Ethics Statement: No identifiable human data is collected.

Reproducibility: Implementation detail is given in the paper and below, with datasets and code to be released upon publication.

Appendix

A. Supermodel Example

We first introduce the "supermodel" definition:

Definition 1.1: Given a model $M^{(A)}_{\theta_A}(I_A)$ (weights θ_A and input I_A), and a model $M^{(B)}_{\theta_B}(I_B)$ (weights θ_B and input I_B), if for any θ_A, there is a θ_B, such that $M^{(A)}_{\theta_A}(I_A) = M^{(B)}_{\theta_B}(I_B)$ 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. 3. 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 θ_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. 1.1. 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.

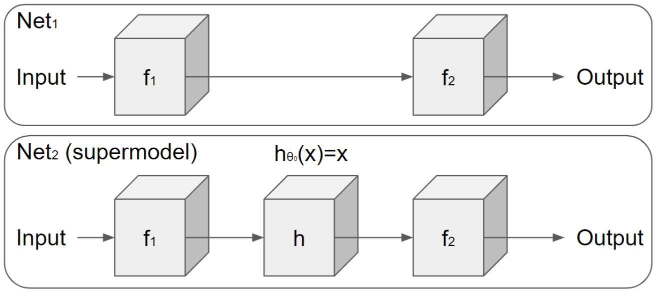

Fig. 3. A simple example of supermodel. Net_1 contains two blocks f_1 and f_2. Net_2 contains the same block f_1 and f_2, and another block h which is possible to be set as an identical function.

B. Implementation Details

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 correlations are $\alpha = 1, \gamma = 1$, while β are set with different values for different knowledge distillation methods following [14] (See details in Appendix I), and $\lambda = \beta/10$. We pick epoch number in each round $k = 5$ 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.

Evaluation metric. We use the same evaluation metric as a lastest work [37], i.e., the accuracy w.r.t a threshold τ as $acc_\tau = count(|v_{predicted} - v_{actual}| < \tau)/n$, where n denotes the number of test cases; $v_{predicted}$ and v_{actual} indicate the predicted and ground-truth value, respectively. We compute mean accuracy (mAcc) as $\sum_\tau acc_{\tau \in \mathcal{T}}/|\mathcal{T}|$, where $\mathcal{T} = \{1.5, 3.0, 7.5, 15, 30, 75\}$ contains empirically selected thresholds of steering angles.

Dataset. For the end-to-end steering task, we do experiments on Audi and Honda datasets. The Audi dataset [38] is the most recent (2020). We use the semantic segmentation subset since it contains both steering angle from bus data and semantic segmentation labels paired with RGB images, which can be used as an additional modality in our method. It contains 41,277 frames in total. The Honda dataset [33] has 100+ long-time driving videos, which is one of the largest autonomous driving datasets. We extract 110k images with 1Hz from the original videos and split them into 100k training images and 10k test images.

Backbone. We choose the Nvidia PilotNet described in [30] as the main backbone. We select this model as it has been used to steer an autonomous vehicle successfully in both the real world [30] and virtual world [49], and also work for the latest autonomous driving datasets [47]. In addition, four other networks are tested to show generalizability.

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.

C. Simple Experiment

We introduce a lemma on the optimal training loss of the supermodel and its base model.

Lemma 1.1: Given a model M and its supermodel $M^{(s)}$, the optimal training loss of $M^{(s)}$ (which is $\arg\min_{\theta^{(s)}} L(M^{(s)}_{\theta^{(s)}}(I^{(s)}), GT)$) is less than or equal to the optimal training loss of 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 $\theta^* = \arg\min_\theta L(M_\theta(I), GT)$ represent the weights that lead to the best training performance for model M, then according to the definition of supermodel, there is a $\theta^{(s)*}$ that meet $M_{\theta^*}(I) = M^{(s)}_{\theta^{(s)*}}(I^{(s)})$, equivalent to $L(M_{\theta^*}(I), GT) = L(M^{(s)}_{\theta^{(s)*}}(I^{(s)}), GT)$. That is, there's at least one solution for training $M^{(s)}$ can get the same performance as training M. Furthermore, if θ^* is the optimal solution that achieves the minimal training loss of $M^{(s)}$, then the equal condition in Lemma. 1.1 holds, if not, the less condition holds.

