Distilling Object Detectors with Fine-grained Feature Imitation

Tao Wang[1]   Li Yuan[1]   Xiaopeng Zhang[1,2]   Jiashi Feng[1]

[1]Department of Electrical and Computer Engineering, National University of Singapore, Singapore

[2]Huawei Noah's Ark Lab, Shanghai, China

twangnh@gmail.com   ylustcnus@gmail.com   zhangxiaopeng12@huawei.com   elefjia@nus.edu.sg

Abstract

State-of-the-art CNN based recognition models are often computationally prohibitive to deploy on low-end devices. A promising high level approach tackling this limitation is knowledge distillation, *which let small student model mimic cumbersome teacher model's output to get improved generalization. However, related methods mainly focus on simple task of classification while do not consider complex tasks like object detection. We show applying the vanilla knowledge distillation to detection model gets minor gain. To address the challenge of distilling knowledge in detection model, we propose a* fine-grained feature imitation *method exploiting the cross-location discrepancy of feature response. Our intuition is that detectors care more about local near object regions. Thus the discrepancy of feature response on the near object anchor locations reveals important information of how teacher model tends to generalize. We design a novel mechanism to estimate those locations and let student model imitate the teacher on them to get enhanced performance. We first validate the idea on a developed lightweight toy detector which carries simplest notion of current state-of-the-art anchor based detection models on challenging KITTI dataset, our method generates up to 15% boost of mAP for the student model compared to the non-imitated counterpart. We then extensively evaluate the method with Faster R-CNN model under various scenarios with common object detection benchmark of Pascal VOC and COCO, imitation alleviates up to 74% performance drop of student model compared to teacher. Codes released at* `https://github.com/twangnh/Distilling-Object-Detectors`

1. Introduction

Object detection has benefited a lot from recent advances of deep CNN architectures. However state-of-art detectors are cumbersome to deploy on low computation devices. Previous works mainly focus on Quantization [12, 14, 36, 28] which efficiently reduces computation
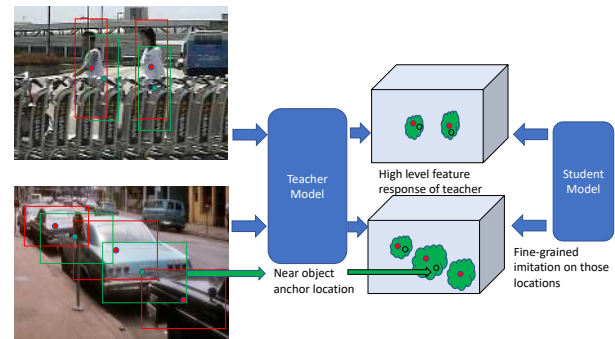

Figure 1. Illustration on principle of the proposed method. Red and green bounding boxes on the left two images are selected prior anchor boxes on corresponding locations. The red anchors have the largest overlap with ground truth bounding boxes and the green ones indicate near object samples. The motivation is that the discrepancy of feature response on near object anchor locations reveals how a learned teacher model tends to generalize (*e.g.*, how the teacher responses on those intersections of crowed objects compared to on-object locations reflects how it separates and detects those crowded instances). Our method thus first locates these knowledge-dense locations and let the student model imitate teacher's high-level feature responses on them.

and model size, and network pruning [14, 13, 1, 34] that prunes redundant connections in large models. However these approaches may require dedicated hardware or software customization to get practical speedup.

A promising high level method to directly learn compact models end-to-end is *knowledge distillation* [16]. A student model learns the behavior of a stronger teacher network to get enhanced generalization. However, prior works on knowledge distillation [16, 32, 38, 6, 18] are mostly devoted to classification and rarely consider object detection. A detection model may only involve a few classes, with which much less knowledge can be distilled from inter-class similarity of teacher's softened outputs. Also, detection requires reliable localization in addition to classification, vanilla distillation can not be applied for distilling localization knowledge. Besides, the extreme imbalance of foreground and background instances also makes bounding box annotations

less voluminous. We find that merely adding distillation loss only gives minor boost for student (ref. Sec. 4.2).

Similar to knowledge distillation, hint learning [32] improves student models by minimizing the discrepancy of full high level features of the teacher and student models. But we find that directly applying hint learning on detection model hurts performance (ref. Sec. 4.2). The intuition is that detectors care more about local regions that overlap with ground truth objects while classification models pay more attention to global context. So directly doing full feature imitation would unavoidably introduces large amount of noise from uncared areas, especially for object detection where background instances are overwhelming and diverse.

Recall in knowledge distillation, relative probabilities on different classes indeed tell a lot about how the teacher model tends to generalize. Similarly, since detectors care more about local object regions, the discrepancy of feature response on close anchor locations near the object also conveys important information about how a complex detection model detects the object instances. Aiming to utilize this *inter-location discrepancy* for distilling knowledge in object detector, we develop a novel mechanism exploiting ground truth bounding boxes and anchor priors to effectively estimate those informative near object anchor locations, then make student model imitate teacher on them, as shown in Figure 1.

We term this method as *fine-grained feature imitation*. Our method effectively addresses the above mentioned challenge: 1) We do not rely on softened output of teacher model as in vanilla knowledge distillation of classification model, but depends on a *inter-location* discrepancy of teacher's high level feature response. 2) Fine-grained feature imitation before classification and localization heads improves both sub-tasks. We show in Sec 4.4.2 and Sec 4.4.3 that our method effectively enhanced the student model's ability on class discrimination and localization. 3) Our method avoids those noisy less informative background area which leads to degraded performance of full feature imitation, study of the *per-channel variance* on high level feature maps in Sec 4.4.5 validates this intuition.

