Instance, Scale, and Teacher Adaptive Knowledge Distillation for Visual Detection in Autonomous Driving

Qizhen Lan and Qing Tian

***Abstract*—Efficient visual detection is a crucial component in self-driving perception and lays the foundation for later planning and control stages. Deep-networks-based visual systems achieve state-of-the-art performance, but they are usually cumbersome and computationally infeasible for embedded devices (e.g., dash cams). Knowledge distillation is an effective way to derive more efficient models. However, most existing works target classification tasks and treat all instances equally. In this paper, we first present our Adaptive Instance Distillation (AID) method for self-driving visual detection. It can selectively impart the teacher's knowledge to the student by re-weighing each instance and each scale for distillation based on the teacher's loss. In addition, to enable the student to effectively digest knowledge from multiple sources, we also propose a Multi-Teacher Adaptive Instance Distillation (M-AID) method. Our M-AID helps the student to learn the best knowledge from each teacher w.r.t. certain instances and scales. Unlike previous KD methods, our M-AID adjusts the distillation weights in an instance, scale, and teacher adaptive manner. Experiments on the KITTI, COCO-Traffic, and SODA10 M datasets show that our methods improve the performance of a wide variety of state-of-the-art KD methods on different detectors in self-driving scenarios. Compared to the baseline, our AID leads to an average of 2.28% and 2.98% mAP increases for single-stage and two-stage detectors, respectively. By strategically integrating knowledge from multiple teachers, our M-AID method achieves an average of 2.92% mAP improvement.**

***Index Terms*—Instance adaptive distillation, knowledge distillation, multi-teacher learning, self-driving visual perception.**

I. Introduction

In RECENT years, deep learning (DL) has revolutionized many fields, including autonomous driving perception [1], [2], [3], [4], [5], [6], [7], [8]. However, high-performance deep models usually come with large memory footprints and high computational requirements, which makes them impractical for mobile devices (e.g., dash cams). As a result, many DL-based self-driving vehicles have a full trunk of servers, which does not only require a lot of power, but also increases the response latency. Knowledge Distillation (KD) is a way to overcome such efficiency issues. It can derive a high-performance and lightweight student model by mimicking the knowledge from a powerful and sophisticated teacher model. In the past few years, many KD methods [9], [10] have been explored, and promising results have been achieved in classification problems. However, only a limited number of studies have attempted to apply KD to more challenging visual detection tasks, especially for autonomous vehicles. In object detection KD, most methods investigate what types of knowledge should be mimicked, like feature maps [11], [12], head soft prediction [13], attention-guided feature maps [14], [15], or relation between bounding boxes [16]. They usually treat all examples equally when transferring knowledge of location and category from the teacher to the student. However, due to the uneven quality and difficulty of the examples, teacher models do not learn the instances[1] equally well. Thus, the quality of knowledge provided by teachers varies with the instance. We argue that the distillation weight should adaptively change based on different instances. Sample reweighting is an effective training method in machine learning. Some studies [17], [18], [19], [20] have used hard mining to improve model performance in object detection. However, the hard mining idea in object detection has shown to be unsuitable when it comes to knowledge distillation [21]. Zhang et al. [21] add an auxiliary task branch to the student model, and the variance of the features from that extra branch, which they called data uncertainty, is utilized for reweighing image instances.

In this paper, we first present our Adaptive Instance Distillation (AID) method that reweighs distilled instances based on teacher-judged difficulty. In contrast to Zhang et al. [21], our AID does not rely on student auxiliary features' uncertainty because the variance of auxiliary features may not always represent the distillation utility of an instance, and it results in additional computation. More importantly, we argue that the importance of instances should not be determined by the feature statistics of the student network but rather by the teacher's prediction. Our AID reweighs an instance based on the teacher's original loss, which reflects the reliability of the teacher on that instance. Specifically, an instance with a larger teacher's prediction loss will receive small distillation weights and thus less attention from the student model. In other words, our AID allows the

Manuscript received 22 August 2022; revised 27 September 2022; accepted 10 October 2022. This work was supported by the National Science Foundation (NSF) under Grant 2153404. *(Corresponding author: Qing Tian.)*

The authors are with the Department of Computer Science, Bowling Green State University, Bowling Green, OH 43403 USA (e-mail: qlan@bgsu.edu; qtian@bgsu.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TIV.2022.3217261.

Digital Object Identifier 10.1109/TIV.2022.3217261

[1] In our KD scenario, 'instance' means 'image example' by default.

2379-8858 © 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

student to learn more from the teacher on instances where the teacher performs well while giving the student more freedom to learn "teacher-uncertain" instances on their own.

We also argue that multiple teachers can be more beneficial than a single teacher and that knowledge distillation should be scale-aware. Few works [22], [23] in KD have adopted the idea of adaptive reweighting in a multi-teacher framework. Both You et al. [22] and Liu et al. [23] are designed for relatively simple classification tasks and they are not scale-aware. It is not easy to apply those classification distillation methods to more challenging detection tasks because the dimensionality of the soft targets often varies with the structure of the detection heads. You et al. [22] use fixed weights to combine predictions from multiple teachers. However, fixed weights cannot adaptively distinguish high-quality teachers from low-quality teachers. Liu et al. [23] weigh the teachers based only on their intermediate features, without considering the features' quality (whether they lead to correct prediction). This label-free method can easily lead student training astray.

In the multi-teacher distillation scenario, each teacher has different judgments about an instance and a scale, and it is crucial for the student to determine which teacher's knowledge is more valuable. We point out that knowledge distillation should focus on not only what kind of knowledge to imitate, but also on which instance/scale, and from which teacher comes more valuable knowledge. Specifically, knowledge from instances/scales that a teacher can accurately predict should be identified and transferred to the student with emphasis, while the student should avoid paying too much attention to instances/scales where the teacher has no expertise.

In this paper, as an extension to AID, we propose a Multi-teacher Adaptive Instance Distillation (M-AID) framework. Our M-AID allows a student to choose the best knowledge from each teacher w.r.t. certain instances/scales. To our best knowledge, it is the first exploration of multi-teacher KD for object detection, especially in autonomous driving scenarios. Guided by our M-AID, a student can learn from a group of experts with each excelling in a particular area (a certain set of instances and scales). Thus, the risk of the student being misled by unreliable instances, scales, and teachers can be greatly reduced. Therefore, the distilled visual detectors by our methods can be more accurate and safer when deployed on automated vehicles.

In summary, the contributions of this paper can be summarized as follows:

- We present our Adaptive Instance Distillation (AID) method that allows a student to discern the reliability of the teacher's knowledge on a per-instance basis according to the teacher's performance.
- We are the first to introduce a multi-teacher distillation framework for self-driving visual detection or visual detection in general. Our M-AID guides the student to selectively learn more from more knowledgeable teachers w.r.t. an instance and a scale, rather than blindly learn from all instances and all scales equally. When all the teachers' knowledge is unreliable, the student has to rely on itself. A group of teachers with different sets of expertise are shown to benefit student learning.
- Our methods advance the state-of-the-art of visual detection in autonomous driving. Our AID has achieved 2.75%, 2.61% and 1.3% average mAP improvement on the KITTI, COCO-Traffic, and SODA10 M datasets, respectively. The proposed M-AID has improved the average mAP by 2.95% and 2.91% on KITTI and COCO-Traffic, respectively.

It is worth noting that this journal paper extends our previous work [24] significantly by: (1) proposing a new multi-teacher adaptive instance distillation (M-AID) framework that allows the student model to selectively absorb knowledge from multiple sources. (2) Our M-AID is adaptive to instance, scale, and teacher. It can significantly improve the distillation performance of our AID and other competing approaches for visual detection in self-driving vehicles. (3) We have experimented with more competing approaches (including [11], [12], [13], [15], [16], [21], [25], [26]), new backbones (ResNet-18, and MobileNetV2 [27]), and a new intelligent-vehicle-related dataset (SODA10M [28]). (4) we have also visualized different models' attention saliency maps for better understanding their differences. (5) Finally, we point out or address some typos in [24]: 1. In [24]'s Fig. 3 caption, the first prediction result is by the student baseline model (as indicated in the text), not the teacher baseline model; 2. "teacher confidence" is loosely defined; 3. In the discussion of focal loss [17] in the related work section of [24], "low cross-entropy," not "high cross-entropy," corresponds to "easy" samples (e.g., most backgrounds); 4. We have also fixed some reference and expression issues of [24].

II. Related Works

This section reviews the most relevant works in the areas of Visual Object Detection, Adaptive Sample Weighting, and Knowledge Distillation.

