

清華大學 Tsinghua University

交叉信息研究院
Institute for Interdisciplinary Information Sciences

脑与智能实验室
ArChip Lab
Algorithm ARchitecture & Chipsets

Knowledge Self-distillation and Scalable Neural Networks: Towards Accurate, Efficient and Robust Models

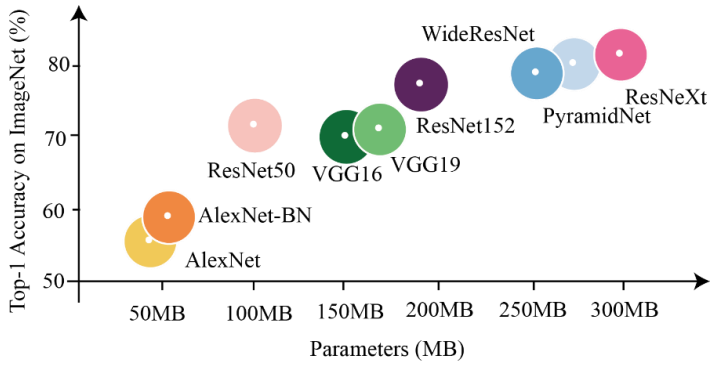
Kaisheng Ma
IIIS, Tsinghua University

Deep Collaboration with
Chenglong Bao @ YMSC THU
Jingwei Chen @ HiSilicon

kaisheng@tsinghua.edu.cn

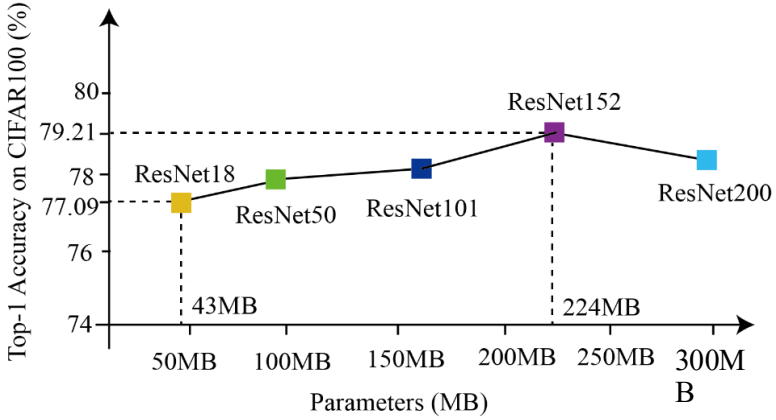
May 5, 2025

Challenge1： More Parameters and Computation



Different Architectures

- Advanced neural networks (e.g. ResNeXt) contains a large amount of parameters.



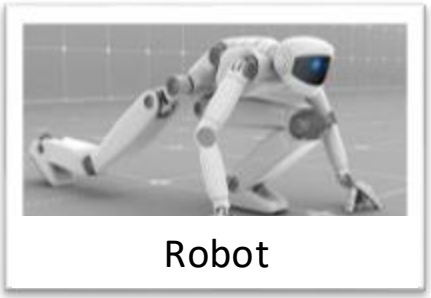
Same Architecture

- Deeper neural networks always achieve better accuracy at the expense of more parameters

Generally speaking, models with **higher accuracy has more parameters and computation.**

❖He K. *et al.* Deep residual learning for image recognition.

Challenge2： Limited Computation Resource



Where we train the model

- **Unlimited** Computation resource
- **Sufficient** Power supply.
- **Tolerant** of long response time.

Where we deploy the model

- **Limited** Computation resource
- **Little** Power supply.
- **Intolerant** of long response time.

Towards Accurate, Efficient and Robust Models

- The training process of deep neural networks can be formulated as

$$\min_{\theta} \|f(x,\theta) - \mathrm{Y}\|$$ f *Model,* θ *weights,* x *Input,* Y *labels.*

- Refine the x and Y in neural networks training towards accurate, efficient and robust models.