To validate our method, we first experiment on a developed lightweight toy detector that carries main principle of current state-of-the-art anchor based detection models. Applying the method to this lightweight architecture, we can produce much smaller model with up to 15% boost of mAP compared to the non-imitated counterpart. We then perform extensive experiments on the state-fo-the-art Faster R-CNN model under various scenarios including imitation over shallow student, halved student and multi-layer imitation, on the widely used common object detection benchmarks of PASCAL VOC [7] and MSCOCO [23]. The experiments demonstrate the broad applicability and superior performance of our proposed method.

2. Related Works

Object detection Recently with the development of deep CNN model for image classification task, various approaches [10, 9, 31, 4, 29, 30, 24, 22] are proposed for object detection which significantly outperform traditional methods. The line of works are pioneered by R-CNN [10] that extracts and classifies each region of interest (ROI) to detect objects. [9, 31] extend and improve the framework for improved performance. One-stage detectors [29, 24] are proposed driven by the requirement of real time inference. Similarly we design the lightweight detector partly for implementation on mobile devices.

Knowledge distillation Following the seminal work [15], various knowledge distillation approaches were proposed [32, 38, 6, 18]. Hint learning [32] explores an alternative way for distillation, where the supervision from teacher models comes from high level features. [38] proposed to force the student model to mimic the teacher model on the features specified by an attention map. [6] proposed to exploit relationship between different samples, and utilizes cross sample similarities to improve distillation. [18] formalizes distillation as a distribution matching problem to optimize the student model. A few recent works explored distillation approach for compressing detection models. [5] tried adding both full feature imitation and specific distillation loss on detection heads, but we find full feature imitation brings degraded performance for student model and it is unclear how to deal with region proposal [11] inconsistency between teacher and student when performing the distillation. [20] proposed to only transfer knowledge under the area of proposals, but the mimicking regions depend on the output of model itself and it is not applicable for one-stage detector.

Model acceleration To speed up deep neural network model without losing accuracy, quantization [40, 28, 37, 12, 14, 36] uses low-precision model parameter representation. Connection pruning or weight sparsifying [14, 13, 27] prune redundant connections in large models. However, these approaches require specific hardware or software customization to get practical speedup. For example, weight pruning needs support of sparse computations and quantization relies on low-bit operations. Some prior works [19, 25, 2] propose to do channel level pruning. But when pruning ratio is higher, those methods unavoidably hurt performance significantly. Some works employ low rank approximation to large layers [33, 35]. But the actual speedup are usually much less than theoretical values.

3. Method

In this work, we developed a simple to implement fine-grained feature imitation method utilizing inter-location discrepancy of teacher's feature response on near object
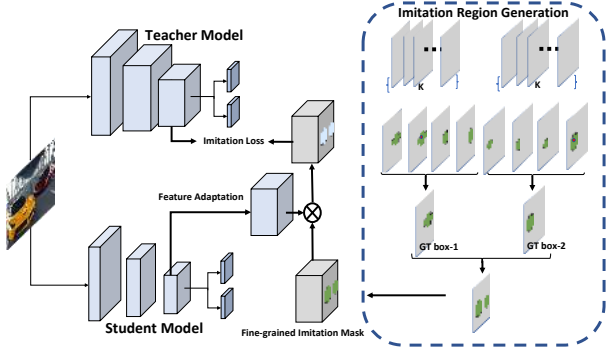
Figure 2. Illustration of the proposed fine-grained feature imitation method. The student detector is trained by both ground truth supervision and imitating teacher's feature response on close object anchor locations. The feature-adaptation layer makes student's guided feature layer compatible with the teacher. To identify informative locations, we iteratively calculate IOU map of each groundtruth bounding box with anchor priors, filter and combine candidates, and generate the final imitation mask, ref. to Sec. 3.1 for details.

anchor locations for distilling the knowledge in cumbersome detection models. Our Intuition is that the discrepancy of feature response on the near object anchor locations reveals important information of how large detector tends to generalize, with which learned knowledge can be distilled. Specifically, we propose a novel mechanism to estimate those anchor locations which forms fine-grained local feature regions close to object instances, and let a student model imitate teacher model's high level feature response on those regions to get enhanced performance. This intuitive method is general for current state-of-the-art anchor based detection models (*e.g.*, Faster R-CNN [31], SSD [24], YOLOV2 [30]), and is orthogonal to other model acceleration methods including network pruning and quantization.

3.1. Imitation region estimation

As shown in Fig. 1, the near object anchor locations form local feature region for each object. To formally define and study the local feature region, we utilize ground truth bounding boxes and anchor priors to calculate those regions as a mask I for each independent image, and control the size of regions by a thresholding factor ψ. In the following, with feature maps, we always refer to the last features where anchor priors are defined on [31].

Specifically, as shown in Fig. 2, for each ground truth box, we compute the IOU between it and all anchors, which forms a $W \times H \times K$ IOU map m. Here W and H denote width and height of the feature map, and K indicates the K preset anchor boxes. Then we find the largest IOU value $M = max(m)$, times the thresholding factor ψ to obtain a filter threshold $F = \psi * M$. With F, we filter the IOU map to keep those larger then F locations and combine them with OR operation to get a $W \times H$ mask. Loop over all

ground truth boxes and combine the masks, we get the final fine-grained imitation mask I.

When $\psi = 0$, the generated mask includes all locations on the feature map while no locations are kept when $\psi = 1$. We can get varied imitation mask by varying ψ. In all experiments, a constant $\psi = 0.5$ is used. We show $\psi = 0.5$ offers the best distillation performance in detailed ablation study (ref. to Sec 4.4.4). The reason we do not use fixed value of F to filter the IOU map is that object size usually varies in a large range. Fixed threshold values would be biased for objects at certain scales and ratios (ref. Sec. 4.2).

