001 002 003

004 005 006

007 008

009

010

000

# Auxiliary Modality Learning with Generalized Curriculum Distillation

Anonymous Authors<sup>1</sup>

# Abstract

Driven by the need from real-world applications, Auxiliary Modality Learning (AML) offers the pos-012 sibility to utilize more information from auxiliary data in training, while only requiring data from one or fewer modalities in testing, to save the 015 overall computational cost and reduce the amount of input data for inferencing. In this work, we formally define "Auxiliary Modality Learning" 018 (AML), systematically classify types of auxiliary modality (in visual computing) and architectures 020 for AML, and analyze their performance. We also analyze the conditions under which AML works well from the optimization and data distribution perspectives. To guide various choices to achieve optimal performance using AML, we propose a 025 novel method to assist in choosing the best auxiliary modality and estimating an upper bound performance before executing AML. In addition, 028 we propose a new AML method using generalized 029 curriculum distillation to enable more effective 030 curriculum learning. Our method achieves the best performance compared to other SOTA methods.

# 1. Introduction

Learning from images and videos is among some of the most popular research focuses (Esteva et al., 2021; Guo et al., 2022; Chai et al., 2021), as RGB images are informative and easy to acquire. In addition, RGB camera is cheap and can be easily deployed. There are also works considering multiple modalities, i.e., multi-modal learning (Wang, 2021; Jiang et al., 2021; Joshi et al., 2021). Furthermore, some works consider to use multiple modalities in training but use fewer modalities during test since in certain applications it's difficult to use all modalities during inference. For example, it is expensive to deploy Lidar on commodity self-driving cars, but it's reasonable to equip a few developer's cars with Lidar for training. However, this specific type of learning task, i.e., "test with fewer modalities than during training", is not standardized yet. For example, there is no formal term or definition. There have been concepts, such as "learning with side information" (Hoffman et al., 2016), "learning with privileged information" (Garcia et al., 2019), "learning with auxiliary modality" (Piasco et al., 2021), "learning with partial-modalities" (Wang et al., 2018), and "modality distillation" (Garcia et al., 2018), etc. We therefore formalize these learning tasks as *Auxiliary Modality Learning (AML)* in Sec. 3.

To apply AML to real-world tasks, there are some key issues: "what types of auxiliary modalities can be used, and how to add the auxiliary modalities into the network and make them most effective?" We systematically list and classify auxiliary modalities in visual computing and network architectures for AML, and then conduct experiments to address these questions. Specifically, we classify the auxiliary modalities into 3 types: low-level sensing data (Type 1), middle-level equivalent representation (Type 2), and high-level conceptual information (Type 3) in Sec. 4.1.1, according to the types of information. We also classify the network architectures into four types, according to the mechanism that introduces the auxiliary modality. They are auxiliary modalities in the input (Type A), in the middle (Type B), in the end (Type C), and in the teacher (Type D), as defined in Sec. 4.1.2. In addition, we design experiments to see which architecture and which auxiliary modality perform best within each task and across tasks (Sec. 4.1.3) that provides experimental guidelines and theoretical foundation to our method in Sec. 5.

Given this formal framing, we can apply AML to real-world tasks. There remains the question of explainability: "Why AML can work without auxiliary modality in the test?" It's not obvious that adding auxiliary modality only in training can always help improve test performance with only the main modality. In Sec. 4.2, we explore this line of inquiries from optimization and data perspectives. Specifically, we introduce a new concept of "supermodel" to support our claim, which also offers insights and inspiration to design the new AML method presented in Sec. 5.2.

Based on the detailed analysis in Sec. 4, we propose a simple

034

035

036

038

039

041

043

045

046

 <sup>&</sup>lt;sup>1</sup>Anonymous Institution, Anonymous City, Anonymous Region,
 Anonymous Country. Correspondence to: Anonymous Author
 <anon.email@domain.com>.</a>

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

<sup>052</sup> 053 054

vet effective method, Smart Auxiliary Modality Distillation (SAMD), that can smartly choose the best auxiliary modality 057 and perform a special auxiliary modality distillation with 058 generalized curriculum distillation. Firstly, in Sec. 4.1.3, 059 we show that different auxiliary modalities can contribute 060 to different tasks at a different level, thus we propose a 061 method to choose the best auxiliary modality and estimate 062 upper-bound performance for a given task before actually 063 executing AML. Inspired by Squeeze-and-Excitation Net-064 work (SENet) (Hu et al., 2018), we use channel-level atten-065 tion in the SE block to estimate the AML performance for 066 each modality, and show the consistently positive correla-067 tion between them through experiments on different tasks 068 and auxiliary modalities. See details in Sec. 5.1. Also, in 069 Sec. 4.1.3, we show the knowledge distillation based ar-070 chitecture (Type D) is better than other forms. However, 071 when analyzing the reason for the effectiveness of AML in Sec. 4.2, we find the "supermodel condition", which helps AML to perform better, is not fully utilized in the general 074 knowledge distillation based architecture. We thus introduce 075 a new method that uses supermodel in a more effective way 076 that allows the teacher network to be aware of the student's 077 status in a curriculum way, leading to a better distillation. 078 Our method achieves better performance compared to other 079 SOTA methods (Sec. 5.2).

Our analysis provides experimental understanding and theoretical underpinning for the simple yet effective method design. To the best of our knowledge, this is the first detailed analysis to guide the design and choices of AML methods for visual computing based on tasks, datasets, and network architectures. In summary, our contributions are:

087 · Systematically list and classify different types of aux-088 iliary modalities and architectures (Sec. 4.1.2) for 089 AML, and analyze the performance behavior of dif-090 ferent types of auxiliary modalities and architectures 091 for AML across different datasets, backbones and tasks 092 (Sec. 4.1.3). We find (1) architecture effectiveness is 093 relatively consistent across different tasks, datasets 094 and backbones; (2) auxiliary modality effectiveness is 095 consistent within one task with different datasets and 096 backbones, but not consistent across tasks. 097

• Propose a novel AML method, "Smart Auxiliary 098 Modality Distillation (SAMD)", that automatically (1) 099 100 chooses the best auxiliary modality for the main distillation process, and (2) performs knowledge distillation under a special "supermodel condition" to enable the teacher network to be aware of the student's status. 104 SAMD achieves SOTA results on variant tasks, with improvement up to 10% on end-to-end steering task, 105 5% on multi-view handwriting classification task, and 106 up to 15.6% across tasks, etc. (Sec. 5).

• Analyze and explain the reasons for the effectiveness

of AML from both optimization perspective and data perspective (Sec. 4.2), providing theoretical support to the SAMD method.

# 2. Related Work

Auxiliary Modality Learning aims to use auxiliary modality in training to boost the test performance without the auxiliary modality during inference. Cross-modality Learning and Knowledge Distillation are comparatively promising solutions and we discuss related works in each here.

#### 2.1. Cross-modality Learning

To utilize the prior knowledge between different modalities, Gupta et al. (Gupta et al., 2016) learned the representation of one modality with a pretrained network on another modality. Hoffman et al. (Hoffman et al., 2016) presented early work on modality hallucination, which used a hallucination network with RGB image as input but tried to mimic a depth network, by combining with RGB network to achieve multimodal learning. Some (Garcia et al., 2018; 2019) train the hallucination network with a different process to achieve better performance, while others (Wang et al., 2018; Piasco et al., 2021) use GAN or U-Net to generate another paired modality data with one modality. MSD (Jin et al., 2021) transfers knowledge from a teacher on multimodal tasks by learning the teacher's behavior within each modality. A recent work (Garcia et al., 2021) trains the different modality data in different pipelines and distills the best modality pipeline knowledge to other modality pipelines. In addition to action recognition, AML has also been applied in medical image processing (Gao et al., 2019; Li et al., 2020). Specifically, Zheng et al. (Zheng, 2015) investigated the effectiveness of shape priors learned from a different modality (e.g., CT) to improve the segmentation accuracy on the target modality (e.g., MRI). Valindria et al. (Valindria et al., 2018) proposed dual-stream encoder-decoder framework, which assigns each modality with a specific branch and extracts cross-modality features with carefully designed parameter sharing strategies. Li et al. (Li et al., 2020) exploited the priors of assisted modality to promote the performance on another modality by enhancing model generalization ability, where only target-modality data is required in the test.

#### 2.2. Knowledge Distillation

Knowledge Distillation can be classified as one-way or mutual-learning knowledge distillation. One-way knowledge distillation mainly distills the knowledge of a fixed teacher model (usually large) to a student model (usually small). In the early days, Hinton et al. (Hinton et al., 2015) proposed compressing the knowledge in an ensemble of multiple models into a single model that is much easier to deploy by mimicking the class distribution via softened

softmax from the ensemble teacher. Some studies (Ding 111 et al., 2019; Wen et al., 2019) went further to explore the 112 trade-off between the supervision of soft logits and hard 113 task label. Furthermore, there are also methods exploiting 114 the intermediate feature (Romero et al., 2015; Kim et al., 115 2018a; Jin et al., 2019) as transferred knowledge, which 116 can improve the middle layer's representational ability in a 117 student network. Other than the one-way distillation from 118 teacher to student, some focus on mutual knowledge dis-119 tillation among models trained from scratch. This line of 120 research is especially notable for scenarios without an avail-121 able pretrained teacher model. A significant work is deep 122 mutual learning (DML) (Zhang et al., 2018). During the training phase, DML uses a pool of randomly initialized 124 models as the student pool, and every student is guided by 125 the output of other students and the task label. 126

We share a similar philosophy with distillation, but aim to design a cross-modality learning framework to utilize the hidden information from auxiliary modalities, resulting in a different methodology.