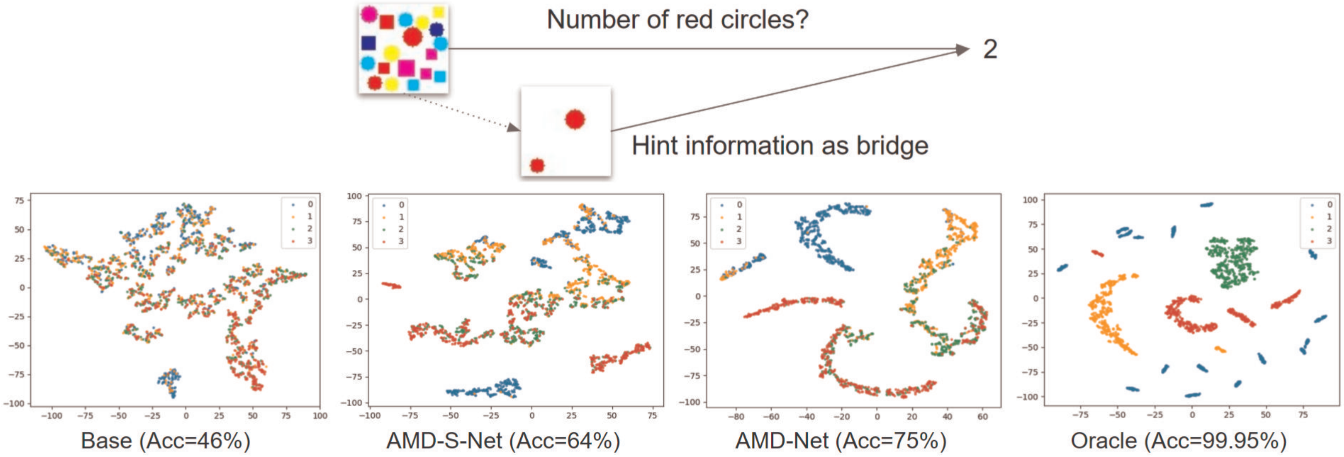

Fig. 4. Simple experiment explanation. TOP: task definition, counting *the number of red circles* in an image of arbitrary shapes of varying colors, with the images that only contain red circles as the auxiliary modality. BOTTOM: t-SNE visualization for the features generated from networks trained by our methods. The base features are mixed together thus can not be well classified, while our features can be grouped better than the base ones (less mixing points). The oracle features have almost no mixing up in each group because the model has sufficient training data (10x as the base and ours).

D. Simple Experiment

We consider a simple task of counting *the number of red circles* in an image of arbitrary shapes of varying colors. In this case, the main modality I^M is the image containing a random layout of arbitrary shapes. We create an auxiliary modality I^A as the images that only contain *red circles*, as shown in Fig. 4. For the experiment, we generate 4000 image samples of the main modality with ground-truths $\{i_n^M, y_n^M\} \in \mathcal{I}_{train}^M$, and test on another 2000 randomly generated layouts for $\mathcal{I}_{test}^M$. For the auxiliary modality I^A we generate 4000 samples $\{i_n^A\} \in \mathcal{I}^A$. To confirm the hypothesis that the knowledge of I^A can be distilled to improve the task on I^M, we design an ablation study comparing the following baselines:

- Oracle $\mathcal{I}^M$: $\mathcal{I}_{train}^M$ is sufficient for θ^{M*};
- Underfitted $\mathcal{I}^M$: $\mathcal{I}_{train}^M$ is not sufficient for θ^{M*};
- Underfitted $\mathcal{I}^M$ plus auxiliary $\mathcal{I}^A$: Add auxiliary samples from $\mathcal{I}^A$ to insufficient $\mathcal{I}^M$.
- Underfitted $\mathcal{I}^M$ plus insufficient auxiliary $\mathcal{I}^A$: Add auxiliary but a small number of samples from $\mathcal{I}^A$ to insufficient $\mathcal{I}^M$.