A. Visual Object Detection

Efficient visual detection is critical to self-driving perception, which lays the foundation for self-driving planning, control, and coordination. In fact, some intelligent car makers like Tesla even commit to the pure vision approach for their autopilot products [29], [30]. Compared to traditional object detectors like [31], [32], detectors based on deep convolutional networks have received more and more attention. Deep object detection models can be categorized into two-stage [33], [34], [35], anchor-based one-stage [17], [20], [36], [37], and anchor-free one-stage [38], [39], [40], [41] detectors. The two-stage detectors employ a region proposal network (RPN) to generate a set of proposals for potential foreground objects and then classify and localize the selected proposed regions for final prediction. In contrast, one-stage detectors perform classification and localization directly without proposals for regions of interest. They can achieve high efficiency compared to two-stage detectors. Anchor-based one-stage detectors need to traverse a large number of anchor boxes to find possible matches for the ground truth objects, adding to the computational burden. The anchor-free detectors [38], [39], [40], [41] directly predict an object's center point or key-points from feature maps, which reduces the computational cost and achieves promising results

compared to anchor-based one-stage detectors. Although different detectors may have various detection heads and losses, most state-of-the-art detectors adopt the well-known FPN idea [42] or its variants like [43] to improve the detection ability on different scales. Deep detectors normally come with high computational and storage costs, which constrain their wide deployment on intelligent vehicles.

B. Adaptive Sample Weighting

Adaptive sample (e.g., bounding box, image,...) weighting by adjusting the contributions of each sample can help with effective learning in object detection. "Hard mining" is one reweighting technique that puts non-uniform attention to samples based on difficulty. In object detection, hard-mining plays a critical role in improving detection performance [17], [18], [19], [20]. It helps reduce the relative weight of simple samples (e.g., most background bounding boxes) and gives more attention to difficult ones. However, the idea of hard mining is proved to be less effective in knowledge distillation [21]. In contrast, down-weighting hard samples or paying more attention to easy ones leads to better performance distilled models. One important question to ask is: how should the sample difficulty/importance be measured? In object detection, Lin et al. [17] use modified cross-entropy loss (a.k.a. focal loss) to measure the difficulty of bounding boxes. Bounding boxes with high prediction probability of the correct class (e.g., most backgrounds) are considered to be easy and they receive even less attention compared to the unmodified cross entropy case. GHM-C in [18] follows a similar idea to focal loss. Cao et al. [19] use Hierarchical Local Ranks to compute image sample importance in mini-batches. In knowledge distillation, Zhang et al. [21] measure image sample importance through feature variance of an auxiliary branch added to the student model. As in [21], our method applies instance reweighting during knowledge distillation on the image level. However, unlike previous approaches, we utilize the teacher network's prediction losses to determine instance importance for the student.

C. Knowledge Distillation

Knowledge distillation (KD) was introduced by Hinton et al. [9]. The goal of KD is to train a high-performance light-weight student model by transferring a powerful teacher model's knowledge. It can help meet the high accuracy and low complexity requirements of autonomous driving vehicles. The type of distilled knowledge can be categorized into three different forms: feature-based [11], [12], [14], [15], [44], [45], [46], [47], [48], response-based [9], [13] and relation-based [16], [49], [50]. The main difference lies in the kind of knowledge transferred. Unlike distillation for classification, knowledge distillation for object detection is a more challenging task. As a result, KD is less explored in object detection than in classification tasks.

It was not until 2017 that Chen et al. [51] first proposed their KD method for object detection. To deal with the high foreground-background imbalance in object detection, Chen et al. [51] down-weight the background distillation loss in the classification head.

Nguyen et al. [25] propose a label assignment distillation method, where the teacher's encoded labels are used to train a student. However, this KD method is only applied to the Probabilistic Anchor Assignment (PAA) [26] detector and is hard to generalize to other detectors. Hao et al. [52] introduce an auxiliary network to estimate the label assignment function to supervise intermediate layer training. However, the auxiliary network requires extra computation, and the distillation performance highly depends on the design of the auxiliary network. Zhang et al. [14] propose to utilize an attention-guided method to improve the distillation results. Wang et al. [11] consider only imitating the features near ground truth boxes. Yang et al. [15] treat all background features as noise and only focus attention on the foregrounds. In contrast, Guo et al. [12] decouple foreground and background features and distill them using different weights. Dai et al. [16] locate distinctive areas for distillation through finding places where the prediction gap between the teacher and the student is large. Those feature-based KD methods [11], [12], [14], [15] assign different weights to target pixels based on whether they belong to the foreground or the background. Nevertheless, the foreground-and-background assignment is meticulously determined in a subjective manner with the help of the ground truth. It follows that some informative areas could potentially be ignored and some less important locations could receive too much attention. Zheng et al. [13] develop the idea of generalized focus detectors [53] to enable students to mimic the teacher's localization soft-logits knowledge to improve their performance. However, it does not consider the different quality of teacher's prediction like our approach, and it can only be applied to the single-stage detectors with GFL [53]. All the above works [11], [12], [13], [14], [15], [16], [25], [52] have a common issue: they do not consider the teachers' prediction ability on different instances. It follows that when the teacher is wrong about certain instances, those distillation methods can no longer provide valuable knowledge to the student. Or even worse, they can mislead the student learning. Deng et al. [54] apply KD to video-based object detection. Zeni et al. [55] is focused on weakly supervised object detection. Both are interesting directions, but they are beyond the scope of this paper.

To the best of our knowledge, there is only one work [21] that attempts to apply the idea of instance-based reweighting to the domain of distillation. They add an auxiliary task branch to the student model and utilize the variance of its feature maps to measure the importance of a sample. They give larger weights to samples with low variances. However, there is no enough justification why auxiliary feature variance and sample importance are related. In contrast to Zhang et al. [21] that uses the student network's information to measure instance weight, we leverage the teacher's prediction for each instance to calculate the reliability of the distilled knowledge.

Most detection KD works follow a one-to-one distillation paradigm. Multi-teacher distillation methods have been applied in a limited number of studies for classification tasks [22], [23], [56], [57], [58]. You et al. [22] and Fukuda et al. [57] assign a uniform or fixed weight to each teacher and each instance, which cannot adaptively differentiate teachers and instances.

Liu et al. [23] use a latent factor to represent a teacher's intermediate features to measure their importance. However, this label-free method can mislead the student training when teachers produce a wrong prediction. Du et al. [58] use multi-objective optimization in the gradient space to derive teacher importance weights. However, their weighting method does not consider a teacher's prediction performance on a certain instance. As a result, the student can be misled by low-quality teacher prediction. Some studies [59], [60], [61], [62] try to let different students learn from each other to derive a high-performance model by ensemble methods. They are orthogonal/complimentary to the multi-teacher paradigm and are beyond the scope of this paper.

What's more, all the previous methods [22], [23], [56], [57], [58], [59], [60], [61], [62] are focused on classification tasks. Things become more complicated when it comes to knowledge distillation for object detection. For example, we must take into account the head architecture variations across different teachers and students. Also, we need to consider the different foreground-background assignment, bounding box regression, and classification methods between the teachers and the student. In this paper, targeting visual detection in autonomous driving, we propose multi-teacher adaptive instance distillation (M-AID) method to guide the student to learn more from more knowledgeable teachers on more useful scales and instances. To the best of our knowledge, our M-AID is the first multi-teacher distillation framework for object detection, especially in the autonomous driving domain. Our M-AID evaluates the quality of different teachers' knowledge based on their predictions and helps the student to learn more valuable knowledge across different scales and instances.

III. Methodology

Object detection involves multiple tasks, e.g., bounding box regression, category classification, and objectness prediction. Therefore, knowledge distillation for object detection is more complex than for classification. To deal with the imbalance problem between the foreground and background, many adaptive weighting strategies, such as hard mining [17], have been proposed. However, Zhang et al. [21] show that hard image instance mining does not work well in knowledge distillation. Instead, they use an auxiliary task branch to estimate uncertainty in the data and make students pay more attention to the 'stable' samples. However, the variety of the auxiliary features is not necessarily a reliable indicator for image instance importance, and it does not reflect the importance of the knowledge from the teacher. In contrast to their approach, we propose to measure the value of the teacher's knowledge on a per-instance basis by calculating the gap between the ground truth and the teacher's prediction. In other words, if the teacher model cannot predict an example well, it implies that the teacher's knowledge about that instance is less trustworthy. On the other hand, valuable knowledge comes from those instances that can be accurately predicted by the teacher model. The student network should pay more attention to such instances. In addition, we propose to employ multiple teachers to allow students to selectively absorb knowledge from multiple sources. The two approaches (AID and M-AID) will be detailed in the following two subsections.