# 3. Auxiliary Modality Learning

Auxiliary modality learning offers promising potential, but has not been fully examined. Previous works studied aux-135 iliary modality learning in certain applications without a 136 formal definition or a unified terminology. (Hoffman et al., 137 2016) named this process "learning with side information", 138 (Garcia et al., 2019) called it "learning with privileged in-139 formation", (Piasco et al., 2021) referred to it as "learning 140 with auxiliary modality", (Wang et al., 2018) suggested 141 "learning with partial-modalities", (Garcia et al., 2018) in-142 troduced the term "modality distillation", etc. In this paper, 143 we formally define the Auxiliary Modality Learning (AML) 144 as follows: 145

**Definition 3.1** Given data with one set of modalities  $I_M$ 147 and data with another set of modalities  $I_A$ , if a model  $\mathcal{M}$ 148 can take both  $I_M$  and  $I_A$  as input during training, but only 149 use  $I_M$  during test, then we call model  $\mathcal{M}$  an auxiliary 150 modality model,  $I_M$  as the main modality data and  $I_A$  as the 151 auxiliary modality data. Furthermore, we call the training 152 process of an auxiliary modality model as auxiliary modality 153 learning. 154

155 Formally, the training process is 156  $\min_{\theta} L(\mathcal{M}_{\theta}, (I_M, I_A), GT)$ , where  $\theta$  is the weights 157 of the model, L is the loss function, and GT is the ground 158 truth, while the test process is  $\mathcal{M}_{\theta}(I_M)$ .

The goal of AML is to achieve better performance with the help of auxiliary modalities  $I_A$  than only train on main modalities  $I_M$ . AML can be found in the real world, e.g., when you cannot solve a problem in class, the teacher gives you some hints so the students can understand the relationship between the problem and the answer better. Then, after class, the student can solve similar types of new problems without hints. In this paper, for visual computing, we fix the main modality as RGB images, but the auxiliary modality can be others, like point cloud, depth map, or other customized formats.

AML is useful in the following scenarios: (1) Getting the extra modality data during test is not feasible. For example, the extra modality can be the human-labeled attention map, which is achievable during training, but we cannot ask the user to label the attention map in real time. (2) Getting the extra modality data during test is feasible but expensive. For example, in autonomous driving, we need to use Lidar to get point cloud data. Using point clouds during training only requires several Lidar sensors on the cars for development, but using point clouds during test means every car needs to install Lidar, which is costly. AML can reduce the cost dramatically compared with the solutions that require the Lidar+camera, and can perform better than the solutions that only use camera. Similarly for robot navigation.

# 4. Analysis

In this section, we aim to do analysis on two key problems of AML when applying on real-world tasks: (1) What kinds of auxiliary modalities can we use, and how can we add them into the network to make them effective? (2) Why AML can work without auxiliary modality in test?

#### 4.1. Auxiliary Modality and Architecture

In this section, we first systematically list and classify the types of auxiliary modalities and the types of auxiliary modality learning architecture, then analyze how different auxiliary modalities and architectures can affect auxiliary modality learning through experiments.

#### 4.1.1. TYPES OF AUXILIARY MODALITY

Previous auxiliary modality learning works usually only consider one or several given types of auxiliary modality without systematic analysis (Hoffman et al., 2016; Garcia et al., 2019; Piasco et al., 2021; Wang et al., 2018; Garcia et al., 2018; Gupta et al., 2016; Jin et al., 2021). This is usually because of the limitation of data sources, e.g., limited sensor types. However, there is actually a wide range of auxiliary modality from the sensors (like depth map or infra-red image), other data generated from the original image (like segmentation image or frequency image) or even annotated by a human expert (like attention map) can also be used as an auxiliary modality. In our work, we classify the potentially useful auxiliary modalities that are

#### Auxiliary Modality Learning with Generalized Curriculum Distillation



172

173

174

175

176

177

178

179

208

*Figure 1.* Architectures for auxiliary modality learning. Type A: Auxiliary modality in the input. Type B: Auxiliary modality in the middle. Type C: Auxiliary modality in the end. Type D: Auxiliary modality in the teacher network. The dashed area in each type is the test pipeline that only use main modality. See Sec. 4.1.2.

commonly seen in daily life and show their effectiveness through experiments.

Formally, we suggest the following three types of data,which can be used as auxiliary modality in visual computing,according to the information contained in them:

Type 1: Low-level sensing data with additional information. For example, given the main modality is RGB image, depth map or infra-red image can be used as an auxiliary modality, which is already commonly used (Wang et al., 2018; Xiao et al., 2020). The additional depth or infra-red information can be used when the RGB image is not able to capture enough information like at night.

**Type 2: Middle-level representation with equivalent information** but in *different spaces*. For example, RGB image can be transferred to/from frequency space with 2D FFT, which is a one-to-one mapping. Although they contain the same information, one presentation in one space may have a closer relation to the goal, thus helping the network to learn better.

Type 3: High-level conceptual data with compacted information. For example, expert annotated image with emphasized key features (like attention image). This kind of auxiliary modality helps reduce the redundant noises and helps the network focus on key elements quickly.

Type 1 is the most common type of auxiliary modality, but Type 2 and 3 are also auxiliary modalities that can potentially contribute to the task. See example tasks for different modalities in Appendix A.1.

# 209 4.1.2. ARCHITECTURES FOR AUXILIARY MODALITY210 LEARNING

211 Existing works explore variant ways to achieve the goal of 212 auxiliary modality learning. However, to the best of our 213 knowledge, no one compares architectures in a systematic 214 way (Hoffman et al., 2016; Garcia et al., 2019; Piasco et al., 215 2021; Wang et al., 2018; Garcia et al., 2018; Gupta et al., 216 2016; Jin et al., 2021). In this section, we list and compare 217 the existing architecture designs for the auxiliary modality 218 learning systematically. 219

We classify the possible auxiliary modality learning architectures into four types:

**Type A:** Auxiliary modality in the input, same architecture as multi-modality learning during training, but only use the main modality branch for test, as shown in Fig. 1(a). General multi-modality architecture is already been studied (Xiao et al., 2020), but multi-modality based AML still needs to be explored.

**Type B: Auxiliary modality in the middle as supervision**. The basic idea is to generate auxiliary modality with the main modality first, and then use the multi-modality architecture, as shown in Fig. 1(b). Existing works like (Wang et al., 2018; Li et al., 2020; Wei et al., 2016) show the effectiveness of this type of solution.

**Type C:** Auxiliary modality in the end as supervision, same architecture as multi-task learning (Ruder, 2017) or indirect supervision (Chang et al., 2010), but only need to use the original task pipeline for test, as shown in Fig. 1(c). The basic idea is the original task and the auxiliary modality generation task share certain common features, thus the auxiliary modality can help the learning of the original task.

**Type D: Auxiliary modality in a teacher network** and *teach a student network without auxiliary modality, refer to cross-modality knowledge distillation*, as shown in Fig. 1(d). Existing works like (Garcia et al., 2021; 2018) show the effectiveness of this type of solution.

Notice existing works mostly focus on Type B and D, but few discuss or conduct experiments with Type A and C, which are also potential solutions.

#### 4.1.3. EXPERIMENTS

We conduct experiments to see how different auxiliary modalities and architectures of AML perform within single task and across tasks.

**Single Task** In this experiment, we consider four factors, auxiliary modality, architectures, backbones and datasets. We design experiments to answer: (1) Given fixed dataset and backbone, do all the architectures help auxiliary modality learning? What's the order among them w.r.t. performance improvement? (2) Given fixed datasets and backbones, do all the auxiliary modalities help auxiliary modality learning? What's the order among them w.r.t. performance improvement? (3) Are the previous two answers consistent across different datasets and backbones?

The experiment setting is described in Appendix A.1. In Table 5 (Appendix A.2), we show performance comparison (Mean Accuracy %) with a combination of three auxiliary modalities, four architectures, two backbones, and two datasets. We find: (1) Knowledge distillation based architecture (Type D) per-form best, followed by generation based architecture (Type

B). Multi-task based architecture does slightly better than

223 baseline (Type C), while Multi-modality based architecture

sometimes hurt the performance (Type A).

(2) All types of auxiliary modality used can help improve performance. Attention image (Type 3) achieves the highest performance improvement, followed by depth map (Type 1). Frequency image (Type 2) only performs slightly better than baseline.

(3) Within this task, the effectiveness order for different
auxiliary modalities or architectures are consistent across
different datasets or backbones.

Multiple Tasks. In this experiment, we consider four
factors: auxiliary modality, architectures, backbones and
datasets. We design experiments to answer: (1) Is the effectiveness of different architectures consistent across different
tasks? (2) Is the effectiveness of different auxiliary modalities consistent across different tasks?

The experiment setting is presented in Appendix A.1. In
Table 6 (Appendix A.2), we show performance comparison
with a combination of two auxiliary modalities and four
architectures across three tasks. Notice when comparing
across tasks, we only focus on the relative accuracy order
of one task, since the metrics of different tasks are different.
We find:

(1) The effectiveness order of different auxiliary modalities
may be different for different tasks.

(2) The effectiveness order of different architectures is con-sistent for different tasks.