For the *Oracle* experiment, all 4,000 training samples are used, the model is nearly perfect at inference, achieving 99.95% accuracy on $\mathcal{I}_{test}^M$. In an *Underfitted* situation, we randomly sample 10% from $\mathcal{I}_{train}^M$ and have created an insufficiently trained classifier with 46% accuracy. After that, in addition to the *Underfitted* model we add 400 auxiliary samples from $\mathcal{I}^A$, the accuracy improves to 75% (29 percent improvement). Even without sufficient auxiliary samples, when we select merely 80 samples from $\mathcal{I}^A$ and use AMD-S-Net, we still witness a plausible improvement to 64% (18 percent improvement). We generate the t-SNE visualizations for the representations from each baseline in Fig. 4, and observe clearly enhanced clusters with the distilled knowledge from auxiliary modality achieved by our methods.

E. Modifications on SOTA Frameworks

We compare our framework with 2 straightforward frameworks and 2 modified frameworks based on SOTA modality distillation methods. We use 100% RGB images and 20% segmentation data in this experiment. Specifically, the one stream (RGB only) method uses 100% RGB images only with the student network. Two streams (shared regressor) method contains RGB and segmentation pipelines with a feature extractor for each pipeline and a shared regressor. The total loss is the sum of RGB loss and segmentation loss. During test, only the RGB pipeline is used. For the modified [1] and modified DMCL [26], we keep the 20% paired RGB and segmentation to go through the original pipeline with backpropagation, and let the rest 80% RGB data go through a single RGB pipeline with backpropagation. During test, only the RGB pipeline is used.

F. Comparison with SOTA Knowledge Distillation Methods

We show the detailed comparison data our AMD-S-Net in Table X. With our method, the performance improvement can get up to 18.1%.

G. Multi-Modal End-to-End Waypoint Prediction

To show the generalizability of our method, we do experiments on another end-to-end autonomous driving task, way points prediction task. Following the setting of [50], 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, such as “follow lane”, “turn left/right”, and “change lane”, constitute the routes. Only the sparse GPS locations are used in our method. Each route is constituted of several scenarios that 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 areas crossing the road at random locations. The agent needs to complete the route within a certain amount of time, while following traffic restrictions and dealing with large numbers of dynamic agents. For dataset, we use the CARLA [51] simulator for training and testing, specifically CARLA 0.9.10 which consists of 8 publicly available towns. We use 7 towns for training and hold out Town05 for evaluation as in [50]. See Table. III.

Model	DS$\uparrow$	RC$\uparrow$	IP$\downarrow$	CP$\downarrow$	CV$\downarrow$	CL$\downarrow$	RLI$\downarrow$	SSI$\downarrow$
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	**22.0**	**63.1**	0.45	0.02	**0.05**	**0.00**	0.20	0.03

TABLE III

Performance comparison on long routes way points prediction between base (100% RGB), multi-modality (28% RGB + 28% point cloud), and our method (100% RBG + 28% point cloud). 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.

H. Handwriting Classification

We also perform comparison on multi-feature handwritten classification task [52] in Table. IV. The dataset [53], [54], [55] consists of six features of handwritten numerals ('0'–'9') with 2000 samples in total. We regard the six feature sets as six modalities, and treat each of them as target modality in each experiment. Teacher network is able to get all 6 modalities (but only 20% amount of data). During test, only one target modality is available. Our method outperforms others with 5.1% in average.

	Accuracy (%) on different modalities (ID:1∼6)						
Method	1	2	3	4	5	6	mean
Other KD	84.92	**62.98**	68.75	61.10	70.35	43.17	65.2
Ours	**87.42**	62.29	**70.86**	**66.34**	**71.97**	**49.49**	**68.1**

TABLE IV

Performance comparison on handwritten classification task. Our method outperforms other KD methods with 5.1% on average.