A. Adaptive Instance Knowledge Distillation

In general, knowledge distillation tasks have two kinds of losses. One is the distillation loss $\mathcal{L}_{distill}$ which measures the knowledge (or prediction) difference between the student and the teacher model. The other one is the task loss, which is used to guide the student to learn the original task. In this paper, we first present our Adaptive Instance distillation (AID) to adaptively distill the knowledge of the teacher model on a per-instance basis for object detection tasks. The idea is that the student model should pay more attention to instances in which the teacher has more authority/trustworthiness rather than learn all instances equally from the teacher model. Fig. 1 illustrates how our AID guides the student model to better learn the most valuable and reliable knowledge from the teacher.

We define the overall loss for student learning as:

$$\mathcal{L}_i^{\mathcal{S}} = \mathcal{L}_{task,i}^{\mathcal{S}} + \lambda \mathcal{L}_{AID,i}^{\mathcal{S},\mathcal{T}}. \tag{1}$$

where i indicates the i-th instance. The superscripts $\mathcal{S}$ and $\mathcal{T}$ imply that a corresponding loss term depends on the student and/or the teacher prediction. λ is a weighting factor balancing the contribution between the task loss $\mathcal{L}_{task}$ and our instance adaptive distillation loss $\mathcal{L}_{AID}$. The latter is defined as follows:

$$\mathcal{L}_{AID,i}^{\mathcal{S},\mathcal{T}} = \exp^{-\alpha \mathcal{D}_i^{\mathcal{T}}} \mathcal{L}_{distill,i}^{\mathcal{S},\mathcal{T}}, \tag{2}$$

where

$$\mathcal{D}_i^{\mathcal{T}} = \mathcal{L}_{task,i}^{\mathcal{T}} \tag{3}$$

is the teacher's object detection task loss, i.e., the distance between the ground truth and the prediction, on the i-th instance. α is a hyper-parameter that needs to be tuned empirically (we set it to 0.1 in all our experiments). As we can see from Eq. (2), the adaptive weight of the instance i has a negative exponential correlation with the teacher's prediction loss. The larger the teacher's error on a certain instance i (i.e., $\mathcal{D}_i^{\mathcal{T}}$) is, the smaller weight or less attention the instance i will receive from the student model during the knowledge distillation process. On the other hand, instances where the teacher predicts accurately (i.e., with smaller $\mathcal{D}_i^{\mathcal{T}}$ values) deserve more of the student's attention in the knowledge transfer process. The instance weight degrades exponentially with the increase of the teacher's prediction error. The exponential function sets an appropriate range of the punishment. Take the extreme cases for example. An instance where the teacher's loss is extremely large will receive approximately zero attention while there will be no knowledge transfer degradation for instances where the teacher model makes 'perfect' prediction (zero task loss).

Putting all things together, the final loss for our instance-adaptive student learning will be:

$$\mathcal{L}_i^{\mathcal{S}} = \mathcal{L}_{task,i}^{\mathcal{S}} + \lambda \exp^{-\alpha \mathcal{L}_{task,i}^{\mathcal{T}}} \mathcal{L}_{distill,i}^{\mathcal{S},\mathcal{T}}. \tag{4}$$

It is worth noting that Feature Pyramid Networks (FPN) [42] or its variants have been widely adopted by state-of-the-art object detectors. To improve knowledge transfer for objects of different scales, we can apply our AID strategy to each output layer of the FPN (Fig. 1). In this case, our AID adaptively weighs not only the instance-wise knowledge but also scale-wise feature knowledge during the knowledge distillation process.

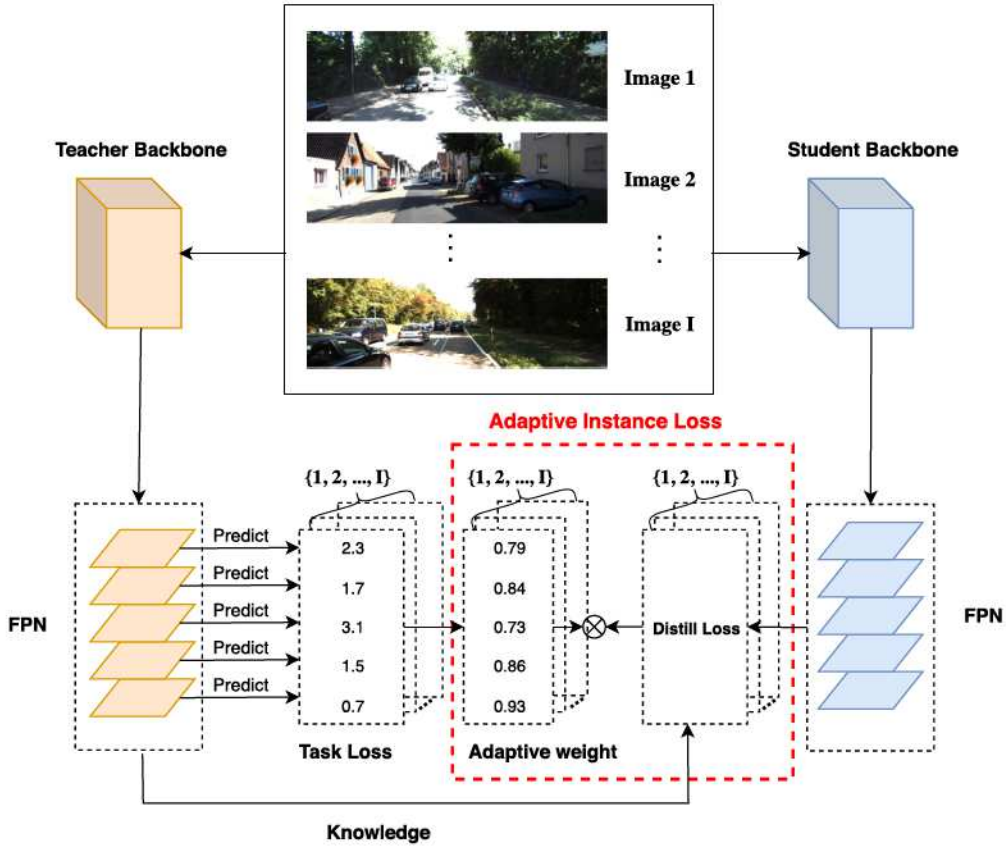

Fig. 1. Illustration of the proposed adaptive instance distillation (AID) method. The teacher losses associated with different instances and scales will be transformed into weights to guide the knowledge distillation process. The transformation is based on (2). I is the total number of images.

The student will rely more on the teacher for scales where the teacher feels more confident.[2] For scales where the teacher performs bad, the student will rely more on itself to learn instead of being misled by the teacher. Such scale-adaptive knowledge distillation contributes to better object detection on different scales. In the case of autonomous driving, a car can better detect road objects of different sizes and distances. More details will follow in the experiment section.

B. Multi-Teacher Adaptive Instance Distillation

Most successful knowledge distillation methods are based on the one-to-one framework, where one teacher teaches one student. As an old Chinese saying goes: "In a party of three, there must be one whom I can learn from." Some multi-teacher knowledge distillation methods [22], [56], [58] have been proposed and proven to be beneficial for classification tasks. They combine predictions from multiple teachers with fixed weight assignments or with gradient weighting schemes [58]. However, fixed weights cannot adaptively distinguish high-quality teachers from low-quality teachers, while the gradient-based weighting method [58] may easily mislead the student by low-quality teachers. In this paper, we propose Multi-teacher Adaptive Instance Distillation (M-AID) method to assign different weights to different teachers in a dynamic manner. By combining the strategy with our instance-and-scale-aware AID, we can adaptively select valuable knowledge for the student across different instances, scales, and teachers.

Fig. 2 illustrates how M-AID works. The two main types of losses for our multi-teacher framework are as follows:

$$\mathcal{L}_i^{\mathcal{S}} = \mathcal{L}_{task,i}^{\mathcal{S}} + \lambda \sum_{k=1}^{K} w_{k,i} \mathcal{L}_{AID,k,i}^{\mathcal{S},\mathcal{T}}, \tag{5}$$

where $\mathcal{L}_{task,i}^{\mathcal{S}}$ is the original detection task loss of the student, and k in $\mathcal{L}_{AID,k,i}$ stands for the k-th teacher. One difference between Eq. (1) and Eq. (5) is that adaptive distillation loss $\mathcal{L}_{AID,k,i}$ is weighted by $w_{k,i}$, which is defined as:

$$w_{k,i} = \frac{\exp^{-\mathcal{D}_{k,i}^{\mathcal{T}}}}{\sum_{k=1}^{K} \exp^{-\mathcal{D}_{k,i}^{\mathcal{T}}}}, \tag{6}$$

where $\mathcal{D}_{k,i}^{\mathcal{T}}$ is the k-th teacher's object detection task loss, i.e., the distance between the ground truth and the prediction, on the i-th instance. The purpose of the denominator in Eq. (6) is to normalize the distillation losses of multiple teachers.