#### 4.2. Why AML Works?

254

266

267

274

255 We provide experimental results in Sec. 4.1.3 to show aux-256 iliary modality learning can work, i.e., although only test 257 with the main modality, using auxiliary modality during 258 training can do better than only using the main modality. 259 This is also supported by other works (Hoffman et al., 2016; Garcia et al., 2019; Piasco et al., 2021; Wang et al., 2018; 261 Garcia et al., 2018). However, most of them use experimental results to illustrate their effectiveness, there is no detailed 263 analysis to demonstrate why AML can work. In this section, 264 we explain why AML works from two perspectives. 265

#### 4.2.1. OPTIMIZATION PERSPECTIVE

Here we explain why AML can work from the optimization
perspective.

(1) The optimal solution of AML is no worse than learning
with the main modality. Inspired by "superset", we first
introduce a new concept "supermodel".

**Definition 4.1** Given a model  $\mathcal{M}_{\theta_A}^{(A)}(I_A)$  (weights  $\theta_A$  and input  $I_A$ ), and a model  $\mathcal{M}_{\theta_B}^{(B)}(I_B)$  (weights  $\theta_B$  and input  $I_B$ ), if for any  $\theta_A$ , there is a  $\theta_B$ , such that  $\mathcal{M}_{\theta_A}^{(A)}(I_A) = \mathcal{M}_{\theta_B}^{(B)}(I_B)$  for any arbitrary valid input data  $I_A$  and its superset  $I_B$ . We call model  $\mathcal{M}_B$  as a "supermodel" of  $\mathcal{M}_A$ .

See an example of the supermodel in Fig. 5. We then introduce a lemma based on the supermodel:

**Lemma 4.1** Given a model  $\mathcal{M}$  and its supermodel  $\mathcal{M}^{(s)}$ , the optimal training loss of  $\mathcal{M}^{(s)}$  (which is  $\arg \min_{\theta^{(s)}} L(\mathcal{M}^{(s)}_{\theta^{(s)}}(I^{(s)}), GT))$  is less than or equal to the optimal training loss of  $\mathcal{M}$  (which is  $\arg \min_{\theta} L(\mathcal{M}_{\theta}(I), GT)$ ). where L is the loss function and GT is the ground truth.

See the proof in Appendix A.4. Now we consider the single network architectures (Type A, B, C in Sec. 4.1.2) and the teacher network of Type D, all of them are supermodels of their related main modality pipeline network. Specifically, we can black out the auxiliary modality related branch (e.g., for Type A and C, use the pipeline in the dashed box, for Type B, blackout the auxiliary modality generation branch, for teacher network in Type D, it's the same as Type 1) by setting the weights of connection layers to specific values (e.g., zeros, depends on the specific type of layer), then the model takes both main and auxiliary modalities will have exactly the same results of the model with only main modality, thus meeting the supermodel definition A.1. According to Lemma. 4.1, the AML model is no worse than the original model with only the main modality.

(2) AML allows the optimizer to search in a higher dimension with a higher possibility to find a path to the same solution as learning with the main modality.

Suppose an optimizer g takes model  $\mathcal{M}$  and its initial weights  $\theta_0$ , loss function L, training data  $I_M$  as input, and output a path of model weights:

$$g(\mathcal{M}, \theta_0, L, I_M) = \{\theta_0, \theta_1, ..., \theta_{p_1}\} = P_1$$

where p1 is the step number,  $\theta_{p_1} = \theta^*$  is the optimal solution, and  $P_1$  is the path. Then the AML process on its supermodel  $\mathcal{M}^{(s)}$  is

$$g(\mathcal{M}^{(s)}, \theta_0 \oplus \delta_0, L, (I_M, I_A))$$
$$= \{\theta_0 \oplus \delta_0, \theta_1 \oplus \delta_1, ..., \theta_{p_2} \oplus \delta_{p_2}\} = P2$$

where  $\oplus$  is the dimension-level connection,  $\delta$  as the weights for the auxiliary dimension,  $\delta_0 = \delta_{p_2} = 0$ ,  $\theta_{p_2} = \theta^*$ . This means that the start and end positions are only on the main modality dimension, while in the middle it can explore on



*Figure 2.* "Why AML works" from optimizer perspective. The blue path (AML) is easier to be found by a gradient-based optimizer since there is no uphill, as the red path (main modality only).

a higher dimension (main+auxiliary modality). For any path  $P_1^*$ , there is a path  $P_2^*$  that represents the same path (by setting  $p_2 = p_1$ ,  $\delta_i = 0$  and use the same  $\theta_i$  for i = $0, 1, ..., p_1$ ). But for a  $P_2^*$ , there's no  $P_1^*$  that can represents the same path when there is a  $\delta_i \neq 0$  in  $P_2^*$ . It shows even with the same start and end points, the AML can have more path options, which may be easier to be found by a given optimizer, e.g., the blue path in Fig. 2 is a gradient descent path in a higher dimension, while the red path in low dimension needs to go uphill in the middle, which is more difficult for the gradient-based optimizer to find the best solution.

#### 4.2.2. DATA PERSPECTIVE

<sup>o</sup> Next, we explain why AML works from data perspective.

(1) The auxiliary modality can help the main modality train-308 ing better when main modality data is imbalanced or in 309 shortage. For example, the main modality data has few 310 examples that are the 'hard cases', which lead to a wrong 311 decision boundary. This is common in real-world datasets, 312 e.g., the autonomous driving dataset usually has fewer night 313 data, even worse, has few accident data. After adding the 314 auxiliary modality that provides more information on the 315 hard cases, it would be easier to learn a correct decision 316 boundary, then use this information to guide the training 317 process with the main modality. For example, the infra-red 318 image or depth map contains more information than RGB 319 320 image when captured at night. This observation explains why the low-level sensing data (Type 1 in Sec. 4.1.1) can help AML. See figures in Appendix A.5. 322

(2) The auxiliary modality data reveal a simpler mapping function from input to output. As we know, the network is used to learn a mapping function from input to output, e.g.,  $f(I_M) = y$ . However, the function f may be complex and difficult to learn. Then, one solution is to split the complex function f into two parts, i.e.,  $f(I_M) = f_2(f_1(I_M)) =$   $f_2((I_M, I_A)) = y$ , where  $f_1$  is "data reformating function" that contains as much as inductive bias (according to the domain expert experiences) for the given task, thus the  $f_2$  will be simpler than the original f and easier to be learned. This observation explains why middle-level and high-level conceptual data (Type 2 and 3 in Sec. 4.1.1) can help AML.

# 5. Smart Auxiliary Modality Distillation (SAMD)

Based on the detailed analysis in Sec. 4, we propose a simple yet effective method "Smart Auxiliary Modality Distillation" to choose the best auxiliary modality and do an auxiliary modality distillation.

#### 5.1. Auxiliary Modality Choice for a Given Task

As discussed in Sec. 4.1.1, there are three types of auxiliary modality that are potentially useful, and each type can have multiple kinds of modalities. Given the conclusion in Sec. 4.1.3, there is no consistent best auxiliary modality that can be used for all the tasks, we need to choose the best auxiliary modality that can boost the performance most for a given task. Suppose there are n types of auxiliary modalities, do we need to train n times to find out the best one? The answer is no. In this section, we propose a method that can assist in deciding the importance order for a set of auxiliary modalities within one training process.

Inspired by Squeeze-and-Excitation Network (SENet) (Hu et al., 2018), we use channel-level attention to represent the importance of each modality. Suppose we already have a network f that can take the main modality  $I_m$  as input and perform prediction for a given task. Now we have n types of auxiliary modalities that potentially can help. We first pack the different modality data in the channel level, and feed them into the Squeeze-and-Excitation (SE) block (Hu et al., 2018), followed by a 1x1 convolutional layer to make the channel number to be the same as the main modality  $I_m$ , so that the original network f can take that as input and perform prediction. If different modality data has different image sizes then they should be resized (or add a shallow network to pre-process the data, if necessary) before being packed in the channel level. In our experiments, all the image data with different modalities have the same size, so they can be packed directly. After training the modified network, the channel weights in the SE block can be used to determine the relative importance for the auxiliary modalities, i.e., the modality that has the largest channel weight is the one that can lead to the best AML performance.

#### 5.2. Auxiliary Modality Distillation

Sec. 4.1.3 shows the knowledge-distillation (KD) based architecture performs best in most cases. However, when



*Figure 3.* SAMD architecture. In each round, a new curriculum learning is started by resetting the teacher weights. Then we train the model with our online distillation, until the student converges. The teacher network should be a supermodel of the student network to enable reset operation, which helps the teacher be aware of the student's status and perform more effectively.

341

342

343

344

345

346347348349350351

352

353

354

355

356 357

	Accuracy (%) on various angle threshold $\tau$ (degree)								
Method	$\tau = 1.5$	$\tau=3.0$	$\tau=7.5$	$\tau = 15$	mAcc				
(Hoffman et al., 2016)	51.7	70.6	89.6	94.7	83.6				
(Garcia et al., 2018)	26.1	54.1	81.8	91.0	74.6				
(Xiao et al., 2020)	28.6	51.2	80.0	92.0	74.4				
(Garcia et al., 2021)	40.2	67.8	88.7	94.3	81.0				
Ours (SAMD)	54.3	72.2	90.1	94.6	84.4				

Table 1. Performance comparison on Audi dataset with Nvidia PilotNet (Bojarski et al., 2016). All the methods are trained on RGB+segmentation, and tested on RGB only. Our method outperforms others by up to 10% improvement in accuracy.

analyzing reasons for the effectiveness of AML in Sec. 4.2,
we find the "supermodel condition", which helps AML to
perform better, is not fully utilized in the general KD-based
architecture. We thereby introduce a new method that uses
"supermodel condition" that allows the teacher network to be
aware of the student's status and leads to a better distillation.