3.2. Fine-grained feature imitation

In order to carry out imitation, we add a full convolution adaptation layer after corresponding student model before calculating distance metric between student and teacher's feature response, as shown in Figure 2. We add the adaptation layer for two reasons: 1) The student feature's channel number may not be compatible with teacher model. The added layer can align the former to the later for calculating distance metric. 2) We find even when student and teacher have compatible features, forcing student to approximate teacher feature directly leads to minor gains compared to the adapted counterpart.

We now introduce the feature imitation details. Define s as student model's guided feature map and t as corresponding teacher's feature map. For each near object anchor location (i, j) on the feature map of width W and height H, we train student model to minimize the following objective:

$$l = \sum_{c=1}^{C}(f_{\text{adap}}(s)_{ijc} - t_{ijc})^2, \tag{1}$$

to learn the teacher detection model's knowledge. Together with all estimated near anchor location(the imitation mask I), the distillation objective is to minimize:

$$L_{imitation} = \frac{1}{2N_p}\sum_{i=1}^{W}\sum_{j=1}^{H}\sum_{c=1}^{C}I_{ij}(f_{\text{adap}}(s)_{ijc} - t_{ijc})^2,$$

$$\text{where } N_p = \sum_{i=1}^{W}\sum_{j=1}^{H}I_{ij}. \tag{2}$$

Here I is the imitation mask, N_p is the number of positive points in the mask, $f_{\text{adap}}(\cdot)$ is the adaptation function. Then the overall training loss of a student model is:

$$L = L_{gt} + \lambda L_{imitation}, \tag{3}$$

where L_{gt} is the detection training loss and λ is imitation loss weight balancing factor.

Models	Flops/G	Params/M	car			pedestrian			cyclist			mAP
			Easy	Moderate	Hard	Easy	Moderate	Hard	Easy	Moderate	Hard	
1×	5.1	1.6	84.56	74.11	65.64	65.28	55.95	50.79	70.39	50.09	46.88	62.63
0.5×	1.5	0.53	76.39	68.35	59.74	63.69	54.34	49.58	64.52	43.67	41.57	57.98
0.5×-I	1.5	0.53	80.56	71.46	61.71	64.18	54.62	49.95	68.25	48.28	45.09	60.46
-	-	-	**+4.2**	**+3.1**	**+2.0**	**+0.5**	**+0.3**	**+0.4**	**+3.7**	**+4.6**	**+3.5**	**+2.5**
0.25×	0.67	0.21	60.36	54.85	46.56	52.41	43.63	39.84	51.35	33.41	31.26	45.96
0.25×-I	0.67	0.21	74.26	61.63	53.94	59.80	50.15	46.28	54.64	38.13	34.84	52.63
-	-	-	**+13.9**	**+6.8**	**+7.4**	**+7.4**	**+6.5**	**+6.4**	**+3.3**	**+4.7**	**+3.6**	**+6.7**
0.25×-F	0.67	0.21	-12.9	-14.5	-11.3	-2.9	-1.9	-1.3	-16.7	-9.3	-9.4	-8.9
0.25×-G	0.67	0.21	+8.8	+2.3	+1.2	+3.1	+0.8	+2.4	-0.5	-0.1	-0.3	+2.0
0.25×-D	0.67	0.21	+3.5	+1.2	+1.3	+1.1	+0.8	+0.3	+0.2	-0.3	-0.1	+0.9
0.25×-ID	0.67	0.21	+10.8	+5.8	+6.3	+6.2	+4.1	+3.6	+2.2	+4.7	+3.1	+5.2

Table 1. Imitation result on the toy detector and results of some comparing methods. 1× is the base detector, 0.5× and 0.25× are directly pruned model trained with ground truth supervision, serving as baselines. -I means with additional proposed imitation loss, -F indicates with full feature imitation, -G means using directly scaled ground truth boxes as imitation region, -D means adding only vanilla distillation loss, -ID indicates the case that both proposed imitation loss and distillation loss are imposed.

4. Experiments

To validate our method, we first perform experiments on a developed lightweight toy detector with the KITTI detection benchmark which contains three road object classes. We then further validate the method on state-of-the-art Faster R-CNN model under various network setting with widely used common object detection benchmarks. The toy detector carries simplest principle of state-of-the-art anchor based detection model, while the performance is not comparable to those cumbersome and multi-stage stage or multi-layer detection models, it can applied to mobile devices. All quantitative results are evaluated in average precision (AP).

4.1. Lightweight detector

We first present a manually designed lightweight detector for evaluating the performance enhancement of the proposed imitation method. This detector is based on the Shuffenet [39] which gives excellent classification performance with limited flops and parameters. However, the Shufflenet architecture itself is dedicated for image classification. We find directly adapting it to detection produces terrible result. This is because each point on the top feature map has an equivalent stride of 32, leading to very coarse alignment of anchor boxes on the input images. Moving to lower output layer with smaller stride also performs not well as features are less powerful therein.

To address the above deficiencies, we make the following refactoring and develop an improved one-stage lightweight model for detection. (1) We change stride of Conv1 from 2 to 1. The original network design quickly downsamples the input image to reduce computational cost. But object detection requires higher resolution feature to make downstream feature decoder (the detector heads) work well. Such modification enables utilization of all con-

volution layers while preserves high resolution for the top feature map. (2) We modify the output channel of Conv1 from 24 to 16, which reduces memory footprint and computation. (3) We reduce the block number of stage-3 from 8 to 6. We find such modification leads to slightly lower pre-training precision, but does not hurt detection performance. The overall runtime is reduced significantly. (4) We add two additional shufflenet blocks which are trained from scratch before the regression and classification head. The added blocks provide additional adaptation of the high level feature for detection. (5) We employ very simple RPN-alike detector which discriminate between classes. Unlike previous layers, the detection heads use full convolution, while parameters are increased, we find this significantly improves accuracy. We refer such lightweight base detector as 1× in the following sections. Refer to the supplementary material for architecture diagram of the model.