[2]here, confident is loosely defined as knowledgeable

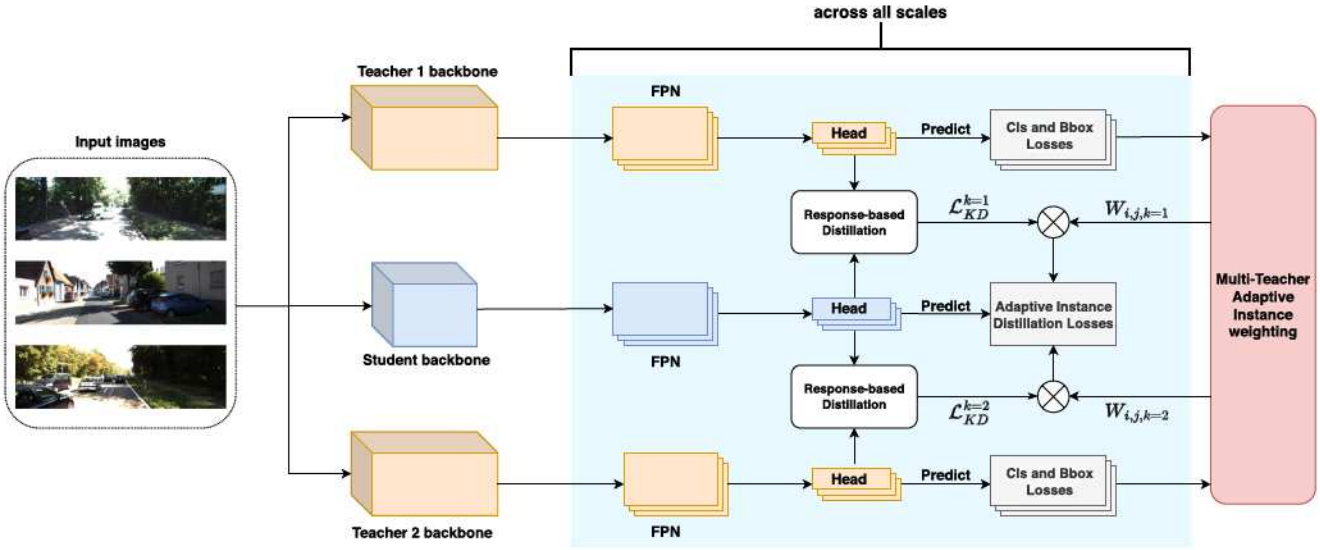

Fig. 2. Illustration of the proposed multi-teacher adaptive instance distillation (M-AID) method. The losses associated with the kth teacher's prediction for the jth scale of the ith instance will be transformed into weights $W_{i,j,k}$ to reweight distillation losses, which directs more student attention to more valuable knowledge across different instances, scales, and teachers. Without loss of generality, only two teachers are shown in this figure.

Putting all things together, we define the loss of our M-AID distilled student as:

$$\mathcal{L}_i^{\mathcal{S}} = \mathcal{L}_{task,i}^{\mathcal{S}} + \lambda \frac{\exp^{-\mathcal{L}_{task,k,i}^{\mathcal{T}}}}{\sum_K \exp^{-\mathcal{L}_{task,k,i}^{\mathcal{T}}}} \exp^{-\alpha \mathcal{L}_{task,k,i}^{\mathcal{T}}} \mathcal{L}_{distill,k,i}^{\mathcal{S},\mathcal{T}}. \tag{7}$$

The $task$ in $\mathcal{L}_{task,k,i}$ includes category classification and bounding box regression. Re-weighting according to the two subtask losses are conducted separately in our M-AID.

IV. Experimental Setup and Results

A. Datasets

To evaluate our methods, we utilize three autonomous driving related datasets in our experiments.

KITTI [63] is a 2D-object detection dataset that includes seven different types of road objects. As suggested in [64], we group similar categories into one. Specifically, we perform the following modification to the original KITTI dataset:

- Car ← *car, van, truck, tram*
- Pedestrian ← *pedestrian, person*
- Cyclist ← *cyclist*

It includes 7481 images with annotations. We split it into a training set and a validation set in the ratio of 8:2.

COCO-Traffic is a dataset containing 13 traffic-related categories. This dataset is obtained by selecting categories related to self-driving from MS COCO 2017 [65]. The COCO-Traffic dataset includes the following categories:

- **Road-related:** *bicycle, car, motorcycle, bus, train, truck, traffic light, fire hydrant, stop sign, parking meter*
- **Others:** *person, cat, dog*

Unlike [24], we keep only images containing at least one road-related object to filter out those images that only contain indoor objects. The selection is applied to both the training and validation sets.

SODA10 M [28] is a recent large-scale 2D dataset, which contains 10 M unlabeled images and 20 k labeled images from 6 object categories (i.e., *Pedestrian, Cyclist, Car, Truck, Tram*, and *Tricycle*). At the time of writing, SODA10 M is the largest public autonomous driving dataset that can be used for 2D visual detection.

B. Implementation Detail

1) Adaptive Instance Distillation: In our AID experiments, we chose Faster-RCNN [33] as an example of two-stage detectors, and selected Generalized Focal Loss (GFL) [53] and Probabilistic anchor assignment (PAA) [26] as examples of single-stage detectors. All teachers have a ResNet101 [66] backbone. We experimented with three different student backbone architectures (i.e., ResNet-50, ResNet-18, and MobileNetsV2). We re-implement the following state-of-the-art KD methods [11], [12], [13], [14], [15], [16], [21], [25] to compare with our AID:

- Attention-Guided by Zhang et al. [14] (ICLR'21)
- GI-imitation by Dai et al. [16] (CVPR'21)
- DeFeat by Guo et al. [12] (CVPR'21)
- FGD by Yang et al. [15] (CVPR'22)
- LD by Zheng et al. [13] (CVPR'22)
- Fine-Grained by Wang et al. [11] (CVPR'19)
- PAD by Zhang et al. [21] (ECCV'20)
- LAD by Nguyen et al. [25] (WACV'22)

For a fair comparison, all re-implemented KD methods and our AID are imposed on multi-level FPN (P3-P7). In our experiments, the competing instance adaptive KD method, PAD [21], is applied on top of Attention-Guided [14]. In the implementation of our AID, we use the sum of the teacher's classification losses and bounding box losses to re-weigh the KD losses. The teacher and student baseline models (without any KD) were directly trained with MMDetection [67]'s default configuration.

2) Multi-Teacher Adaptive Instance Distillation: In our M-AID experiments, we take Zheng et al. [13]'s response-based

TABLE I
PERFORMANCE (MAP) OF DIFFERENT DISTILLATION METHODS WITH GFL DETECTOR [53] ON THE KITTI AND COCO TRAFFIC DATASETS

Student backbones / KD methods	ResNet-50		ResNet-18	
	KITTI	COCO Traffic	KITTI	COCO Traffic
Teacher (w ResNet-101)	89.4	71.8	89.4	71.8
Student-baseline	85.1	67.7	81.9	61.9
PAD-attention-Guided [21]	84.7	63.6	80.8	60.7
Attention-Guided [14]	86.4	69.5	84.4	62.6
Attention-Guided + AID	**88.0**	**70.1**	**84.7**	**64.1**
GI-imitation [16]	86.1	69.3	84.6	63.7
GI-imitation + AID	**87.9**	**69.6**	**85.2**	**64.6**
DeFeat [12]	85.4	69.3	83.3	62.7
DeFeat + AID	**86.4**	**69.5**	**84.7**	**63.8**
LD [13]	85.5	67.8	83.6	62.7
LD + AID	**87.2**	**68.4**	**83.8**	**64.4**
FGD [15]	89.2	71.0	86.7	65.9
FGD + AID	**89.9**	**71.1**	**87.5**	**66.4**
Fine-Grained [11]	84.4	68.6	82.6	62.4
Fine-Grained + AID	**86.6**	**69.1**	**84.4**	**62.9**

Note: The teacher model and the student-baseline are non-distillation GFL models with ResNet-101 and ResNet-50/18 as backbones, respectively.

approach (LD) as a KD baseline, upon which we apply our instance, scale, and teacher adaptive knowledge reweighting methods. All the student models use ResNet-50 or ResNet-18 as the backbone and GFL [53] as the head. For those teacher models that do not have GFL [53], we implement GFL in their heads to ensure the feasibility of multi-teacher distillation.[3] Since the two distillation losses (one for classification and one for bounding box regression) are separated in LD [13], we apply AID and M-AID to the two losses separately.