364 We update the teacher-student in an online-like paradigm. 365 See framework illustration in Fig. 3. The training paradigm 366 contains t rounds. In each round, we first reset the teacher 367 with the student, then train the teacher independently while 368 training the student with both the general label loss and 369 knowledge distillation loss for k epochs. k should not be 370 too large to avoid the teacher being far away from the stu-371 dent. The training process stops when the student converges 372 between different rounds or until finishing t rounds. See loss 373 function, "reset" definition, and algorithm in Appendix A.7. 374

To apply our training paradigm with *reset* operation, the framework should meet the *supermodel* condition (Sec. 4.2), i.e., *the teacher network should be a* supermodel *of the student network*. This condition is what differentiates our learning framework from other existing methods.

### **5.3. Experiments**

380

382

383

384

In this section, we conduct experiments for autonomous steering task (Appendix A.1) and 5 other tasks (Ap-



*Figure 4.* Different types of auxiliary modalities used in our experiments.

	1	Aean Accura	cy (%)
Method	w/o ours	with ours	Improvement
kd (Hinton et al., 2015)	71.5	83.4	11.9
hint (Romero et al., 2015)	67.6	83.2	15.6
similarity (Tung & Mori, 2019)	75.6	83.9	8.3
correlation (Peng et al., 2019)	77.0	74.3	-2.7
rkd (Park et al., 2019)	75.6	84.4	8.8
pkt (Passalis et al., 2020)	75.7	76.4	0.7
vid (Ahn et al., 2019)	83.4	83.2	-0.2
abound (Heo et al., 2019)	74.3	72.0	-2.3
factor (Kim et al., 2018b)	76.9	83.4	6.5
fsp (Yim et al., 2017)	72.0	70.1	-1.9

Table 2. Performance comparison with vs. without our training paradigm (containing reset operation). By applying our training paradigm on other knowledge distillation methods, we can achieve better performance in most cases (up to +15.6%) in either fully paired or merely a small amount of additional modality data.

pendix A.8). See details on the experiment settings in Appendix A.8. We also use different types of data modalities in our experiments, as shown in Fig. 4.

**Comparison with other AML methods.** We compare our SAMD with other AML methods, Hoffman et al. (Hoffman et al., 2016), Garcia et al. (Garcia et al., 2018), Xiao et al. (Xiao et al., 2020), and DMCL (Garcia et al., 2021), using Audi dataset (Geyer et al., 2020) and Nvidia Pilot-Net (Bojarski et al., 2016). For Xiao et al. (Xiao et al., 2020), we adopt the single-sensor version and make it suitable for the Audi dataset by removing the high-level route navigation command and measurement, and using Tao et al. (Tao et al., 2020) as the segmentation generator. In Table 1, ours outperforms others by up to **10%**.

Effectiveness when combining with different knowledge distillation methods. Since our training paradigm can be applied to existing knowledge distillation methods, we do experiments by combining ours with kd (Hinton et al., 2015), hint (Romero et al., 2015), similarity (Tung & Mori, 2019), correlation (Peng et al., 2019), rkd (Park et al., 2019), pkt (Passalis et al., 2020), abound (Heo et al., 2019), factor (Kim et al., 2018b), fsp (Yim et al., 2017). From Table 2, our method achieves up to **15.6%** improvement in both settings, showing the effectiveness of our training paradigm (containing *reset* operation). See more details in Appendix A.8.

Auxiliary Modality Learning with Generalized Curriculum Distillation

Dataset	Train Mod	Test Mod	Method	mAcc
Audi	RSDE	RSDE	Teacher	83.7
Audi	RSDE	RGB	Best Others	72.9
	RSDE	RGB	Ours	74.3
SullyChen	RDE	RDE	Teacher	81.0
SullyChen	RDE	RGB	Best Others	88.9
	RDE	RGB	Ours	89.7
Honda	RSDE	RSDE	Teacher	79.8
Honda	RSDE	RGB	Best Others	77.4
	RSDE	RGB	Ours	78.1

*Table 3.* Comparison on different datasets and different modalities. "RSDE" refers to RGB + segmentation + depth map + edge map, and "RDE" for RGB + depth map + edge map. Our method outperforms others on different datasets and different additional modalities by up to +11% accuracy improvement. "Best others" stand for the best performance with methods in Table 1.

**Comparison on different datasets and modalities.** We also perform comparison with other knowledge distillation methods on different datasets (Audi (Geyer et al., 2020), Honda (Ramanishka et al., 2018), and SullyChen (Chen, 2018)) and different modalities (RGB, segmentation, depth map, and edge map). Specifically, Audi dataset contains ground truth segmentation, and other segmentation is generated by Tao et al. (Tao et al., 2020), while the depth map is generated by (Bian et al., 2019) and the edge map is generated by DexiNet (Poma et al., 2020). In Table 3, Our method outperforms others in nearly all cases by up to +11% accuracy improvement. See more details in Appendix A.8.

Comparison on other tasks and modalities. We perform comparison on multi-feature handwritten classification task (Han et al., 2021). We regard the six feature sets as six modalities, and treat each of them as a target modality in each experiment. Our method outperforms others with 5.1% on average. We also conducted experiments on another end-to-end autonomous driving task, "way-point prediction" task (Prakash et al., 2021). We use RGB image as main modality, and point cloud as auxiliary modality, and achieve 19% improvement on average route completion, compared to RGB image baseline. In the materials classification task (Wilson et al., 2022), we use RGB image as main modality, while using sound wave as auxiliary modality, achieving 6.4% performance gain. For the bird-eye-view segmentation task (Li et al., 2022), point-cloud from multiple vehicles are used during training, and point cloud from 432 only one vehicle is used during test. We get 0.78% accuracy 433 434 improvement. See Table 4 for a simplified comparison, and 435 more details in Appendix A.8.

Relation of Channel-level Importance and AML Performance. To show the channel-level attention for different auxiliary modalities is positively correlated to the final per-

Task	Train Mod	Test Mod	Ours	Best Others
Handwritten Clas (Han et al., 2021)	Multi-features	Single Feature	70.3	65.2
Waypoint Pred (Prakash et al., 2021)	Image Point Cloud	Image	79.5	71.4
Materials Clas (Wilson et al., 2022)	Image Audio	Image	83.2	76.8
Bird-eye-view Seg (Li et al., 2022)	Multi-view Point Cloud	Single-view Point Cloud	45.30	44.91

*Table 4.* **Performance comparison on different tasks with dif-ferent auxiliary modalities.** Our method outperforms other methods on all tasks. See details in Appendix A.8.

formance of AML with different auxiliary modalities, we conduct experiments on three tasks with the same setting stated in Sec. 4.1.3, then use the same three auxiliary modalities and an additional random noise modality (whose importance should be the lowest). As shown in Table 11, in Task 1, the importance order from the channel-level attention is attention image > depth map > frequency image, and the performance order from AML is also attention image > depth map > frequency image to perform one-time training to select the best modality for a given task. See more details in Appendix A.8.

#### 6. Conclusion

This paper aims to introduce and understand 'Auxiliary Modality Learning (AML)'. We first formalize the concept of AML, systematically list and classify the types of auxiliary modality and architectures for AML. We then analyze how types of auxiliary modality and architectures can affect AML performance on a single task and across tasks, i.e., best architecture is consistent within a task or across tasks, while best auxiliary modality is consistent within one task but not consistent across tasks. We also analyze the reason of the effectiveness of AML in optimization and data perspectives to provide theory support for our method. Given the observation, we propose a method SAMD that can first determine the best auxiliary modality, and then do a special auxiliary modality distillation that enables the teacher network to be aware of the student's status, leading to a better distillation that achieves the SOTA performance.

**Limitations and Future Work:** We assume that the teacher network is a *supermodel* of the student's. In modality distillation, such an assumption is reasonable, as this task focuses on the reduction of modality, instead of model size, like general knowledge distillation. A possible future direction for AML is to further examine the impact of auxiliary modality data size, e.g., can we use only a small amount of auxiliary modality data to achieve better performance? What if data is not paired with the main modality? Are there better architectures?

# 440 **References**

441

- Ahn, S., Hu, S. X., Damianou, A., Lawrence, N. D., and Dai, Z. Variational information distillation for knowledge transfer. In *Proceedings of the IEEE/CVF Conference* on Computer Vision and Pattern Recognition, pp. 9163– 9171, 2019.
- Bian, J., Li, Z., Wang, N., Zhan, H., Shen, C., Cheng, M. M., and Reid, I. Unsupervised scale-consistent depth and
   ego-motion learning from monocular video. *Advances in neural information processing systems*, 32:35–45, 2019.
- Bojarski, M., Del Testa, D., Dworakowski, D., Firner, B., Flepp, B., Goyal, P., Jackel, L. D., Monfort, M., Muller, U., Zhang, J., et al. End to end learning for self-driving cars. arXiv preprint arXiv:1604.07316, 2016.
- Chai, J., Zeng, H., Li, A., and Ngai, E. W. Deep learning
   in computer vision: A critical review of emerging techniques and application scenarios. *Machine Learning with Applications*, 6:100134, 2021.
- Chang, M.-W., Srikumar, V., Goldwasser, D., and Roth, D. Structured output learning with indirect supervision. In *ICML*, pp. 199–206, 2010.
- Chen, S. A collection of labeled car driving datasets, https://github.com/sullychen/driving-datasets, 2018.
- Cochran, W. T., Cooley, J. W., Favin, D. L., Helms, H. D.,
  Kaenel, R. A., Lang, W. W., Maling, G. C., Nelson, D. E.,
  Rader, C. M., and Welch, P. D. What is the fast fourier
  transform? *Proceedings of the IEEE*, 55(10):1664–1674,
  1967.
- Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler,
  M., Benenson, R., Franke, U., Roth, S., and Schiele, B.
  The cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- Ding, Q., Wu, S., Sun, H., Guo, J., and Xia, S.-T. Adaptive
  regularization of labels. *arXiv preprint arXiv:1908.05474*, 2019.
- 481
  482
  482
  483
  484
  484
  484
  484
  485
  484
  484
  486
  486
  487
  487
  488
  488
  488
  489
  489
  489
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
  480
- 485 Esteva, A., Chou, K., Yeung, S., Naik, N., Madani, A.,
  486 Mottaghi, A., Liu, Y., Topol, E., Dean, J., and Socher,
  487 R. Deep learning-enabled medical computer vision. *NPJ*488 *digital medicine*, 4(1):1–9, 2021.
- Gao, Z., Chung, J., Abdelrazek, M., Leung, S., Hau, W. K., Xian, Z., Zhang, H., and Li, S. Privileged modality distillation for vessel border detection in intracoronary imaging. *IEEE transactions on medical imaging*, 39(5):1524– 1534, 2019.

