with the above two detectors, our distilled detector runs 2$\times$ faster.

5 Conclusion

In this paper, we design an adaptive distillation loss for single-stage detectors and demonstrate its effectiveness with RetinaNet in a typical distillation setting. We also propose a semi-supervised learning framework using this adaptive distillation. Our experimental results prove the student model can gain significant improvement using both hard targets and

soft targets produced by the teacher from unlabeled data. The student even out-performs the teacher given enough transferred data.

References

[1] Olivier Chapelle, Bernhard Scholkopf, and Alexander Zien. Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]. *IEEE Transactions on Neural Networks*, 20(3):542–542, 2009.

[2] Guobin Chen, Wongun Choi, Xiang Yu, Tony Han, and Manmohan Chandraker. Learning efficient object detection models with knowledge distillation. In *Advances in Neural Information Processing Systems*, pages 742–751, 2017.

[3] Jifeng Dai, Yi Li, Kaiming He, and Jian Sun. R-fcn: Object detection via region-based fully convolutional networks. In *Advances in neural information processing systems*, pages 379–387, 2016.

[4] Cheng-Yang Fu, Wei Liu, Ananth Ranga, Ambrish Tyagi, and Alexander C Berg. Dssd: Deconvolutional single shot detector. *arXiv preprint arXiv:1701.06659*, 2017.

[5] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[6] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

[7] Song Han, Jeff Pool, John Tran, and William Dally. Learning both weights and connections for efficient neural network. In *Advances in neural information processing systems*, pages 1135–1143, 2015.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015.

[9] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[10] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[11] Durk P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. Semi-supervised learning with deep generative models. In *Advances in neural information processing systems*, pages 3581–3589, 2014.

[12] Vadim Lebedev, Yaroslav Ganin, Maksim Rakhuba, Ivan Oseledets, and Victor Lempitsky. Speeding-up convolutional neural networks using fine-tuned cp-decomposition. *arXiv preprint arXiv:1412.6553*, 2014.

[13] Quanquan Li, Shengying Jin, and Junjie Yan. Mimicking very efficient network for object detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7341–7349. IEEE, 2017.

[14] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[15] Tsung-Yi Lin, Piotr Dollár, Ross B Girshick, Kaiming He, Bharath Hariharan, and Serge J Belongie. Feature pyramid networks for object detection. In *CVPR*, volume 1, page 4, 2017.

[16] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. *arXiv preprint arXiv:1708.02002*, 2017.

[17] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[18] Ilija Radosavovic, Piotr Dollár, Ross Girshick, Georgia Gkioxari, and Kaiming He. Data distillation: Towards omni-supervised learning. *arXiv preprint arXiv:1712.04440*, 2017.

[19] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. *arXiv preprint*, 2017.

[20] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[21] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[22] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. *arXiv preprint arXiv:1412.6550*, 2014.

[23] Chuck Rosenberg, Martial Hebert, and Henry Schneiderman. Semi-supervised self-training of object detection models. In *WACV/MOTION*, pages 29–36, 2005.

[24] Abhinav Shrivastava, Abhinav Gupta, and Ross Girshick. Training region-based object detectors with online hard example mining. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 761–769, 2016.

[25] Cheng Tai, Tong Xiao, Yi Zhang, Xiaogang Wang, et al. Convolutional neural networks with low-rank regularization. *arXiv preprint arXiv:1511.06067*, 2015.

[26] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500, 2017.

[27] Junho Yim, Donggyu Joo, Jihoon Bae, and Junmo Kim. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, 2017.

[28] X Zhang, X Zhou, M Lin, and J Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. arxiv 2017. *arXiv preprint arXiv:1707.01083*, 2017.

[29] Aojun Zhou, Anbang Yao, Yiwen Guo, Lin Xu, and Yurong Chen. Incremental network quantization: Towards lossless cnns with low-precision weights. *arXiv preprint arXiv:1702.03044*, 2017.

[30] Xiaojin Jerry Zhu. Semi-supervised learning literature survey. Technical report, University of Wisconsin-Madison Department of Computer Sciences, 2005.