4.2. Imitation with lightweight detectors

We first apply the proposed method to the toy detector presented above. We use the base model as teacher (denoted as 1×), and directly halve channels of each layer for student model. Specifically, we halve once of teacher model to get the 0.5× model, and halve twice (75% channels removed) to obtain the 0.25× model. We conduct the experiments on challenging KITTI [8] dataset. Since test set annotation is not available, we follow [3, 26] to split training dataset into training and validation sets and carefully make sure they do not come from the same video sequence. We use the official evaluation tool to evaluate detector performance on the validation set. Table 3.2 shows overall imitation results of the student models, as well as comparison to other methods. It is well known that reduction on parameters and computation always brings exponential performance drop, *e.g.*,

the $0.5\times$ model sacrifices only around 4.7 mAP compared to the teacher, while $0.25\times$ halving results in 16.7 mAP drop. In such hard cases, the presented method still achieves significant boost for student models, *i.e.*, the $0.5\times$ model gets 2.5 mAP improvement, the $0.25\times$ model is boosted by 6.6 mAP ($0.25\times$-I), which is 14.7% of the non-imitated one. Note the improvement for $0.5\times$ model on pedestrian is smaller than other classes as the gap between teacher and non-imitated student is minor on pedestrian.

We conduct experiments on 4 comparing settings with the $0.25\times$ model. As shown in last 4 lines of Table 3.2. The first is hint learning [32] (*i.e.* full feature imitation, denoted as $0.25\times$-F). Though performing well for classification, it brings large performance drop (8.9 mAP) to the original $0.25\times$ model. We conjecture this is because background noise overwhelms the informative supervision signal from teacher model which is verified in Sec. 4.4.5. The very simple setting ($0.25\times$-G) of directly scaling ground truth boxes with same stride on the feature layer and applying imitation on those areas gives much less gain than the proposed method. The reason is that while noise from background regions is avoided, the method also missed the important supervision from some near object locations. In the third setting ($0.25\times$-D), we find adapting the vanilla knowledge distillation [16] to detection setting produces unpleasant result (only 0.9 increase of mAP), verifies our intuition in Sec. 1. Finally, we try to combine distillation loss with imitation loss (denoted as 0.25-ID), but the performance is worse than only using imitation term, implying high level feature imitation and distillation on model outputs have very divergent objectives.

4.3. Imitation with Faster R-CNN

We further perform extensive experiments with the more general architecture of Faster R-CNN model under three settings: 1) halved student model. 2) shallow student model. 3) multi-layer imitation.

Halved student model   In this setting, we use Resnet101 based Faster R-CNN as teacher model and halve channel number of each layer including the fully connected layers to construct the student model. As shown in Table 4 and Table 2, we perform experiments with COCO and Pascal VOC07 dataset. Clearly halving the whole teacher model cause the performance to drop significantly. With imitation, the halved student model gets significant boost, *i.e.*, 2.8 absolute mAP gain both in Pascal style average precision and COCO style average precision with COCO dataset; and 3.8 absolute mAP gain for Pascal VOC07 dataset. The results demonstrate that our method can effectively distill the teacher detector's knowledge into the halved student.

Shallow student network   For this setting, instead of halving layer channels of teacher model, we choose shal-

lower student backbone with similar architecture of teacher model. Specifically, we perform two imitation experiments: VGG11 based Faster R-CNN as student and VGG16 based one as teacher; Resnet50 based Faster R-CNN as student and Resnet101 based one as teacher. As shown in Table 3, the shallow backbone based student model all gets significant improvement, especially for the VGG11 based student model, the imitated model gets 8.0 absolute gain in mAP, our method nearly recovers **74%** of the performance drop due to shallow backbone.

Multi-layer imitation   The previous imitation experiments are with single layer of feature map, we further extend the experiment to multi-layer imitation with seminal work of Feature Pyramid Networks (FPN) [21]. The FPN combined with Faster R-CNN framework perform region proposal on different layer with different anchor prior size, and pools feature on corresponding layer according to roi size. We compute the imitation region on each layer with corresponding prior anchors, and let student model imitate feature response on each layer. The teacher detection model is a Resnet50 FPN based Faster R-CNN, and student is a halved counterpart. As shown in Table 5, imitated student gets 3.2 absolute mAP gain in Pascal style average precision and 3.6 mAP gain with COCO style average precision.

4.4. Analysis

4.4.1   Visualization of imitation mask

To better understand the imitation region generated by our approach, we visualize some example masks I on input image with the toy detector given sample from KITTI dataset. Specifically we scale the generated imitation mask I on the feature map to input image with corresponding stride(16 for the toy detector). Fig 3 shows example imitation masks scaled and overlaid on input image. Of the 6 images, Fig 3(a) is original image; Fig 3(b) 3(c) 3(d) are generated with $\psi = 0.2$, $\psi = 0.5$, and $\psi = 0.8$ respectively; Fig 3(e) 3(f) are filtered with constant threshold value of $F = 0.5$ and $F = 0.8$ respectively. It is obvious that some objects are missing with only $F = 0.5$, and nearly all imitation mask disappeared with $F = 0.8$. This is because constant filter threshold of F biases for those ground truth boxes of similar size with prior anchors. Our method with adaptive filter threshold greatly mitigates this problem.