- Garcia, N. C., Morerio, P., and Murino, V. Modality distillation with multiple stream networks for action recognition. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 103–118, 2018.
- Garcia, N. C., Morerio, P., and Murino, V. Learning with privileged information via adversarial discriminative modality distillation. *IEEE transactions on pattern analysis and machine intelligence*, 42(10):2581–2593, 2019.
- Garcia, N. C., Bargal, S. A., Ablavsky, V., Morerio, P., Murino, V., and Sclaroff, S. Distillation multiple choice learning for multimodal action recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2755–2764, 2021.
- Geyer, J., Kassahun, Y., Mahmudi, M., Ricou, X., Durgesh, R., Chung, A. S., Hauswald, L., Pham, V. H., Mühlegg, M., Dorn, S., Fernandez, T., Jänicke, M., Mirashi, S., Savani, C., Sturm, M., Vorobiov, O., Oelker, M., Garreis, S., and Schuberth, P. A2D2: Audi Autonomous Driving Dataset. 2020. URL https://www.a2d2.audi.
- Guo, M.-H., Xu, T.-X., Liu, J.-J., Liu, Z.-N., Jiang, P.-T., Mu, T.-J., Zhang, S.-H., Martin, R. R., Cheng, M.-M., and Hu, S.-M. Attention mechanisms in computer vision: A survey. *Computational Visual Media*, pp. 1–38, 2022.
- Gupta, S., Hoffman, J., and Malik, J. Cross modal distillation for supervision transfer. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2827–2836, 2016.
- Han, Z., Zhang, C., Fu, H., and Zhou, J. T. Trusted multiview classification. *arXiv preprint arXiv:2102.02051*, 2021.
- He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.
- Heo, B., Lee, M., Yun, S., and Choi, J. Y. Knowledge transfer via distillation of activation boundaries formed by hidden neurons. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 3779–3787, 2019.
- Hinton, G., Vinyals, O., and Dean, J. Distilling the knowledge in a neural network. *Advances in Neural Information Processing Systems (NIPS)*, 2015.
- Hoffman, J., Gupta, S., and Darrell, T. Learning with side information through modality hallucination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 826–834, 2016.

Hu, J., Shen, L., and Sun, G. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7132–7141, 2018.

- Jiang, X., Ma, J., Xiao, G., Shao, Z., and Guo, X. A review
  of multimodal image matching: Methods and applications. *Information Fusion*, 73:22–71, 2021.
- Jin, W., Sanjabi, M., Nie, S., Tan, L., Ren, X., and
  Firooz, H. Modality-specific distillation. *arXiv preprint arXiv:2101.01881*, 2021.
- Jin, X., Peng, B., Wu, Y., Liu, Y., Liu, J., Liang, D., Yan, J.,
  and Hu, X. Knowledge distillation via route constrained
  optimization. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1345–1354, 2019.
- Joshi, G., Walambe, R., and Kotecha, K. A review on
  explainability in multimodal deep neural nets. *IEEE*Access, 2021.
- Kim, J., Park, S., and Kwak, N. Paraphrasing complex network: Network compression via factor transfer. *Advances in Neural Information Processing Systems (NIPS)*, pp. 2765–2774, 2018a.
- Kim, J., Park, S., and Kwak, N. Paraphrasing complex network: Network compression via factor transfer. *arXiv preprint arXiv:1802.04977*, 2018b.
- Li, K., Yu, L., Wang, S., and Heng, P.-A. Towards crossmodality medical image segmentation with online mutual knowledge distillation. In *Proceedings of the AAAI con- ference on artificial intelligence*, volume 34, pp. 775–783, 2020.
- Li, Y., Ren, S., Wu, P., Chen, S., Feng, C., and Zhang,
  W. Learning distilled collaboration graph for multi-agent perception. *Advances in Neural Information Processing Systems*, 34:29541–29552, 2021.
- Li, Y., Ma, D., An, Z., Wang, Z., Zhong, Y., Chen, S., and
  Feng, C. V2x-sim: Multi-agent collaborative perception
  dataset and benchmark for autonomous driving. *IEEE Robotics and Automation Letters*, 7(4):10914–10921,
  2022.
- Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. Microsoft coco:
  Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.
- Liu, Y.-C., Tian, J., Glaser, N., and Kira, Z. When2com: Multi-agent perception via communication graph grouping. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, pp. 4106–4115, 2020a.

- Liu, Y.-C., Tian, J., Ma, C.-Y., Glaser, N., Kuo, C.-W., and Kira, Z. Who2com: Collaborative perception via learnable handshake communication. In 2020 IEEE International Conference on Robotics and Automation (ICRA), pp. 6876–6883. IEEE, 2020b.
- Park, W., Kim, D., Lu, Y., and Cho, M. Relational knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3967–3976, 2019.
- Passalis, N., Tzelepi, M., and Tefas, A. Probabilistic knowledge transfer for lightweight deep representation learning. *IEEE Transactions on Neural Networks and Learning Systems*, 32(5):2030–2039, 2020.
- Peng, B., Jin, X., Liu, J., Li, D., Wu, Y., Liu, Y., Zhou, S., and Zhang, Z. Correlation congruence for knowledge distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5007–5016, 2019.
- Piasco, N., Sidibé, D., Gouet-Brunet, V., and Demonceaux, C. Improving image description with auxiliary modality for visual localization in challenging conditions. *International Journal of Computer Vision*, 129(1):185–202, 2021.
- Poma, X. S., Riba, E., and Sappa, A. Dense extreme inception network: Towards a robust cnn model for edge detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 1923– 1932, 2020.
- Prakash, A., Chitta, K., and Geiger, A. Multi-modal fusion transformer for end-to-end autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7077–7087, 2021.
- Ramanishka, V., Chen, Y.-T., Misu, T., and Saenko, K. Toward driving scene understanding: A dataset for learning driver behavior and causal reasoning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7699–7707, 2018.
- Romero, A., Ballas, N., Kahou, S. E., Chassang, A., Gatta, C., and Bengio, Y. Fitnets: Hints for thin deep nets. *International Conference on Learning Representations* (*ICLR*), 2015.
- Ruder, S. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*, 2017.
- Shen, Y., Zheng, L., Shu, M., Li, W., Goldstein, T., and Lin, M. C. Gradient-free adversarial training against image corruption for learning-based steering. In *Neural Information Processing Systems (NIPS)*, 2021.

- Tao, A., Sapra, K., and Catanzaro, B. Hierarchical multiscale attention for semantic segmentation. *arXiv preprint arXiv:2005.10821*, 2020.
- Tian, Y., Krishnan, D., and Isola, P. Contrastive representation distillation. In *International Conference on Learning Representations*, 2020.
- Tung, F. and Mori, G. Similarity-preserving knowledge distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1365–1374, 2019.
- 561 UCI. Multiple Features Data Set,
  562 https://archive.ics.uci.edu/ml/datasets/multiple+features,
  563 0.
- Valindria, V. V., Pawlowski, N., Rajchl, M., Lavdas, I.,
  Aboagye, E. O., Rockall, A. G., Rueckert, D., and
  Glocker, B. Multi-modal learning from unpaired images: Application to multi-organ segmentation in ct and
  mri. In 2018 IEEE winter conference on applications of
  computer vision (WACV), pp. 547–556. IEEE, 2018.
- Wang, L., Gao, C., Yang, L., Zhao, Y., Zuo, W., and Meng,
  D. Pm-gans: Discriminative representation learning for action recognition using partial-modalities. In *Proceedings of the European Conference on Computer Vision* (*ECCV*), pp. 384–401, 2018.
- Wang, Y. Survey on deep multi-modal data analytics: Collaboration, rivalry, and fusion. ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM), 17(1s):1–25, 2021.
- Wei, S.-E., Ramakrishna, V., Kanade, T., and Sheikh, Y.
  Convolutional pose machines. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 4724–4732, 2016.
- Wen, T., Lai, S., and Qian, X. Preparing lessons: Improve knowledge distillation with better supervision. *arXiv* preprint arXiv:1911.07471, 2019.
- Wilson, J., Rewkowski, N., and Lin, M. C. Audio-visual depth and material estimation for robot navigation. In 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 9239–9246. IEEE, 2022.
- Xiao, Y., Codevilla, F., Gurram, A., Urfalioglu, O., and López, A. M. Multimodal end-to-end autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, 2020.
- Yim, J., Joo, D., Bae, J., and Kim, J. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4133–4141, 2017.