3) Hyperparameters: All the detection experiments are conducted in the MMDetection framework [67] using Pytorch [68]. We do not perform much hyperparameter tweaking. In our re-implementation of the state-of-the-art KD methods [11], [12], [14], [15], [16], [25], we adopt the same hyperparameter values as provided by their authors. As for LD [13], we set both classification distillation and localization distillation weights to 0.05 and make slight adjustments based on different detectors and datasets. Furthermore, we set $\alpha = 0.1$ in Eq. (2) for our AID and M-AID throughout all experiments. All models are sufficiently trained to convergence (i.e., 24 epochs for models with ResNet-101 backbone, 12 epochs for models with ResNet-50 backbone, ResNet-18, and MobileNetV2).

We verify the effectiveness of our proposed AID and M-AID on the autonomous-driving-related KITTI, COCO-Traffic and SODA10 M datasets. We present our AID's and M-AID's results in Sections IV-C and IV-D, respectively. Also, we will show the intuitive differences in CAM visualization of our approach from other baselines. All models are evaluated in terms of mean averaged precision (mAP) with 0.5 as the Intersection over Union (IoU) threshold.

C. Adaptive Instance Distillation (AID) Results

1) Quantitative Analysis: In this section, we report our AID's quantitative performance on three state-of-the-art detectors, including double-stage Faster RCNN, single-stage GFL [53] and PAA [26].

[3]This ensures the teacher models' localization heads to have same-dimension generalized logits.

We first compare our proposed AID with several state-of-the-art KD methods [11], [12], [13], [14], [15], [16], [21] using the GFL detector. The results are reported in Table I. All teachers have a ResNet-101 backbone, and we test two students (i.e., one with a ResNet-50 backbone and the other with a ResNet-18 backbone). As can be seen from Table I, by applying our AID method, we can achieve consistent improvement over the competing KD baselines on the KITTI and COCO-Traffic datasets for both ResNet-50 and ResNet-18 backbones. In particular, with a ResNet-50-backbone student on the KITTI dataset, our AID achieves 2.2% mAP improvement over the Fine-Grained [11] baseline (bottom two rows). Also in the ResNet-50-KITTI case, FGD + AID (third-to-last row) even beats the larger teacher model (with a ResNet-101 backbone). The main reason is that our AID gives the student more freedom to rely on itself to learn when the teacher provides untrustworthy prediction on certain instances/scales. We can also observe a general trend that the improvement brought about by our AID is larger on the smaller ResNet-18 backbone than on ResNet-50. On average, ResNet-50-based students gain 2.57% mAP and 1.93% mAP on KITTI and COCO-Traffic, respectively. Students with ResNet-18 backbones gain an average of 3.15% and 2.47% mAP on the two datasets.

Table II shows that our AID outperforms state-of-the-art KD baselines on the Faster-RCNN detector [33] as well. On average, our AID improves student performance by 2.53% mAP and 3.43% mAP on the KITTI and COCO-Traffic datasets, respectively.

Table III demonstrates the results with the PAA detector [26] on another autonomous driving dataset - SODA10M [28]. It can be observed that our AID results in a 1.3% mAP improvements with the more compact MobileNetV2 as backbone.

2) Qualitative Analysis: Fig. 3 shows a random example on the KITTI dataset. The qualitative results of three GFL models are demonstrated. They are (from top to bottom): 1) teacher GFL model, 2) Zhang et al. [14]'s Attention-Guided model with a ResNet-50 backbone, and 3) our AID-distilled model with a ResNet-50 backbone. For the readers' convenience, we highlight

TABLE II
PERFORMANCE (MAP) OF DIFFERENT DISTILLATION METHODS WITH FASTER R-CNN DETECTOR [33] ON THE KITTI AND COCO TRAFFIC DATASETS

KD methods \ Student backbones	ResNet-50		ResNet-18	
	KITTI	COCO Traffic	KITTI	COCO Traffic
Teacher (w ResNet-101)	89.3	67.9	89.3	67.9
Student-baseline	88.9	67.5	84.1	63.1
PAD-attention-Guided [21]	88.9	67.6	86.4	68.2
Attention-Guided [14]	89.0	67.8	87.2	65.3
Attention-Guided + AID	**89.6**	**69.0**	**88.4**	**68.4**
FGD [15]	88.9	67.7	87.0	64.1
FGD + AID	**89.5**	**70.1**	**88.6**	**67.4**

Note: The teacher model and the student-baseline are non-distillation Faster-RCNN models with ResNet-101 and ResNet-50/18 backbones, respectively.

Fig. 3. Qualitative Analysis on KITTI – From top to bottom, the prediction results are respectively from 1) Teacher baseline model, 2) Zhang et al. [14]'s KD student baseline model, and 3) our AID distilled student model. We have cropped and zoomed in on portions where the models disagree most. The zoomed-in views are alongside each image and unlike the original view, they do not contain category labels and confidence scores for clarity. Best viewed in color and zoomed in.

TABLE III
PERFORMANCE (MAP) OF LAD [25] WITH PAA DETECTOR [26] ON THE SODA10 M DATASETS

KD methods \ Student backbones	MobileNetV2
Teacher (w ResNet-101)	55.2
Student-baseline	49.7
LAD [25]	50.1
LAD + AID	**51.0**

Note: The teacher model and the student-baseline are non-distillation PAA [26] models with ResNet-101 and MobileNetV2 as backbones, respectively.

the prediction differences between our AID-based model and the other baseline models using cyan boxes and ovals. According to Fig. 3, generally speaking, our AID-distilled model has better detection capability for overlapping objects and small-scale objects. For example, in the top image, the teacher baseline model generates a bunch of approximate bounding boxes in order to locate the pedestrian and the car that overlap each other in the right part of the image. Although Zhang et al. [14]'s distilled model improves the detection a bit (the middle image), it still struggles to find the correct bounding boxes for the overlapping objects. On the other hand, the bounding boxes generated by our AID-distilled model are more precise. An example showing our AID's superiority in detecting small-scale objects can be found on the left part of the image. Both the teacher model and Zhang et al. [14]'s Attention-Guided model fail to detect the small-scale car behind the pole, while our AID-distilled model can detect the car without any problem.

Fig. 4 demonstrates another random example on the COCO-Traffic dataset. From left to right, the results are respectively from 1) the Teacher GFL baseline model, 2) Zhang et al. [14]'s distilled GFL model, and 3) our AID-distilled GFL model. Both

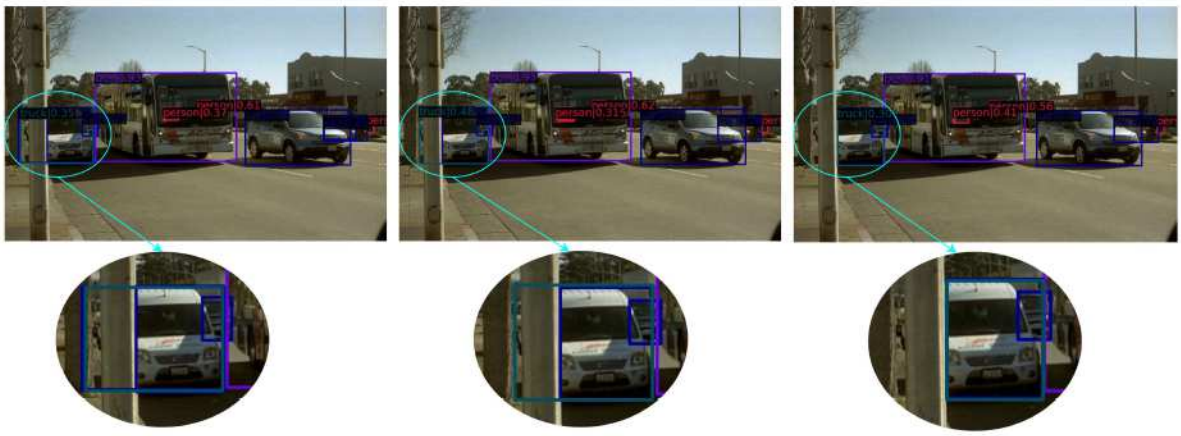

Fig. 4. Qualitative Analysis on COCO traffic – From left to right, the prediction results are respectively from 1) Teacher baseline model, 2) Zhang et al. [14]'s KD student baseline model, and 3) our AID distilled model. We have cropped and zoomed in on portions where the models disagree most. The zoomed-in views are under each image and unlike the original view, they do not contain category labels and confidence scores for clarity. Best viewed in color and zoomed in.

student models use ResNet-50 as the backbone. The left two images show that 1) the teacher and 2) Zhang et al. [14]'s model inaccurately predict the truck in the marked cyan oval boxes. From the zoomed-in view, we can see that both 1) and 2) incorrectly generate two bounding boxes in the truck region (each with a different category, dark blue: car, light blue: truck). The reason for the detection issue is that the object is occluded and the teacher model cannot impart trustworthy information to the student model in such scenarios. Zhang et al. [14]'s distilled model blindly trusts the teacher's prediction and thus makes a similar mistake. In contrast, our AID-distilled model relies more on itself when learned to predict for such instances. Thus, only our model provides the right number of bounding box, of the right category, and at a precise location (rightmost picture).