4.4.2   Qualitative performance gain from imitation

In this subsection, we present some sampled detection outputs reflecting the enhanced ability of student detector through the imitation learning. The results are from VGG11 based Faster R-CNN model on VOC07 dataset (ref. to Table 3 for quantitative results). We only show one example for each type of gain due to space limited, and choose

Model	mAP	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv
res101	74.4	77.8	78.9	77.5	63.2	62.6	79.2	84.4	85.6	54.5	81.5	68.7	85.7	84.6	77.8	78.6	47.1	76.3	74.9	78.8	71.2
res101h	67.4	73.9	78.6	66.3	52.5	42.4	73.8	80.4	80.1	43.5	71.8	61.9	78.7	81.7	74.4	76.8	42.2	66.9	65.	74.3	62.8
res101h-I	71.2	77.2	80.0	72.9	56.0	50.4	77.1	82.3	85.5	47.4	80.2	59.9	84.3	83.9	73.8	79.1	44.6	70.8	69.4	78.7	70.4
	+3.8	+3.3	+1.4	+6.6	+3.5	+8.0	+3.3	+1.9	+5.4	+3.9	+8.4	-2.0	+5.6	+2.2	-0.6	+2.3	+2.4	+3.9	+4.4	+4.4	+7.6

Table 2. Imitation with halved student model with Faster R-CNN model on Pascal VOC07 dataset.

Model	mAP	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv
VGG16	70.4	70.9	78.0	67.8	55.1	53.2	79.6	85.5	83.7	48.7	78.0	63.5	80.2	82.0	74.5	77.2	43.0	73.7	65.8	76.0	72.5
VGG11	59.6	67.3	71.4	56.6	44.3	39.3	68.8	78.4	66.6	37.7	63.2	51.6	58.3	76.4	70.0	71.9	32.2	58.1	57.8	62.9	60.0
VGG11-I	67.6	72.5	73.8	62.8	53.1	49.2	80.5	82.7	76.8	44.8	73.5	64.3	72.6	81.1	75.3	76.3	40.2	66.3	61.8	73.4	70.6
	+8.0	+5.2	+2.4	+6.2	+8.8	+9.9	+11.7	+4.3	+10.2	+7.1	+10.3	+12.7	+14.3	+4.7	+5.3	+4.4	+8.0	+8.2	+4.0	+10.5	+10.6
res101	74.4	77.8	78.9	77.5	63.2	62.6	79.2	84.4	85.6	54.5	81.5	68.7	85.7	84.6	77.8	78.6	47.1	76.3	74.9	78.8	71.2
res50	69.1	68.9	79.0	67.0	54.1	51.2	78.6	84.5	81.7	49.7	74.0	62.6	77.2	80.	72.5	77.2	40.0	71.7	65.5	75.0	71.0
res50-I	72.0	71.5	80.6	71.1	57.0	52.4	82.1	90.0	82.7	51.6	74.5	66.2	82.3	82.3	75.7	78.3	43.5	79.6	69.1	77.3	72.1
	+2.9	+2.6	+1.6	+4.1	+2.9	+1.2	+3.5	+5.0	+1.0	+1.9	+0.5	+3.6	+5.1	+2.3	+3.2	+1.1	+3.5	+7.9	+3.6	+2.3	+1.1

Table 3. Imitation with shallow student model on Pascal-VOC07 dataset with Faster R-CNN model.
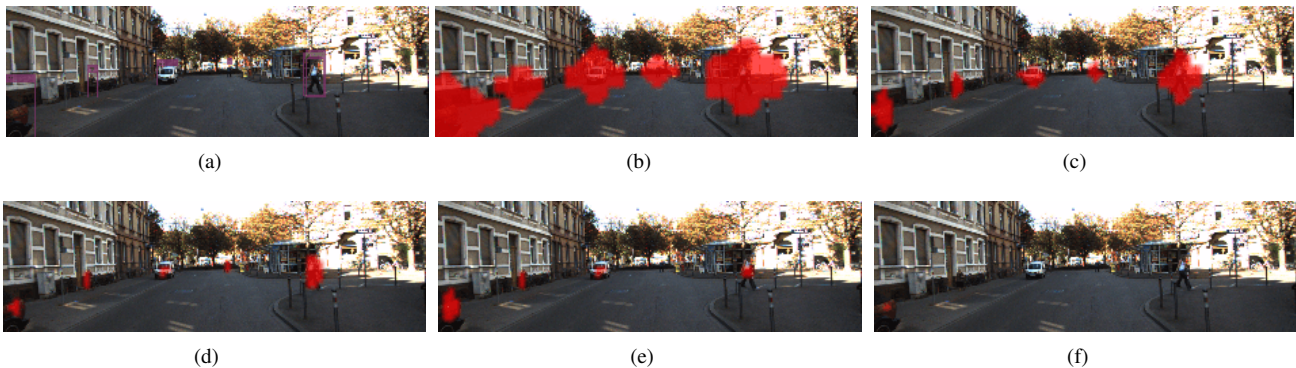

(a)

(b)

(c)

(d)

(e)

(f)

Figure 3. Examples of calculated imitation masks overlaid on input image. Note that the actual masks are calculated on last feature map, we enlarge the mask with corresponding ratio to display on the input image. (a) Original image. (b) $\psi = 0.2$. (c) $\psi = 0.5$. (d) $\psi = 0.8$. (e) Hard-thresh-0.5. (f) Hard-thresh-0.8. Thresh-* indicates different thresholding factor for proposed approach, Hard-thresh-* means using constant threshold of F when filtering the IOU map.