- Zhang, Y., Xiang, T., Hospedales, T. M., and Lu, H. Deep mutual learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4320–4328, 2018.
- Zheng, Y. Cross-modality medical image detection and segmentation by transfer learning of shapel priors. In 2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI), pp. 424–427. IEEE, 2015.

# 605 A. Appendix

606

607

#### A.1. Details on Experimental Settings

608 Single task. We use autonomous driving task since there 609 are datasets for this task that contain all types of auxiliary 610 modality in Sec. 4.1.1. Specifically, the input is one RGB 611 image and the output is one float value which represents the 612 steering angle. Classical computer vision tasks, like object 613 classification or detection, mostly do not have datasets with 614 Low-level sensing data other than RGB image (like depth 615 map is not available in ImageNet or COCO). We use Audi 616 dataset (Geyer et al., 2020) and Honda dataset (Ramanishka 617 et al., 2018) in this experiment. Also, we use depth map 618 (Type 1), frequency image (Type 2), and attention image 619 (Type 3) as Auxiliary modalities. We generate depth map with (Bian et al., 2019), frequency image with standard 2D fast Fourier transform (Cochran et al., 1967), and attention image with segmentation map provided by Audi dataset. We implement all four types of auxiliary modality learning architectures introduced in Sec. 4.1.2, and choose the Nvidia PilotNet (Bojarski et al., 2016) and ResNet (He et al., 2016) as the main backbones. Mean accuracy defined in (Shen et al., 2021) is used as the evaluation metric.

Multiple tasks. We use Audi dataset (Geyer et al., 2020) for end-to-end steering task, COCO dataset (Lin et al., 2014) for real-world classification task, and a customized dataset for customized classification task. We use semantic segmentation label contained in Audi and COCO to generate related attention images. We use blur-level estimation task as the customized task, following Shen et al.'s work (Shen et al., 2021) to add blur perturbation onto the Audi dataset, and use the level ID as the ground truth, see Fig. 6. Also, we use attention image and frequency image as auxiliary modalities, and implement all four types of auxiliary modality learning architectures introduced in Sec. 4.1.2. We choose Nvidia PilotNet (Bojarski et al., 2016) for steering task, ResNet (He et al., 2016) for the classification task, and modified PilotNet for the customized classification task (change the header of the network to general classification header). We use mean accuracy (Shen et al., 2021) for steering task, accuracy for real-world classification and customized classification.

#### A.2. Experiment Results for Auxiliary Modality Types and Architectures

We show experimental results for auxiliary modality in Table 5 and architectures in Table 6. See analysis in Sec. 4.1.

#### A.3. Supermodel Example

We first introduce the "supermodel" definition:

**Definition A.1** Given a model  $M_{\theta_A}^{(A)}(I_A)$  (weights  $\theta_A$  and input  $I_A$ ), and a model  $M_{\theta_B}^{(B)}(I_B)$  (weights  $\theta_B$  and input



Figure 5. A simple example of supermodel.  $Net_1$  contains two blocks  $f_1$  and  $f_2$ .  $Net_2$  contains the same block  $f_1$  and  $f_2$ , and another block h which is possible to be set as an identical function.

 $I_B$ ), if for any  $\theta_A$ , there is a  $\theta_B$ , such that  $M_{\theta_A}^{(A)}(I_A) = M_{\theta_B}^{(B)}(I_B)$  for any arbitrary valid input data  $I_A$  and its superset  $I_B$ . We call model  $M_B$  as a "supermodel" of  $M_A$ .

We show a simple example of supermodel in Fig. 5.  $Net_1$  contains two blocks  $f_1$  and  $f_2$ .  $Net_2$  contains the same block  $f_1$  and  $f_2$ , and another block h. If there is a set of specific weights  $\theta_0$  for h that can meet  $h_{\theta_0}(x) = x$  for any valid x, then  $Net_2$  is a supermodel of  $Net_1$ , according to Definition. A.1. In this case, for any specific weights of  $Net_1$ , we can always construct a set of weights for  $Net_2$  that has exactly the same performance of  $Net_1$ , which means the optimal solution for training  $Net_2$  will be no worse than  $Net_1$ . Furthermore, if these two models are training in parallel, the supermodel can be "repositioned" to the same status of the base model at any time by the construction method above. This property can be used in knowledge distillation to let the teacher get back to the student's position and help find a better way at any time the student is stuck.

#### A.4. Prove of Lemma. 4.1

**Lemma A.1** Given a model  $\mathcal{M}$  and its supermodel  $\mathcal{M}^{(s)}$ , the optimal training loss of  $\mathcal{M}^{(s)}$  (which is  $\arg\min_{\theta^{(s)}} L(\mathcal{M}^{(s)}_{\theta^{(s)}}(I^{(s)}), GT))$  is less than or equal to the optimal training loss of  $\mathcal{M}$  (which is  $\arg\min_{\theta} L(M_{\theta}(I), GT)$ ). where L is the loss function and GT is the ground truth.

Prove: Let  $\theta^* = \arg \min_{\theta} L(\mathcal{M}_{\theta}(I), GT)$  represent the weights that lead to the best training performance for model  $\mathcal{M}$ , then according to the definition of supermodel, there is a  $\theta^{(s)*}$  that meet  $\mathcal{M}_{\theta^*}(I) = \mathcal{M}_{\theta^{(s)*}}^{(s)}(I^{(s)})$ , equivalent to  $L(\mathcal{M}_{\theta^*}(I), GT) = L(\mathcal{M}_{\theta^{(s)*}}^{(s)}(I^{(s)}), GT)$ . That is, there's at least one solution for training  $\mathcal{M}^{(s)}$  can get the same performance as training  $\mathcal{M}$ . Furthermore, if  $\theta^*$  is the optimal solution that achieves the minimal training loss of  $\mathcal{M}^{(s)}$ , then the equal condition in Lemma 4.1 holds, if not, the less condition holds.

		Aud	li (Geyer et al., 2020)		Honda (Ramanishka et al., 2018)		
		AttentionFrequencyDepthAttentionFrequencyi Type A66.364.965.874.572.9i Type B71.666.568.475.973.2i Type C70.165.768.874.373.7i Type D73.467.970.877.474.8		Frequency	Depth		
	Archi Type A	66.3	64.9	65.8	74.5	72.9	73.4
PilotNet (Bojarski et al., 2016)	Archi Type B	71.6	66.5	68.4	75.9	73.2	75.1
	Archi Type C	70.1	65.7	68.8	74.3	73.7	74.2
	Archi Type D	73.4	67.9	70.8	77.4	74.8	76.7
	Archi Type A	78.5	77.9	78.2	82.1	81.1	81.9
ResNet (He et al., 2016)	Archi Type B	80.5	79.1	80.1	84.7	82.4	83.9
	Archi Type C	79.6	78.5	79.3	83.6	82.1	83.1
	Archi Type D	82.4	79	81.8	85.2	83	84.3

*Table 5.* Performance improvement comparison (Mean Accuracy %) with different auxiliary modalities, architectures, backbones and datasets. The relative effectiveness for different architectures is consistent under different datasets, backbones, and auxiliary modalities within one task. Similarly, The relative effectiveness for different auxiliary modalities is consistent under different datasets, backbones, and architectures within one task.

	tas	sk 1	tas	sk 2	task 3		
	Attention	Frequency	Attention	Frequency	Attention	Frequency	
Archi Type A	66.3	64.9	70.1	69.3	64.3	65.2	
Archi Type B	71.6	66.5	82.1	73.6	68.4	72.5	
Archi Type C	70.1	65.7	80.7	71.1	65.3	70.8	
Archi Type D	73.4	67.9	84.3	75.6	70.3	74.9	

*Table 6.* Performance comparison (Mean Accuracy %) across tasks. The effectiveness order of different architectures is consistent across tasks, but not for auxiliary modalities.

#### A.5. More Explanation on Why AML Can Work

In Fig. 7, the main modality data has few examples in the hard and challenging case area, which leads to a wrong decision boundary. This is common in real-world datasets, e.g., autonomous driving datasets usually have fewer datasets for night-time driving, and even fewer on accidents. After adding the auxiliary modality that provides more information in the hard case area, it would be much easier to learn a correct decision boundary, then use this information to guide the training process with the main modality. For example, the infra-red image or depth map contains more information than RGB image when captured at night. This explains why the low-level sensing data (Type 1 in Sec. 4.1.1) can help AML.

#### A.6. Modality Choice

We show a modified network to extract channel-level importance and estimate modality effectiveness with SE block in Fig. 8. See more descriptions in Sec. 5.1.