D. Multi-Teacher Adaptive Distillation (M-AID) Results

1) Quantitative Analysis: We employed five teacher detectors of different types to perform M-AID:

- Generalized Focal Loss (GFL) [53]
- RetinaNet [17]
- Adaptive training sample selection (Atss) [69]
- fully convolutional one-stage (Fcos) [39]
- GFL [53] with DCONV [70] added

Our M-AID experiments were conducted on the KITTI and COCO-Traffic datasets. We experimented with two GFL student models: one with a ResNet-50 backbone (named Student-R50) and the other with a ResNet-18 backbone (named Student-R18). The quantitative results on the two datasets are shown in Tables IV and V, respectively. As comparison, single-teacher student models, i.e., distilled by LD [13] or LD + AID (AID for short), are also included in the two tables. In Tables IV and V, the columns denote the teachers, except for the last two columns where the two students' mAP performance are shown. The rows are grouped by the KD method used. Each row represents a knowledge distillation procedure guided by certain teacher(s). For reference, we add the teacher models' mAP performance under their names. The use of a teacher is marked by a check mark. For example, in our multi-teacher KD case (M-AID), the two check marks in the bottom row of the two tables indicate that the two corresponding teachers (GFL and Atss) jointly guide a student model. It is worth noting that if either teacher model uses DCONV [70], the student model will also use it.

As shown in Tables IV and V, our instance, scale, and teacher aware M-AID outperforms the "distillation-free" student-baseline and the state-of-the-art KD methods (LD [13] and our AID) in a variety of teacher-student combinations on the KITTI and COCO-Traffic datasets. For instance, according to the bottom row of Table IV, we can see that the ResNet-18 based student model (Student-R18) jointly distilled by the two teachers (i.e., GFL and Atss) achieves a mAP of 86.8. This mAP score is higher than separately using either one of the two teachers (GFL or Atss) to distill the student. According to Table IV, the same student distilled by the single-teacher AID achieves 83.8 mAP (with Teacher GFL) and 86.1 mAP (with Teacher Atss). Although the AID results are worse than M-AID's, they are still better than Zheng et al.'s LD knowledge distillation results (83.6 mAP using Teacher GFL and 84.5 mAP using Teacher Atss). On average, there is a 2.93% mAP improvement on KITTI and 2.91% mAP improvement on COCO-Traffic over the student baseline by using our M-AID. Although, in most cases, M-AID distilled models can achieve higher performance than those distilled by AID, exceptions may exist when the performance gap between the teacher models is large. In such scenarios, we find that the student can be misled by the worse teacher model. For example, in Table IV, the student model trained by GFL [53] and Retina [17] performed no better than the model distilled by a single GFL teacher using AID.

2) Qualitative Analysis: To better understand the distilled models, we visualize the differences in their focus/attention using saliency maps. Specifically, we visualize the Eigen Class Activation Mapping (EigenCAM [71]) of different models' FPN neck. The results are shown in Fig. 5 .

According to Fig. 5, in image A, the "distillation-free" model focuses on the object's surroundings, but has a low attention overlap with the object. It incorrectly detects the fence as a car.

TABLE IV
RESULTS OF MULTI-TEACHER EXPERIMENTS ON THE KITTI DATASET

KD methods \ Teacher Models (w ResNet-101)	GFL [53] 89.4	GFL-dconv [70] 88.8	Retina [17] 84.8	Fcos [39] 88.3	Atss [69] 88.7	Student-R50	Student-R18
Student-baseline	✓					85.1	81.9
		✓				87.1	83.0
			✓			82.9	79.4
				✓		75.4	79.2
					✓	85.4	82.6
LD [13]	✓					85.5	83.6
		✓				85.5	84.6
			✓			83.7	82.2
				✓		76.9	81.1
					✓	86.1	84.5
AID	✓					87.2	83.8
		✓				87.1	85.1
			✓			85.4	84.4
				✓		81.2	82.0
					✓	87.3	86.1
M-AID	✓	✓				**88.0**	**85.8**
	✓		✓			**86.1**	**85.6**
	✓			✓		**86.6**	**84.8**
	✓				✓	**87.7**	**86.8**

Note: each row represents a knowledge distillation procedure guided by certain teacher(s). The use of a teacher is marked by a check mark. For our multi-teacher approach (M-AID), the two check marks in a row indicate that the two corresponding teachers jointly guide a student model. We have tested on two student models. Student-R50 stands for a student model with a ResNet-50 backbone. Student-R18 is similarly defined. Their mAP performance are appended to the table as the last two columns. For reference, we add the teacher models' mAP performance under their names. AID and M-AID are applied on top of LD. GFL-dconv stands for a GFL model with the deformable convolutional networks [70] trick added. The student-baseline is the certain detector without applying any distillation method.

TABLE V
RESULTS OF MULTI-TEACHER EXPERIMENTS ON THE COCO-TRAFFIC DATASET

KD methods \ Teacher Models (w ResNet-101)	GFL [53] 71.8	GFL-dconv [70] 73.8	Retina [17] 68.9	Fcos [39] 72.7	Atss [69] 72.7	Student-R50	Student-R18
Student-baseline	✓					67.7	61.9
		✓				72.0	63.5
			✓			66.7	59.7
				✓		59.8	59.8
					✓	67.4	62.2
LD [13]	✓					67.8	62.7
		✓				72.5	64.3
			✓			67.8	61.3
				✓		67.3	61.9
					✓	68.7	63.9
AID	✓					68.4	64.4
		✓				72.3	65.7
			✓			68.6	62.8
				✓		67.8	63.6
					✓	69.8	64.2
M-AID	✓	✓				**72.8**	**65.9**
	✓		✓			**69.5**	**63.8**
	✓			✓		**69.7**	**64.9**
	✓				✓	**70.0**	**65.1**

Note: each row represents a knowledge distillation procedure guided by certain teacher(s). The use of a teacher is marked by a check mark. For our multi-teacher approach (M-AID), the two check marks in a row indicate that the two corresponding teachers jointly guide a student model. We have tested on two student models. Student-R50 stands for a student model with a ResNet-50 backbone. Student-R18 is similarly defined. Their mAP performance are appended to the table as the last two columns. For reference, we add the teacher models' mAP performance under their names. AID and M-AID are applied on top of LD. GFL-dconv stands for a GFL model with the deformable convolutional networks [70] trick added. The student-baseline is the certain detector without applying any distillation method.

We can see that LD [13] does improve the model's attention, but still produces false detection on the fence. Our AID and M-AID both succeed in avoiding the detection mistake. However, only our M-AID distillation model pays attention to the full body of the car on the bottom left of the image, and the M-AID model achieves the highest mAP.

Similarly, from Fig. 5's image B, we can see that AID improves the model's ability to detect occluded objects. Our M-AID further optimizes the attention of the model and its ability to detect small and overlapping objects. For example, in image B, the AID and M-AID models successfully detect the partially occluded car behind the one closest to the camera. The red attention area of M-AID covers more pixels of the partially occluded car. Moreover, the M-AID model is the only one that succeeds to detect each of the three overlapping cars at the far end of image B.

E. Computational Complexity

In addition to mAP performance, we also compared different architectures' efficiency in terms of FLOPs[4] and

[4]In the DL literature, there are two FLOP versions: 1) multiply-and-add (e.g., [13]), and 2) multiply/add (e.g., [72]). We follow the former convention.