I. Knowledge Distillation Methods Settings

For different knowledge distillation methods, different values of β (weight of the consistency loss) is used. We use the same setting as [14]. Specifically:

- kd [4]: $\beta = 0$
- hint [11]: $\beta = 100$
- similarity [13]: $\beta = 3000$
- correlation [15]: $\beta = 0.02$
- rkd [16]: $\beta = 1$
- pkt [9]: $\beta = 30000$
- abound [10]: $\beta = 1$
- factor [8]: $\beta = 200$
- fsp [6]: $\beta = 50$
- attention [7]: $\beta = 1000$

J. Comparison on Different Datasets and Modalities

We also do comparison with other knowledge distillation methods on different datasets (Audi [38], Honda [33], and SullyChen [39]) 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. [40], while the depth map is generated by [41] and the edge map is generated by DexiNet [42]. In Table. V, our method outperform others in all cases with up to 11% accuracy improvement.

K. Comparison on different backbones.

Except for the Nvidia PilotNet [30], we change the backbone to four other backbones, ResNet [43], ShuffleV2 [44], MobileNetV2 [45], and WRN [46], and do comparison in Table. VI. Our method outperforms other methods in all the cases with up to 18.1% accuracy improvement.

L. Comparison on different tasks.

M. Effectiveness on Different Modalities

We do comparison on different types of auxiliary modalities on Audi dataset in Table. VII, with a basic L2-norm feature loss for the knowledge distillation process. We show that all the auxiliary modalities can perform better than the base model by at least 2.8%. This shows our algorithm can utilize different types of auxiliary modalities well, even with a basic knowledge distillation loss.

N. Robustness

We test the robustness of our distilled model following a SOTA work [47] on clean and perturbed Audi dataset (generated with ImageNet-C effects [48]). Table. VIII shows our method can also improve the robustness while not seen any of the perturbed images.

O. Random auxiliary data

When use random noisy as auxiliary modality, our method will not be affected, see Table. IX.

				Accuracy (%) on different angle threshold τ (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	best others	30.3	51.0	78.2	88.4	94.4	98.2	73.4
	RGB+seg	RGB	ours	**52.6**	**72.7**	**91.3**	**95.0**	**97.0**	**98.3**	**84.5**
Audi	RSDE	RSDE	Teacher	49.9	72.1	89.5	94.9	97.1	98.6	83.7
Audi	RSDE	RGB	best others	27.7	47.8	77.4	90.8	95.6	98.3	72.9
	RSDE	RGB	ours	**30.2**	**50.3**	**79.7**	**91.0**	**96.2**	**98.6**	**74.3**
SullyChen	RDE	RDE	Teacher	41.1	63.7	88.6	95.9	97.9	99.1	81.0
SullyChen	RDE	RGB	best others	59.5	82.1	93.9	98.2	99.5	100.0	88.9
	RDE	RGB	ours	**63.4**	**83.0**	**94.3**	98.2	99.5	100.0	**89.7**
Honda	RSDE	RSDE	Teacher	41.3	61.1	83.9	94.0	98.3	99.9	79.8
Honda	RSDE	RGB	best others	**38.9**	**57.7**	79.7	91.7	97.5	99.3	77.4
	RSDE	RGB	ours	37.9	**57.7**	**81.7**	**93.5**	**98.2**	**99.6**	**78.1**

TABLE V

Comparison on different datasets and different modalities. RSDE is short for RGB + segmentation + depth map + edge map, and RDE is short for RGB + depth map + edge map. Our method outperforms others on different datasets and different additional modalities with up to 11% accuracy improvement.