#### A.7. AML in SAMD

Formally, given a task, we denote a learner composed of a feature network F and a predictor of fully-connected layers D. We design a student that takes  $I_M$  as input, and update

via iterations of mini-batches,

$$\theta_{stu} \leftarrow \theta_{stu} - \eta \nabla L^M \tag{1}$$

where  $\theta_{stu}$  is the parameter of the student network,  $L^M$  is the loss function, and  $\eta$  is the learning rate. Meanwhile, we design a teacher that takes  $\{I_M, I_A\}$  as input, and update via an independent feature network  $F_{tea}$  ( $F_1, F_2, F_3$  in Fig. 3) and a predictor D that share weights with that of the student network. The teacher network is updated via

$$\theta_{tea} \leftarrow \theta_{tea} - \eta \nabla L^A \left( D(F_{tea}(\{I_M, I_A\})), GT) \right).$$
(2)

The teacher and student learn different representations related to the same task by being exposed to different modalities. The teacher has access to the auxiliary modality  $I_A$ , the knowledge of the teacher is distilled to assist the student through a consistency loss  $L_{con}$  that measures the pairwise distance between  $F_{stu}(I_M)$  and  $F_{tea}(I_M, I_A)$  as part of the student's objective  $L^M$ , specifically,

$$L^{M} = L_{sup} \left( D(F_{stu}(I_{M})), GT \right) + \beta L_{con} \left( F_{stu}(I_{M}), F_{tea}(\{I_{M}, I_{A}\}) \right)$$
(3)

where  $L_{sup}$  is a term that supervises the learning on the main modality.

**Definition A.2** Given a model  $M_{\theta_A}^{(A)}(I_A)$  (weights  $\theta_A$  and input  $I_A$ ), and its supermodel  $M_{\theta_B}^{(B)}(I_B)$  (weights  $\theta_B$  and



*Figure 6.* Tasks for our experiments. LEFT: end-to-end steering task, input image, output steering angle. MIDDLE: classification task, input image, output object category. RIGHT: classification task, input image, output blur level.



Figure 7. Auxiliary modality helps construct the decision boundary around the difficult cases (e.g. lack of data coverage). Circles are main modality data, and squares are auxiliary modality data.



*Figure 8.* Modified network to extract channel-level importance and estimate modality effectiveness with SE block. (Hu et al., 2018)

input  $I_B$ ), we define "reset B with A" to be the process of constructing a new  $\theta_B$  that meet  $M_{\theta_A}^{(A)}(I_A) = M_{\theta_B}^{(B)}(I_B)$ for given  $\theta_A$  and any arbitrary valid input data  $I_A$  and its superset  $I_B$ .

A simple example is, suppose B is a supermodel of A (e.g., B = A + A'), reset B with A is constructing  $\theta_B = [\theta_A, 0]$ , where  $\theta_A$  is the weights of A and 0 is the weights of A'. In Fig. 3, the teacher network is a supermodel of the student network, because for any weights of student network, we can construct a teacher network that meet  $D(F_{tea}(\{I_M, I_A\})) = D(F_{stu}(\{I_M\}))$  by resetting the  $F_1$  weights with  $F_4$  weights,  $F_2$  weights with  $F_5$  weights, and set  $F_3$  weights to 0. Indeed the reset operation in our method requires that the teacher model is a supermodel of the student model.

As shown in Algorithm 1, the training paradigm contains t rounds. In each round, we first *reset* the teacher with the student, then train the teacher independently while training student with both the general label loss and knowledge distillation loss for k epochs. k should not be too large to avoid the teacher being far away from the student. The training process stops when the student converges between different rounds or until finishing t rounds.

#### A.8. Additional SAMD Results

Setting. All experiments are conducted using one Intel(R) Xeon(TM) W-2123 CPU, two Nvidia GTX 1080 GPUs, and 32G RAM. We use the SGD optimizer with learning rate 0.001 and batch size 128 for training. The number of epochs is 2,000. The loss correlation  $\beta$  is set with different values for different knowledge distillation methods following (Tian et al., 2020). We pick epoch number in each round k = 5

Auxiliary Modality Learning with Generalized Curriculum Distillation

Algorithm 1 SAMD Training Paradigm
<b>Input:</b> Training data from main modality $I_M$ , training
data from auxiliary modality $I_A$ (chosen by method in
Sec. 5.1)
<b>Output:</b> student network weights $\theta_{stu}$
Initialisation:
Training Round number t, epoch number in each round
k, loss correlation $\beta$ , network weights $\theta_{stu}$ and $\theta_{tea}$ .
for $r = 1$ to $t$ do
Reset teacher weights with student weights
for $e = 1$ to $k$ do
Feed $I_M$ and $I_A$ into teacher, update teacher weights
$ heta_{tea}$ with Eq. 2
end for
for $e = 1$ to $k$ do
Feed $I_M$ and $I_A$ into teacher, and feed $I_M$ into stu-
dent, update student weights $\theta_{stu}$ with Eq. 1 and
loss 3
end for
end for

from ablation study of k = 1, 2, 5, 20. We set the round number n = 400 for Audi dataset and n = 40 for Honda dataset. In the experiments, each training process is finished within 24 hours. The main task is the steering task introduced in the single task setting in Appendix A.1.

**Comparison on other tasks.** To show the generalizability of our method, we perform comparison on multi-feature handwritten classification task (Han et al., 2021) in Table 7. The dataset (UCI, 0) consists of six features of handwritten numerals ('0'-'9') with 2,000 samples in total. We regard the six feature sets as six modalities, and treat each of them as target modality in each experiment. Our method outperforms others by 5.1% higher accuracy on average.

	A	ccuracy (	(%) on d	ifferent n	nodalitie	s (ID:1~	6)
Method	1	2	3	4	5	6	mean
Best Others Ours	84.92 <b>89.40</b>	62.98 <b>65.20</b>	68.75 <b>72.80</b>	61.10 <b>69.50</b>	70.35 <b>73.15</b>	43.17 <b>51.75</b>	65.2 <b>70.3</b>

*Table 7.* **Performance comparison on handwritten classification task.** Our method outperforms other KD methods listed in Table 1 by 5.1% higher accuracy *on average*.

We also conducted experiments on another end-to-end autonomous driving task, "way-point prediction" task. Following the setting of (Prakash et al., 2021), we consider the task of navigation along a set of predefined routes in different areas, such as motorways, urban regions, and residential districts. A sequence of sparse goal locations in GPS coordinates, provided by a global planner and the related discrete navigational commands (e.g. "follow lane", "turn left/right", and "change lane"), constitute the routes. Only

Model	DS↑	RC↑	IP↓	CP↓	CV↓	CL↓	RLI↓	SSI↓
RGB	21.0	60.5	0.49	0.01	0.15	0.08	0.14	0.04
RGB+PC	11.2	52.9	0.37	0.02	0.22	0.01	0.38	0.02
Ours(new)	22.1	79.5	0.37	0.01	0.07	0.04	0.26	0.04

Table 8. Performance comparison on long-route waypoints prediction between base (train and test on RGB), multi-modality (train and test on RGB + point cloud), and ours (train on RBG + point cloud, test using *only RGB*). DS: Avg. driving score, RC: Avg. route completion, IP: Avg. infraction penalty, CP: Collisions with pedestrians, CV: Collisions with vehicles, CL: Collisions with layout, RLI: Red lights infractions, SSI: Stop sign infractions.

the sparse GPS locations are used in our method. Each route consists of several scenarios, which are initialized at predefined locations and test the agent's ability to handle various adversarial situations, such as obstacle avoidance, unprotected turns at intersections, vehicles running red lights, and pedestrians emerging from occluded regions crossing the road at random locations. The agent needs to complete the route within a certain amount of time, while following traffic regulations and dealing with large numbers of dynamic agents. For dataset, we use the CARLA (Dosovitskiy et al., 2017) simulator for training and testing, specifically CARLA 0.9.10 that include 8 publicly available towns. We use 7 towns for training and hold out Town05 for evaluation, as in (Prakash et al., 2021). We use both RGB and LiDAR for training in AML, but only RGB data for testing. The results are shown in Table 8. Our method benefits from the auxillary LiDAR modality in training using AML, with only RGB data during query. This set of experimental results demonstrates the effectiveness of AML.

In addition, we apply our method on audio modality based on an audio-visual depth and material estimation work (Wilson et al., 2022). We use RGB image as the main modality, and audio wave as the auxiliary modality. The task is material and depth classification. We use the same dataset in the original audio-visual work, which contains about 16,000 pairs of RGB image and audio wave. Since there's no open-source code, we reimplement the original work, then apply our method to it. Our method outperforms other KD methods listed in Table 1 by 6.4%.

Finally, we apply our method on a bird-eye-view segmentation task (Li et al., 2022). During training, a mixed point cloud from multiple viewpoints is used as input, while a point cloud from one viewpoint is used during test. We use the same virtual autonomous driving dataset (Li et al., 2022), which contains 48,000 datapoints for training, 6,000 datapoints for test, and 6,000 datapoints. We apply our method based on the DiscoNet (Li et al., 2021). In Table 10, we show our method achieves the best performance compared to other methods.

Comparison on different datasets and modalities. We

Auxiliary Modality Learning with Generalized Curriculum Distillation

Accuracy (%) on various angle threshold $ au$ (degree)										
Method	$\mid \tau = 1.5$	$\tau=3.0$	$\tau=7.5$	$\tau = 15$	$\tau=75$	mAcc				
Seg GT	50.6	70.9	85.4	96.1	99.2	80.44				
Seg Infer	48.3	69.5	85.3	95.7	98.6	79.48				

Table 9. Performance comparison between ground truth and generated segmentation. The results show that the inferred segmentation can do nearly as well as ground truth segmentation, when serving as auxiliary modality (within 1% of difference). Therefore, we can use pre-trained models to generate auxiliary modality conveniently.

838 839 also perform comparison with other knowledge distillation methods on different datasets (Audi (Geyer et al., 2020), 840 Honda (Ramanishka et al., 2018), and SullyChen (Chen, 841 2018)) and different modalities (RGB, segmentation, depth 842 843 map, and edge map). Specifically, Audi dataset contains ground truth segmentation, and other segmentation is gen-844 845 erated by Tao et al. (Tao et al., 2020), while the depth map is generated by (Bian et al., 2019) and the edge map is gen-846 erated by DexiNet (Poma et al., 2020). In Table 12, our 847 method outperform others in practically all cases by up to 848 849 +11% accuracy improvement.