Model	AP@0.5	AP	AP_s	AP_m	AP_l	AR	AR_s	AR_m	AR_l
res101	54.6	34.4	14.3	39.1	51.9	45.9	23.0	52.2	66.4
res101h	48.4	28.8	11.8	32.0	44.9	41.5	19.8	45.9	62.3
res101h-I	51.2	31.6	13.2	35.9	47.5	44.0	22.4	50.3	64.5
	+2.8	+2.8	+1.4	+3.9	+2.6	+2.5	+2.6	+4.4	+2.2

Table 4. Imitation with halved student model with Faster R-CNN model on COCO dataset.

Model	AP@0.5	AP	AP_s	AP_m	AP_l	AR	AR_s	AR_m	AR_l
res50	59.0	36.9	21.5	39.8	48.3	50.5	31.4	53.9	63.6
res50h	52.6	31.2	18.5	32.0	42.4	46.3	27.7	47.5	60.6
res50h-I	55.8	34.8	21.0	34.9	45.5	49.1	30.5	52.6	63.5
	+3.2	+3.6	+2.5	+2.9	+3.1	+2.8	+2.8	+5.1	+2.9

Table 5. Result of multi-layer imitation on COCO dataset with Resnet50 FPN based Faster R-CNN model.

the examples containing simple objects for clearer visualization. In Fig 4, the upper row of detection outputs are from raw student model trained with ground truth supervision only, and the lower row of detection outputs are from imitated student model. The improvement of the student model with teacher supervision can be summarized into following aspects: *Improved discrimination ability.* As shown in Fig 4(a) and Fig 4(f), the color and style of lower part of the man's clothes is somewhat similar to that in some sofa objects. The raw student model mistakingly detect that as a sofa object with rather high confidence. While the imitated student avoids the error, indicating better discrimina-
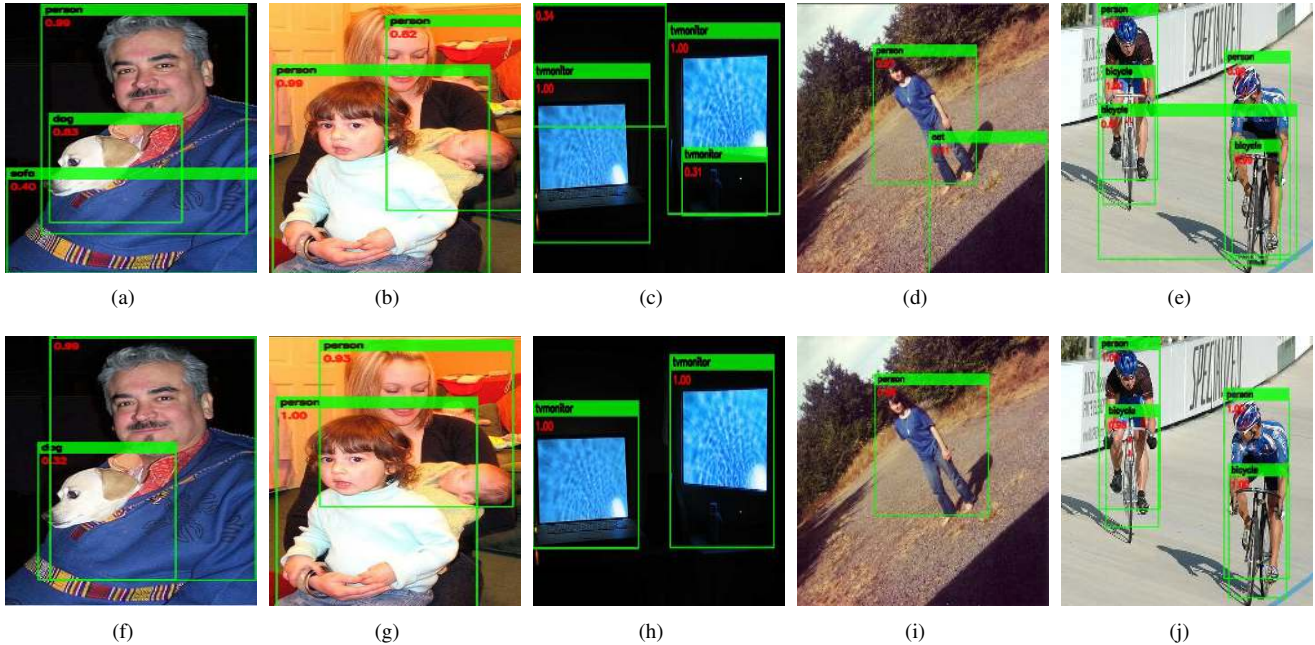
Figure 4. Qualitative results on the gain from imitation learning. The bounding box visualization threshold is set as 0.3. The top row images are student model's output without imitation, the bottom row shows imitated student's output.
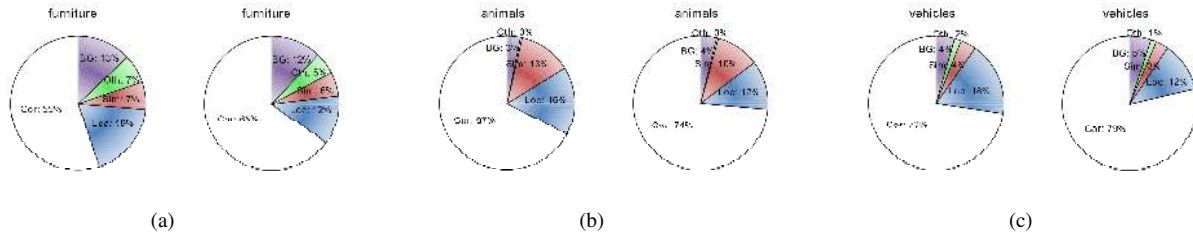