		Accuracy (%) on various angle threshold τ (degree)				
Backbone	Method	$\tau = 1.5$	$\tau = 3.0$	$\tau = 7.5$	$\tau = 15$	**mAcc**
PilotNet	SIM	20.6	38.9	66.7	81.5	66.4
PilotNet	SIM+ours	**52.6**	**72.7**	**91.3**	**95.0**	**84.5**
ResNet34	SIM	30.1	54.4	85.5	**94.1**	76.6
ResNet34	SIM+ours	**37.2**	**60.2**	**85.7**	93.3	**78.6**
ShuffleV2	SIM	39.9	61.3	81.4	89.8	77.7
ShuffleV2	SIM+ours	**47.0**	**71.2**	**90.1**	**94.9**	**83.0**
MobileNetV2	SIM	31.1	51.4	78.2	89.4	73.9
MobileNetV2	SIM+ours	**52.9**	**71.8**	**89.7**	**94.6**	**84.0**
WRN	SIM	22.8	42.9	76.9	92.2	71.7
WRN	SIM+ours	**37.7**	**64.7**	**89.8**	**94.6**	**80.3**

TABLE VI

Performance comparison on different backbones. Our method outperforms SIM [13] on PilotNet [30], ResNet34 [43], ShuffleV2 [44], MobileNetV2 [45], and WRN [46] with up to 18.1% accuracy improvement.

	Accuracy (%) on various angle threshold τ (degree)				
Type of I^A	$\tau = 1.5$	$\tau = 3.0$	$\tau = 7.5$	$\tau = 15$	**mAcc**
Base (No I^A)	28.3	49.5	79.7	89.1	73.1
Depth map [41]	33.3	57.6	81.5	90.4	75.9
Edge map [42]	34.9	56.2	79.6	90.9	76.0
Segmentation [40]	**35.2**	**58.3**	**83.3**	**91.6**	**77.1**

TABLE VII

Comparison on different types of auxiliary modalities on Audi dataset, with a basic L2-norm feature loss for the knowledge distillation process. We show that all the auxiliary modalities can perform better than the base model by at least 2.8%. This shows our algorithm can utilize different types of auxiliary modalities well, even with a basic knowledge distillation loss.

	Clean	Blur				Noise			
	Clean	Defocus	Glass	Motion	Zoom	Gauss	Shot	Impulse	
RGB only	73.1	72.7	71.8	69.8	72.3	67.9	66.9	67.0	
20%$\mathcal{I}^A$	74.8	74.3	73.1	73.2	74.2	69.2	68.3	68.6	
100%$\mathcal{I}^A$	77.1	75.5	75.2	73.1	76.3	71.4	70.1	70.3	
	Clean	Weather				Digital			**mAcc**
	Clean	Snow	Frost	Fog	Bright	Contrast	Pixel	JPEG	**mAcc**
RGB only	73.1	62.8	56.5	54.2	64.2	39.9	73.3	70.7	**65**
20%$\mathcal{I}^A$	74.8	68.1	65.4	63.8	67.6	65.4	74.8	71.8	**69.8**
100%$\mathcal{I}^A$	77.1	63.8	58.7	56.4	65.8	62.0	77.2	75.3	**69.4**

TABLE VIII

AVERAGE ACCURACY(%) OF OUR METHOD ON CLEAN AND PERTURBED DATA (GENERATED WITH IMAGENET-C EFFECTS [48]). THE LAST COLUMN "MEAN" IS THE MEAN ACCURACY ON ALL PERTURBED DATA (BLUR, NOISE, WEATHER AND DIGITAL). WE SHOW THAT BOTH BASIC AND SMALL-SHOT AUXILIARY MODALITY LEARNING CAN GET HIGHER ACCURACY THAN THE BASE METHOD (ABOUT 4.7% IN AVERAGE), I.E., HIGHER ROBUSTNESS.

	Accuracy (%) on different threshold τ (degree)						
Input	$\tau = 1.5$	$\tau = 3.0$	$\tau = 7.5$	$\tau = 15$	$\tau = 30$	$\tau = 75$	**Mean**
RGB	32.4	53.2	78.7	87.7	94.1	97.8	74.0
RGB + 3 Rand	30.3	51.7	79.8	88.4	94.4	97.5	73.7

TABLE IX

RGB IMAGE PLUS 3 RANDOM CHANNELS AS INPUT CAN PERFORM NEARLY AS WELL AS ONLY RGB IMAGE AS INPUT, SHOWING ADDING USELESS CHANNELS WILL NOT INFLUENCE THE PERFORMANCE TOO MUCH.