850 Effectiveness when combining with different knowledge 851 distillation methods. Since our training paradigm can 852 be applied on existing knowledge distillation methods, we 853 do experiments by combining ours with kd (Hinton et al., 854 2015), hint (Romero et al., 2015), similarity (Tung & Mori, 855 2019), correlation (Peng et al., 2019), rkd (Park et al., 2019), 856 pkt (Passalis et al., 2020), abound (Heo et al., 2019), fac-857 tor (Kim et al., 2018b), fsp (Yim et al., 2017). From Ta-858 ble. 13, our method achieves up to 15.6% improvement 859 in both settings, showing the effectiveness of our training 860 paradigm (containing reset operation). 861

**Relation of Channel-level Importance and AML Perfor-**862 mance. To show the channel-level attention for different 863 auxiliary modalities is positively correlated to the final per-864 formance of AML with different auxiliary modalities, we 865 conduct experiments on three tasks with the same setting 866 stated in Sec. 4.1.3, then use the same three auxiliary modal-867 ities and an additional random noise modality (whose impor-868 tance should be the lowest). We use knowledge distillation 869 based architecture (Type D), since it's consistently better 870 than other architectures (see Sec. 4.1.3). 871

As shown in Table 11, in Task 1, the importance order from the channel-level attention is attention image > depth map
> frequency image, and the performance order from AML is also attention image > depth map > frequency image.
The same phenomenon can be observed in Task 2 and 3. This shows we only need to perform one-time training to select the best modality for a given task. **Comparison of ground truth and generated auxiliary modality.** We conduct experiment with ground truth segmentation and generated segmentation (Tao et al., 2020) to see how much it will influence the performance. The model used to generate segmentation for Audi dataset (Geyer et al., 2020) is trained on Cityscapes dataset (Cordts et al., 2016). Table 9 shows that the generated segmentation can do nearly as well as ground truth segmentation, when serving as auxiliary modality (i.e. within 1% of difference), thus we can use pre-trained models to generate auxiliary modality conveniently.

Auxiliary Modality Learning with Generalized Curriculum Distillation

Method	Vehicle	Sidewalk	Terrain	Road	Building	Pedestrian	Vegetation	mIoU
Lower-bound	45.93	42.39	47.03	65.76	25.38	20.59	35.83	40.42
Co-lower-bound	47.67	48.79	50.92	70	25.26	10.78	39.46	41.84
When2com (Liu et al., 2020a)	48.43	33.06	36.89	57.74	29.2	20.37	39.17	37.84
Who2com (Liu et al., 2020b)	48.4	32.76	36.04	57.51	29.17	20.36	39.08	37.62
DiscoNet (Li et al., 2021)	<b>56.66</b>	46.98	<b>50.22</b>	<b>68.62</b>	27.36	22.02	42.5	44.91
Ours	56.52	<b>47.43</b>	49.72	67.72	<b>30.59</b>	<b>22.23</b>	<b>42.86</b>	<b>45.30</b>
Upper-bound	64.09	41.34	48.2	67.05	29.07	31.54	45.04	46.62

Table 10. Performance comparison on bird-eye-view segmentation task. Our method achieves the best performance compared to three other methods, with only 1.32% performance difference from the upper-bound. We follow the same setting of (Li et al., 2021) for the lower-bound, co-lower-bound and upper-bound.

	C			AML Perform	nance			
	Attention	Frequency	Depth	Noise	Attention	Frequency	Depth	Noise
Task 1	0.32	0.08	0.12	6.9e-6	73.4	67.9	70.8	65.2
Task 2	0.65	0.12	-	2.7e-6	84.3	75.6	-	70.1
Task 3	0.09	0.26	-	3.2e-6	70.3	74.9	-	60.8

Table 11. Relation between relative orders of channel-level importance and AML performance for different auxiliary modalities. The relative modality orders are consistent between channel-level importance and AML performance within each task, therefore we can use channel-level importance to choose the best auxiliary modality before AML.

				Accuracy (%) on different angle threshold $\tau$ (degree)							
Dataset	Train Mod	Test Mod	Method	$\tau = 1.5$	$\tau = 3.0$	$\tau = 7.5$	$\tau = 15$	$\tau = 30$	$\tau = 75$	mAcc	
Audi	RGB+seg	RGB+seg	Teacher	42.7	68.0	88.0	94.4	96.6	98.6	81.4	
Audi	RGB+seg RGB+seg	RGB RGB	best others ours	30.3 <b>52.6</b>	51.0 <b>72.7</b>	78.2 <b>91.3</b>	88.4 <b>95.0</b>	94.4 <b>97.0</b>	98.2 <b>98.3</b>	73.4 <b>84.5</b>	
Audi	RSDE	RSDE	Teacher	49.9	72.1	89.5	94.9	97.1	98.6	83.7	
Audi	RSDE RSDE	RGB RGB	best others ours	27.7 <b>30.2</b>	47.8 <b>50.3</b>	77.4 <b>79.7</b>	90.8 <b>91.0</b>	95.6 <b>96.2</b>	98.3 <b>98.6</b>	72.9 <b>74.3</b>	
SullyChen	RDE	RDE	Teacher	41.1	63.7	88.6	95.9	97.9	99.1	81.0	
SullyChen	RDE RDE	RGB RGB	best others ours	59.5 <b>63.4</b>	82.1 <b>83.0</b>	93.9 <b>94.3</b>	98.2 98.2	99.5 99.5	100.0 100.0	88.9 <b>89.7</b>	
Honda	RSDE	RSDE	Teacher	41.3	61.1	83.9	94.0	98.3	99.9	79.8	
Honda	RSDE RSDE	RGB RGB	best others ours	<b>38.9</b> 37.9	57.7 57.7	79.7 <b>81.7</b>	91.7 <b>93.5</b>	97.5 <b>98.2</b>	99.3 <b>99.6</b>	77.4 <b>78.1</b>	

Table 12. Comparison on different datasets and different modalities. "RSDE" refers to results from RGB + segmentation + depth map + edge map, and "RDE" for RGB + depth map + edge map. Our method outperforms others on different datasets and different additional modalities by up to +22.3% accuracy improvement. 

Accuracy on different threshold $\tau$ (%)												
Method	$\mid \tau = 1.5$	$\tau = 3.0$	$\mid \tau = 7.5$	$\mid \tau = 15$	$\mid \tau = 30$	$\tau = 75$	Mean	Improvement				
Train Vanilla												
Teacher (img+seg)	42.7	68.0	88.0	94.4	96.6	98.6	81.4					
Student (img)	27.3	49.0	77.4	90.2	95.4	98.1	72.9					
	Existin	ng Distillat	ion Metho	ds								
kd (Hinton et al., 2015)	28.4	47.7	73.2	87.2	94.3	98.4	71.5					
hint (Romero et al., 2015)	31.7	50.2	69.5	77.0	83.7	93.8	67.6					
similarity (Tung & Mori, 2019)	33.0	55.9	80.8	90.5	95.1	98.3	75.6					
correlation (Peng et al., 2019)	36.2	59.1	81.5	91.7	95.3	98.2	77.0					
rkd (Park et al., 2019)	32.9	53.6	80.3	91.8	96.2	98.5	75.6					
pkt (Passalis et al., 2020)	34.2	55.4	80.8	90.4	94.9	98.5	75.7					
vid (Ahn et al., 2019)	49.7	71.2	89.9	94.8	96.7	98.3	83.4					
abound (Heo et al., 2019)	32.8	53.9	77.8	88.9	94.6	98.0	74.3					
factor (Kim et al., 2018b)	36.8	59.2	82.0	90.6	94.7	97.9	76.9					
fsp (Yim et al., 2017)	30.8	51.6	74.9	85.8	91.6	97.4	72.0					
Existing Di	stillation N	Aethods w	ith Our Tr	aining Pa	radigm							
kd (Hinton et al., 2015)	49.7	71.2	89.9	94.8	96.7	98.3	83.4	11.9				
hint (Romero et al., 2015)	48.6	71.0	90.1	94.8	96.7	98.3	83.2	15.6				
similarity (Tung & Mori, 2019)	52.1	71.8	90.0	94.8	96.6	98.3	83.9	8.3				
correlation (Peng et al., 2019)	31.8	52.7	78.1	89.7	95.2	98.3	74.3	-2.7				
rkd (Park et al., 2019)	54.3	72.2	90.1	94.7	96.6	98.3	84.4	8.8				
pkt (Passalis et al., 2020)	34.5	56.9	82.9	90.3	95.5	98.4	76.4	0.7				
vid (Ahn et al., 2019)	48.6	71.0	90.1	94.8	96.7	98.3	83.2	-0.2				
abound (Heo et al., 2019)	29.6	49.5	74.4	87.3	93.5	97.8	72.0	-2.3				
factor (Kim et al., 2018b)	49.7	71.2	89.9	94.8	96.7	98.3	83.4	6.5				
fsp (Yim et al., 2017)	28.8	48.2	71.5	83.9	91.2	97.4	70.1	-1.9				
						1						

Table 13. Performance comparison with vs. without our training paradigm (containing reset operation). By applying our training paradigm on other knowledge distillation methods, we can achieve better performance in most cases (up to +15.6%) in either fully paired or merely a small amount of additional modality data.