Figure 5. Imitation gain from error perspective with VGG11 based Faster R-CNN student and VGG16 based teacher on the Pascal VOC07 dataset. For each pair, the left figure corresponds to raw student model, and the right corresponds to imitated student.

tion ability. It is interesting to note that the imitated student has lower confidence on the dog instance compared to the raw student model, we have observed the teacher model (VGG16 based Faster R-CNN) outputs confidence of 0.38 for the instance. This phenomenon reveals that the teacher model's learned knowledge has been effectively transfered to the student model. *More reliable localization.* As shown in Fig 4(b) and Fig 4(g), the raw student model outputs a rather inaccurate location of the woman as a person instance. While the imitated student model learns better localization knowledge from the teacher and outputs a rather accurate bounding box for the person instance. *Less repeated detection.* As shown in Fig 4(c) and Fig 4(h), the raw student model outputs repeated detections for the tv-monitors which are unfortunately not able to be suppressed by NMS. While the imitated model predicts single bounding box for each object. This phenomenon indicates imitated student has better ability handling close to object input regions, this improvement comes from improved region proposal and enhanced ROI processing ability. *Less back-*

ground error. As shown in Fig 4(d) and Fig 4(i), the raw student model wrongly predict an area of background as a cat instance. While the imitated the student avoids the error, indicating lower background false positive prediction. *Avoiding grouped detection error.* We have observed grouped detection of near objects is a common error case for the raw student model, as shown in left image of Fig 4(e) and Fig 4(j). The imitated student gets improved ability in avoiding such error case.

4.4.3 Quantitative performance gain from imitation

We use the analysis tool from [17] to understand the type of detection errors reduced by imitating teacher model. The analysis is performed with the VGG11 Faster R-CNN student on Pascal VOC07 dataset (the teacher model is VGG16 based Faster R-CNN, ref. to Table 3 for average precision gain results). We present analysis on 3 grouped object class set: 1) vehicles. 2) animals containing all animals including person. 3) furniture including chair, dining table and sofa. The detections were classified into five groups: **1)** Correct
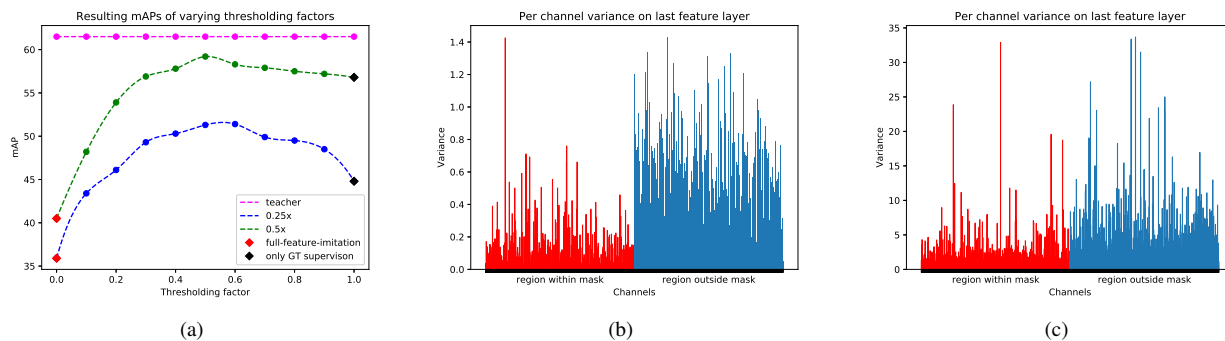
Figure 6. Results for further investigation of the method. (a) Varying imitation thresholding factor ψ for the toy detector experiment. (b),(c) Per-channel variance on high level feature map of learned teacher model. (b) is calculated with toy detector on KITTI dataset.(c) is calculated with Faster R-CNN on COCO dataset.

detection(Cor): correct class and $IoU > 0.5$. **2)** Localization (Loc): correct class, but misaligned bounding box ($0.1 < IoU < 0.5$). **3)** Similar (Sim): wrong class, correct category, $IoU > 0.1$. **4)** Other (Oth): wrong class and category, $IoU > 0.1$. **5)** Background (BG): $IoU < 0.1$ for any object class. Due to limited space we only present pie chart error percentage result, and defer to supplementary file for other analysis result. As shown in Fig 5, we observe for the three sub-set of object class, our method significantly improves the number of correct detections, and effectively reduces all other kinds of detection errors, especially for the Loc term. The error composition analysis reveals following important improvements: 1) Stronger localization ability (Loc); 2) less confusion between the same category and other category objects (Sim and Oth); 3) less background induced errors (BG).

4.4.4 Varying ψ for generating mask

To investigate the effects of region selection for imitation, we perform experiments on the $0.5\times$ and $0.25\times$ student models with varying thresholding factor ψ. We record mean value among three runs and plot the performance curve in Fig. 6(a) When $\psi = 0$, all points will be preserved and the method degenerates to full feature imitation as in hint learning. It is clear that imitated models are misguided severely. The mAP is even much lower than the ones trained with only ground truth supervision. As the threshold value increases, the student model performs much better, even with very low threshold of 0.1. This is strong evidence that the proposed approach effectively finds useful information while filters detrimental knowledges. The neutral value of 0.5 turns out to be optimal. When ψ is larger than 0.5, both students' mAP starts decreasing, but all the way still higher than when the value is 1.0, under which case the imitation reduces to only ground truth supervision. It is worth noting that when the ψ is larger than 0.5, the imitation regions quickly shrink and become extremely tiny and sparse, but imitation on those area still significantly boosts the students.