	Accuracy on different threshold τ (%)							
Method	$\tau = 1.5$	$\tau = 3.0$	$\tau = 7.5$	$\tau = 15$	$\tau = 30$	$\tau = 75$	**Mean**	**Improvement**
train vanilla								
Teacher (img+seg)	40.8	64.1	84.7	92.7	95.8	98.2	79.4	
Student (img)	27.3	49.0	77.4	90.2	95.4	98.1	72.9	
existing distillation methods								
kd [4]	23.4	41.2	68.9	83.7	92.1	97.2	67.7	
hint [11]	28.3	47.6	77.8	89.2	95.0	98.4	72.7	
similarity [13]	20.6	38.9	66.7	81.5	92.6	98.0	66.4	
correlation [15]	21.7	39.5	70.0	86.8	94.6	98.2	68.5	
rkd [16]	26.2	46.5	74.8	87.9	94.1	97.8	71.2	
pkt [9]	30.3	51.0	78.2	88.4	94.4	98.2	73.4	
abound [10]	24.8	45.2	74.9	87.3	93.7	97.7	70.6	
factor [8]	26.8	47.8	76.9	88.8	94.7	98.0	72.2	
fsp [6]	27.1	47.7	74.4	87.9	94.4	97.8	71.6	
attention [7]	27.1	47.0	73.1	84.9	92.8	98.3	70.5	
existing distillation methods with our training paradigm								
kd [4]	30.4	53.7	78.5	88.3	94.8	97.8	73.9	**6.2**
hint [11]	52.7	71.2	88.8	93.6	95.5	97.1	83.1	**10.4**
similarity [13]	52.6	72.7	91.3	95.0	97.0	98.3	84.5	**18.1**
correlation [15]	21.7	39.7	71.2	87.0	94.4	98.2	68.7	**0.2**
rkd [16]	32.4	53.8	79.5	89.3	94.7	97.9	74.6	**3.4**
pkt [9]	54.2	72.5	90.0	94.8	96.7	98.3	84.4	**11**
abound [10]	24.9	45.3	75.1	87.1	93.5	97.7	70.6	0
factor [8]	54.3	72.3	90.1	94.8	96.7	98.3	84.4	**12.2**
fsp [6]	27.5	48.4	75.0	87.5	94.3	97.8	71.8	**0.2**
attention [7]	46.2	68.1	86.8	93.4	96.6	98.2	81.5	**11**

TABLE X

COMPARISON WITH KNOWLEDGE DISTILLATION METHODS ON AUDI DATASET (100% RGB IMAGE + 20% SEGMENTATION) WITH NVIDIA PILOTNET [30]. FIRST SECTION IN THE TABLE SHOWS THE PERFORMANCE OF TEACHER AND STUDENT NETWORK TRAINED DIRECTLY. SECOND SECTION SHOWS THE PERFORMANCE OF STUDENT WITH DIFFERENT KNOWLEDGE DISTILLATION METHODS (TRAIN STUDENT FROM START, USING THE PRETRAINED TEACHER MODEL IN THE PREVIOUS SECTION). THIRD SECTION SHOWS THE PERFORMANCE OF STUDENT AFTER USING OUR TECHNIQUE BASED ON OTHER METHODS (TAKE THE TEACHER AND STUDENT NETWORK IN THE SECOND SECTION OF THIS TABLE AS INIT MODEL, AND RETRAIN THE MODEL WITH OUR METHOD). BY COMPARING BETWEEN THE SECOND AND THIRD SECTION, WE CAN SEE OUR METHOD INCREASE THE PERFORMANCE OF MOST EXISTING METHODS WITH UP TO 18.1%.