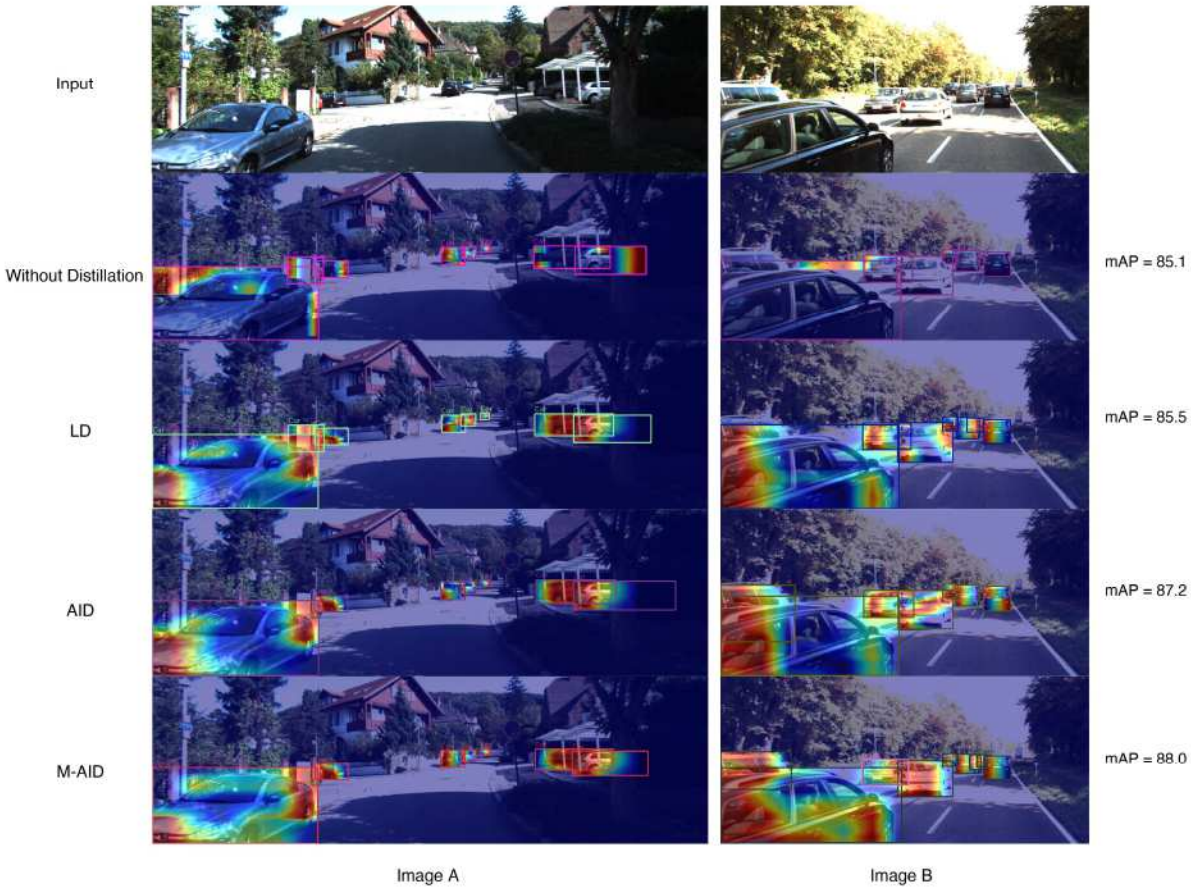

Fig. 5. Attention maps of student models distilled by the state-of-the-art LD, our AID and M-AID methods. We show the CAM saliency of the different student models' FPN neck with ResNet-50 as backbone. Different colors indicate different attention levels, with the red color representing the highest attention and the blue color representing the lowest. The dark red regions contribute most to model decision. Best viewed in color and zoomed in.

TABLE VI
MODEL COMPLEXITY (WITH 224×224 INPUT RESOLUTION)

Model	Backbones	Parames(M)	GFLOPs
GFL [53]	ResNet-101	51.03	13.79
	ResNet-50	32.04	10.05
	ResNet-18	19.09	7.61
ATSS [69]	ResNet-101	51.03	13.78
	ResNet-50	32.04	10.05
	ResNet-18	19.09	7.62
FCOS [39]	ResNet-101	55.06	13.61
	ResNet-50	36.12	9.88
	ResNet-18	19.67	7.63
Retina [17]	ResNet-101	55.35	14.04
	ResNet-50	36.15	10.09
	ResNet-18	19.66	7.60
GFL-Dconv [70]	ResNet-101	52.32	10.57
	ResNet-50	32.62	8.67
	ResNet-18	19.38	6.98
Faster R-CNN [33]	ResNet-101	60.13	27.09
	ResNet-50	41.13	23.36
	ResNet-18	28.13	20.77
PAA [26]	ResNet-101	50.89	13.63
	MobileNetV2	10.28	5.95

parameters. The results are shown in Table VI. According to the table, our distilled models with the smaller backbones (ResNet-50, ResNet-18, or MobileNetV2) are more efficient than the corresponding teacher baselines with larger ResNet-101 backbones. In addition to the previously mentioned promising mAPs, our ResNet-18/50 distillation model enjoys an average of 61.68%/35.46% reduction in the number of parameters and an average of 39.31%/22.69% savings in FLOPs. The MobileNetV2 can achieve 79.80% reduction of parameters and 56.35% saving in FLOPs.

V. CONCLUSION

In this paper, we have proposed adaptive instance distillation (AID) and multi-teacher adaptive instance distillation (M-AID) methods to derive more compact and better-performing visual detectors for self-driving vehicles. The AID method redirects more student attention to instances that the teacher model performs well on. Our M-AID empowers the student model to learn more from more knowledgeable teachers w.r.t an instance and a scale. For the first time, we guide the student detector to actively search for valuable knowledge across different instances, teachers, and scales during distillation. In our experiments, we have compared our methods with a wide array of state-of-the-art knowledge distillation baselines (e.g., [11], [12], [13], [14], [15], [16], [21], [25]) and have tested our strategies using both single-stage and double-stage detectors. Experimental results on the KITTI, COCO-Traffic, and SODA10 M datasets demonstrate our AID and M-AID methods' efficacy. On average, 2.28% and 2.98% mAP increases can be achieved by AID for the

single-stage detectors and two-stage detectors, respectively. Furthermore, our M-AID method leads to an average of 2.92% mAP improvement.

ACKNOWLEDGMENT

This work would not have been possible without the computing resources provided by the Ohio Supercomputer Center.

REFERENCES

[1] C. Chen, A. Seff, A. Kornhauser, and J. Xiao, "DeepDriving: Learning affordance for direct perception in autonomous driving," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 2722–2730.

[2] M. Bojarski et al., "End to end learning for self-driving cars," 2016, *arXiv:1604.07316*.

[3] Z. Chen and X. Huang, "End-to-end learning for lane keeping of self-driving cars," in *Proc. IEEE Intell. Veh. Symp.*, 2017, pp. 1856–1860.

[4] H. Xu, Y. Gao, F. Yu, and T. Darrell, "End-to-end learning of driving models from large-scale video datasets," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 3530–3538.

[5] Z. Rozsa and T. Sziranyi, "Object detection from a few LIDAR scanning planes," *IEEE Trans. Intell. Veh.*, vol. 4, no. 4, pp. 548–560, Dec. 2019.

[6] J. E. Hoffmann, H. G. Tosso, M. M. D. Santos, J. F. Justo, A. W. Malik, and A. U. Rahman, "Real-time adaptive object detection and tracking for autonomous vehicles," *IEEE Trans. Intell. Veh.*, vol. 6, no. 3, pp. 450–459, Sep. 2021.

[7] M. Schutera, M. Hussein, J. Abhau, R. Mikut, and M. Reischl, "Night-to-day: Online image-to-image translation for object detection within autonomous driving by night," *IEEE Trans. Intell. Veh.*, vol. 6, no. 3, pp. 480–489, Sep. 2021.

[8] J. I. Choi and Q. Tian, "Adversarial attack and defense of YOLO detectors in autonomous driving scenarios," in *Proc. IEEE Intell. Veh. Symp.*, 2022, pp. 1011–1017.

[9] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," 2015, *arXiv:1503.02531*.

[10] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," Master's thesis, Dept. Comput. Sci., Univ. Toronto, 2009.

[11] T. Wang, L. Yuan, X. Zhang, and J. Feng, "Distilling object detectors with fine-grained feature imitation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4928–4937.

[12] J. Guo et al., "Distilling object detectors via decoupled features," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 2154–2164.

[13] Z. Zheng et al., "Localization distillation for dense object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 9397–9406.

[14] L. Zhang and K. Ma, "Improve object detection with feature-based knowledge distillation: Towards accurate and efficient detectors," in *Proc. Int. Conf. Learn. Representations*, 2021. [Online]. Available: https://openreview.net/forum?id=uKhGRvM8QNH

[15] Z. Yang et al., "Focal and global knowledge distillation for detectors," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 4633–4642.