WX → Y

- Training: Given X and Y, find a good W to make WX→Y.
- Inference: Compute WX, check Y.

Towards Accurate, Efficient and Robust Models

WX → Y : Self-Distillation, ICCV2019

WX → Y : Scalable Neural Networks, NeuralPS 2019

WX → Y : ADA: Augment Data Augmentation

WX ----> Y : Auxiliary Training

WX → Y : Data Calibrator

Towards Accurate, Efficient and Robust Models

WX → Y : Self-Distillation, ICCV2019

WX → Y : Scalable Neural Networks, NeuralPS 2019

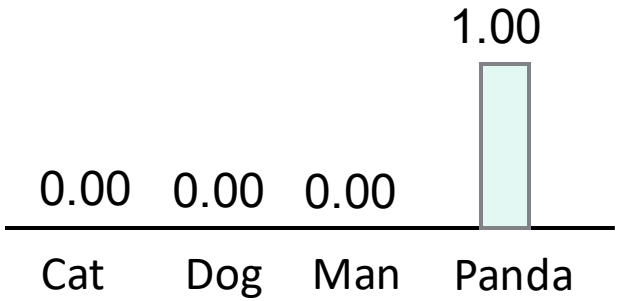
WX → Y : ADA: Augment Data Augmentation

WX ----> Y : Auxiliary Training

WX → Y : Data Calibrator

Towards Accurate, Efficient and Robust Models