4.4.5 Per-channel variance of high level responses

To understand why full feature imitation produces deteriorated performance, we calculate the per-channel variance of the imitation feature map from a trained teacher model. We randomly sample and pass 10 images through the teacher model, calculate and record variances for anchor location within imitation region (with $\psi = 0.5$) and outside the region for each channel separately. Results are shown in Fig 6(b) and Fig 6(c) for the KITTI and COCO dataset on our $1\times$ toy detector and Resnet101 based Faster R-CNN model. Clearly the variances under the regions selected with proposed approach are smaller than those outside the areas, and holds for nearly all channels. This indicates that responses on background areas contain much noise. Features from the regions within the mask are more informative. Since convolution shares weights for whole feature map, directly imitating global feature responses would unavoidably accumulate large amount of noisy gradients from background areas. We also empirically observed that the loss value of full feature imitation is more than ten times that of proposed approach throughout training with same normalization method, which corroborates the analysis.

5. Conclusion

In this work, we developed a simple to implement *fine-grained feature imitation* method which employs the *inter-location* discrepancy of teacher detection model's feature response on near object anchor locations to distill the knowledge in a cumbersome object detector into a smaller one. Extensive experiments and analysis demonstrate the effectiveness of our method. Importantly, the method is orthogonal to and can be further combined with other model acceleration method including pruning and quantization.

References

[1] Jose M Alvarez and Mathieu Salzmann. Learning the number of neurons in deep networks. In *Advances in Neural Information Processing Systems*, pages 2270–2278, 2016.

[2] Dmitriy Anisimov and Tatiana Khanova. Towards lightweight convolutional neural networks for object detection. In *Advanced Video and Signal Based Surveillance (AVSS), 2017 14th IEEE International Conference on*, pages 1–8. IEEE, 2017.

[3] Zhaowei Cai, Quanfu Fan, Rogerio S Feris, and Nuno Vasconcelos. A unified multi-scale deep convolutional neural network for fast object detection. In *European Conference on Computer Vision*, pages 354–370. Springer, 2016.

[4] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. *arXiv preprint arXiv:1712.00726*, 2017.

[5] 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.

[6] Yuntao Chen, Naiyan Wang, and Zhaoxiang Zhang. Darkrank: Accelerating deep metric learning via cross sample similarities transfer. *arXiv preprint arXiv:1707.01220*, 2017.

[7] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.

[8] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

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

[10] 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.

[11] Ross B. Girshick. Fast r-cnn. *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1440–1448, 2015.

[12] Philipp Gysel, Mohammad Motamedi, and Soheil Ghiasi. Hardware-oriented approximation of convolutional neural networks. *CoRR*, abs/1604.03168, 2016.

[13] Song Han, Huizi Mao, and William J. Dally. Deep compression: Compressing deep neural network with pruning, trained quantization and huffman coding. *CoRR*, abs/1510.00149, 2015.

[14] Song Han, Jeff Pool, John Tran, and William J. Dally. Learning both weights and connections for efficient neural network. In *NIPS*, 2015.

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

[16] Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. *CoRR*, abs/1503.02531, 2015.

[17] Derek Hoiem, Yodsawalai Chodpathumwan, and Qieyun Dai. Diagnosing error in object detectors. In *European conference on computer vision*, pages 340–353. Springer, 2012.

[18] Zehao Huang and Naiyan Wang. Like what you like: Knowledge distill via neuron selectivity transfer. *arXiv preprint arXiv:1707.01219*, 2017.

[19] Hao Li, Asim Kadav, Igor Durdanovic, Hanan Samet, and Hans Peter Graf. Pruning filters for efficient convnets. *arXiv preprint arXiv:1608.08710*, 2016.

[20] 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.

[21] 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.

[22] Tsung-Yi Lin, Priyal Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. *IEEE transactions on pattern analysis and machine intelligence*, 2018.

[23] 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.

[24] 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.

[25] Jian-Hao Luo, Jianxin Wu, and Weiyao Lin. Thinet: A filter level pruning method for deep neural network compression. *arXiv preprint arXiv:1707.06342*, 2017.

[26] Jiayuan Mao, Tete Xiao, Yuning Jiang, and Zhimin Cao. What can help pedestrian detection? In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, page 3, 2017.

[27] Jongsoo Park, Sheng Li, Wei Wen, Ping Tak Peter Tang, Hai Li, Yiran Chen, and Pradeep Dubey. Faster cnns with direct sparse convolutions and guided pruning. *arXiv preprint arXiv:1608.01409*, 2016.

[28] Mohammad Rastegari, Vicente Ordonez, Joseph Redmon, and Ali Farhadi. Xnor-net: Imagenet classification using binary convolutional neural networks. In *European Conference on Computer Vision*, pages 525–542. Springer, 2016.

[29] 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.

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

[31] 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.

[32] 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.

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

[34] Wei Wen, Chunpeng Wu, Yandan Wang, Yiran Chen, and Hai Li. Learning structured sparsity in deep neural networks. In *Advances in Neural Information Processing Systems*, pages 2074–2082, 2016.

[35] Wei Wen, Cong Xu, Chunpeng Wu, Yandan Wang, Yiran Chen, and Hai Li. Coordinating filters for faster deep neural networks. *CoRR, abs/1703.09746*, 2017.

[36] Jiaxiang Wu, Cong Leng, Yuhang Wang, Qinghao Hu, and Jian Cheng. Quantized convolutional neural networks for mobile devices. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4820–4828, 2016.

[37] Jiaxiang Wu, Cong Leng, Yuhang Wang, Qinghao Hu, and Jian Cheng. Quantized convolutional neural networks for mobile devices. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4820–4828, 2016.

[38] Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. *arXiv preprint arXiv:1612.03928*, 2016.

[39] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. *CoRR*, abs/1707.01083, 2017.

[40] 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.