[16] X. Dai et al., "General instance distillation for object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 7838–7847.

[17] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2999–3007.

[18] B. Li, Y. Liu, and X. Wang, "Gradient harmonized single-stage detector," in *Proc. AAAI Conf. Artif. Intell.*, 2019, pp. 8577–8584.

[19] Y. Cao, K. Chen, C. C. Loy, and D. Lin, "Prime sample attention in object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 11580–11588.

[20] W. Liu et al., "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 21–37.

[21] Y. Zhang et al., "Prime-aware adaptive distillation," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 658–674.

[22] S. You, C. Xu, C. Xu, and D. Tao, "Learning from multiple teacher networks," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2017, pp. 1285–1294.

[23] Y. Liu, W. Zhang, and J. Wang, "Adaptive multi-teacher multi-level knowledge distillation," *Neurocomputing*, vol. 415, pp. 106–113, 2020.

[24] Q. Lan and Q. Tian, "Adaptive instance distillation for object detection in autonomous driving," in *Proc. Int. Conf. Pattern Recognit.*, 2022.

[25] C. H. Nguyen, T. C. Nguyen, T. N. Tang, and N. L. Phan, "Improving object detection by label assignment distillation," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, 2022, pp. 1322–1331.

[26] K. Kim and H. S. Lee, "Probabilistic anchor assignment with IoU prediction for object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 355–371.

[27] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4510–4520.

[28] J. Han et al., "SODA10M: A large-scale 2D self/semi-supervised object detection dataset for autonomous driving," 2021, *arXiv:2106.11118*.

[29] Transitioning to tesla vision. Accessed: Aug. 19, 2022. [Online]. Available: https://www.tesla.com/support/transitioning-tesla-vision

[30] Tesla commits to pure vision approach, removes radar from model S and model X. Accessed: Feb. 25, 2022. [Online]. Available: https://www.teslarati.com/tesla-commits-pure-vision-approach-models-model-x-no-radar/

[31] R. Vaillant, C. Monrocq, and Y. L. Cun, "An original approach for the localization of objects in images," in *Proc. 3rd Int. Conf. Artif. Neural Netw.*, 1993, pp. 26–30.

[32] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2001, pp. I–I.

[33] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2015, pp. 91–99.

[34] Z. Cai and N. Vasconcelos, "Cascade R-CNN: Delving into high quality object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6154–6162.

[35] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2980–2988.

[36] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6517–6525.

[37] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.

[38] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "YOLOX: Exceeding YOLO series in 2021," 2021, *arXiv:2107.08430*.

[39] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully convolutional one-stage object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 9626–9635.

[40] Z. Yang, S. Liu, H. Hu, L. Wang, and S. Lin, "RepPoints: Point set representation for object detection," in *Proc. Int. Conf. Comput. Vis.*, 2019, pp. 9656–9665. [Online]. Available: https://www.microsoft.com/en-us/research/publication/reppoints-point-set-representation-for-object-detection/

[41] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, and Q. Tian, "CenterNet: Keypoint triplets for object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 6568–6577.

[42] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 936–944.

[43] S. Qiao, L.-C. Chen, and A. Yuille, "DetectoRS: Detecting objects with recursive feature pyramid and switchable atrous convolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 10208–10219.

[44] A. Romero et al., "FitNets: Hints for thin deep nets," in *Proc. Int. Conf. Learn. Representations*, 2015.

[45] Z. Huang and N. Wang, "Like what you like: Knowledge distill via neuron selectivity transfer," 2017, *arXiv:1707.01219*.

[46] S. Ahn, S. X. Hu, A. Damianou, N. D. Lawrence, and Z. Dai, "Variational information distillation for knowledge transfer," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 9155–9163.

[47] B. Heo, M. Lee, S. Yun, and J. Y. Choi, "Knowledge transfer via distillation of activation boundaries formed by hidden neurons," in *Proc. AAAI Conf. Artif. Intell.*, 2019, pp. 3779–3787.

[48] J. Yim, D. Joo, J. Bae, and J. Kim, "A gift from knowledge distillation: Fast optimization, network minimization and transfer learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 7130–7138.

[49] Y. Liu et al., "Knowledge distillation via instance relationship graph," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 7089–7097.
[50] W. Park, D. Kim, Y. Lu, and M. Cho, "Relational knowledge distillation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3962–3971.
[51] G. Chen, W. Choi, X. Yu, T. Han, and M. Chandraker, "Learning efficient object detection models with knowledge distillation," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 742–751.
[52] M. Hao, Y. Liu, X. Zhang, and J. Sun, "LabelEnc: A new intermediate supervision method for object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 529–545.
[53] X. Li et al., "Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2020, pp. 21002–21012.
[54] J. Deng, Y. Pan, T. Yao, W. Zhou, H. Li, and T. Mei, "Relation distillation networks for video object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 7022–7031.
[55] L. F. Zeni and C. R. Jung, "Distilling knowledge from refinement in multiple instance detection networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops*, 2020, pp. 3324–3333.
[56] H. Zhang, D. Chen, and C. Wang, "Confidence-aware multi-teacher knowledge distillation," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2022, pp. 4498–4502.
[57] T. Fukuda, M. Suzuki, G. Kurata, S. Thomas, J. Cui, and B. Ramabhadran, "Efficient knowledge distillation from an ensemble of teachers," in *Proc. 18th Annu. Conf. Int. Speech Commun. Assoc.*, 2017, pp. 3697–3701.
[58] S. Du et al., "Agree to disagree: Adaptive ensemble knowledge distillation in gradient space," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2020, pp. 12345–12355. [Online]. Available: https://proceedings.neurips.cc/paper/2020/file/91c77393975889bd08f301c9e13a44b7-Paper.pdf
[59] S. You, C. Xu, C. Xu, and D. Tao, "Learning with single-teacher multi-student," in *Proc. AAAI Conf. Artif. Intell.*, 2018, pp. 4390–4397.
[60] X. Zhu et al., "Knowledge distillation by on-the-fly native ensemble," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2018, pp. 7528–7538.
[61] D. Chen, J.-P. Mei, C. Wang, Y. Feng, and C. Chen, "Online knowledge distillation with diverse peers," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 3430–3437.
[62] L. Tran et al., "Hydra: Preserving ensemble diversity for model distillation," 2020, *arXiv:2001.04694*.
[63] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 3354–3361.
[64] SSD: Single shot MultiBox detector–train the KITTI dataset, Mar. 2017. [Online]. Available: https://blog.csdn.net/jesse_mx/article/details/65634482
[65] T.-Y. Lin et al., "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.
[66] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
[67] K. Chen et al., "MMDetection: Open MMLab detection toolbox and benchmark," 2019, *arXiv:1906.07155*.
[68] A. Paszke et al., "PyTorch: An imperative style, high-performance deep learning library," in *Proc. 33rd Int. Conf. Neural Inf. Process. Syst.*, 2019, Art. no. 721.
[69] S. Zhang, C. Chi, Y. Yao, Z. Lei, and S. Z. Li, "Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 9756–9765.
[70] J. Dai et al., "Deformable convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 764–773.
[71] M. B. Muhammad and M. Yeasin, "Eigen-CAM: Class activation map using principal components," in *Proc. Int. Joint Conf. Neural Netw.*, 2020, pp. 1–7.
[72] Q. Tian, T. Arbel, and J. J. Clark, "Task dependent deep LDA pruning of neural networks," *Comput. Vis. Image Understanding*, vol. 203, 2021, Art. no. 103154.

Qizhen Lan received the bachelor's degree in information management and information systems from Tiangong University, Tianjin, China, in 2018, the bachelor's degree in business analytics in 2018, the master's degree in intelligence and analytics in 2019 both from the Bowling Green State University, Bowling Green, OH, USA, where he is currently working toward the Ph.D. degree in data science. His research interests include knowledge distillation, deep network pruning, and visual detection of autonomous vehicles.

Qing Tian received the B.Eng. degree in computer science and engineering from Yanshan University, Qinhuangdao, Hebei, China, in 2011, the M.Eng. and Ph.D. degrees in electrical engineering from McGill University, Montreal, QC, Canada in 2013 and 2021, respectively. He is currently an Assistant Professor with Bowling Green State University, Bowling Green, OH, USA. From 2019 to 2020, he was an Applied Scientist Intern with Amazon.com Inc. (Visual Search & AR), Palo Alto, CA, USA. From 2013 to 2014, he was a Software Developer with Nakisa Inc., Montreal, QC, Canada. His research interests include deep neural network compression, efficient neural architecture search, and adversarial AI. His current project in efficient and trustworthy self-driving visual perception is supported by the National Science Foundation.