Training Stage



Input Image , ***One Hot Label***

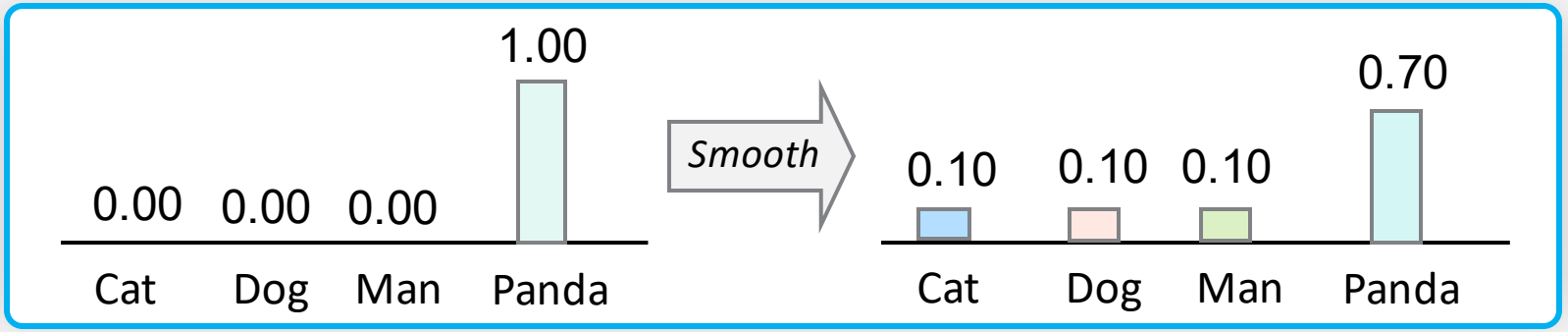
WX → Y

ImageNet X: 14 million images Y: 1000

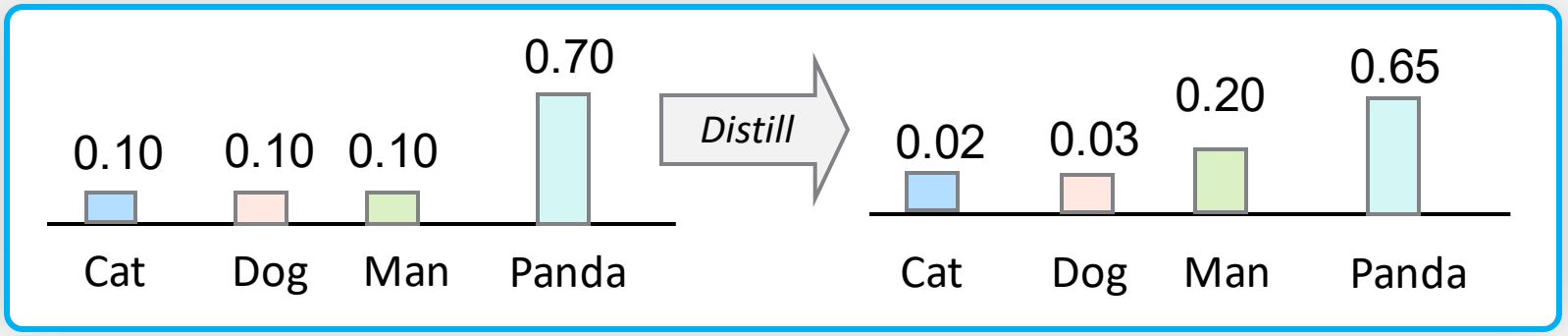
- Forcing every class to one point.
- Problems: NO variance within one class, NO distance between classes.

Refinement on Labels

- From one hot labels to smoothing labels.



- From smoothing labels to relational labels.

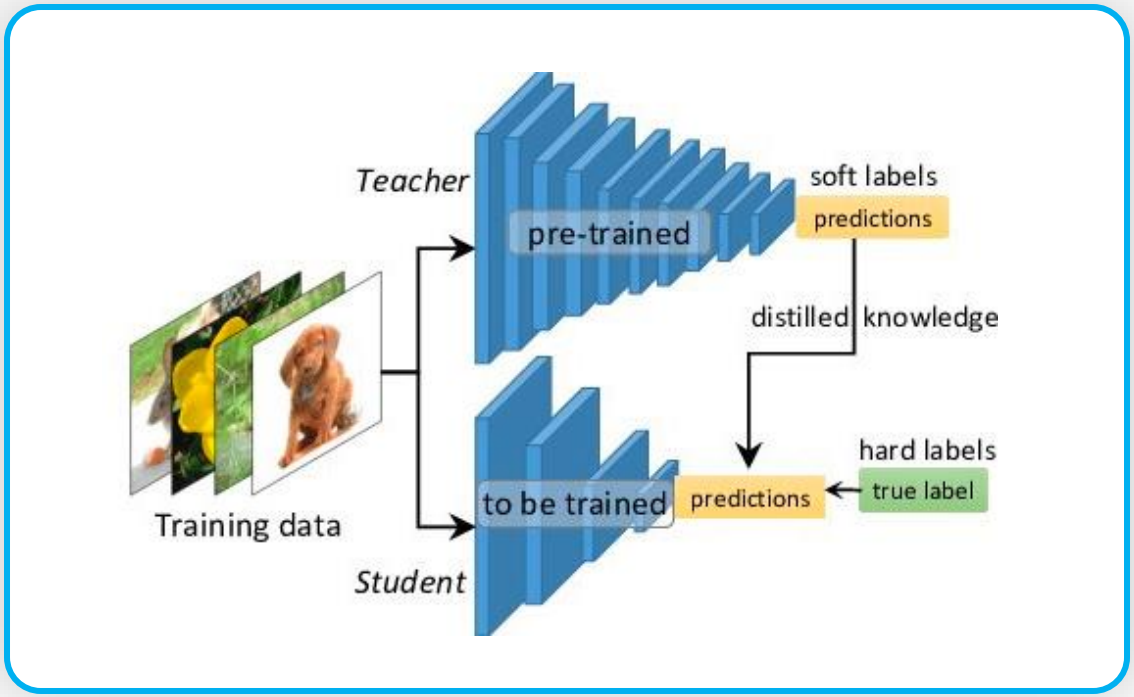


- Design better relations.

❖ Szegedy et al., Rethinking the Inception Architecture for Computer Vision, arxiv.1512.00567

Hinton et al., Distilling the Knowledge in a Neural Network, arxiv.1503.02531

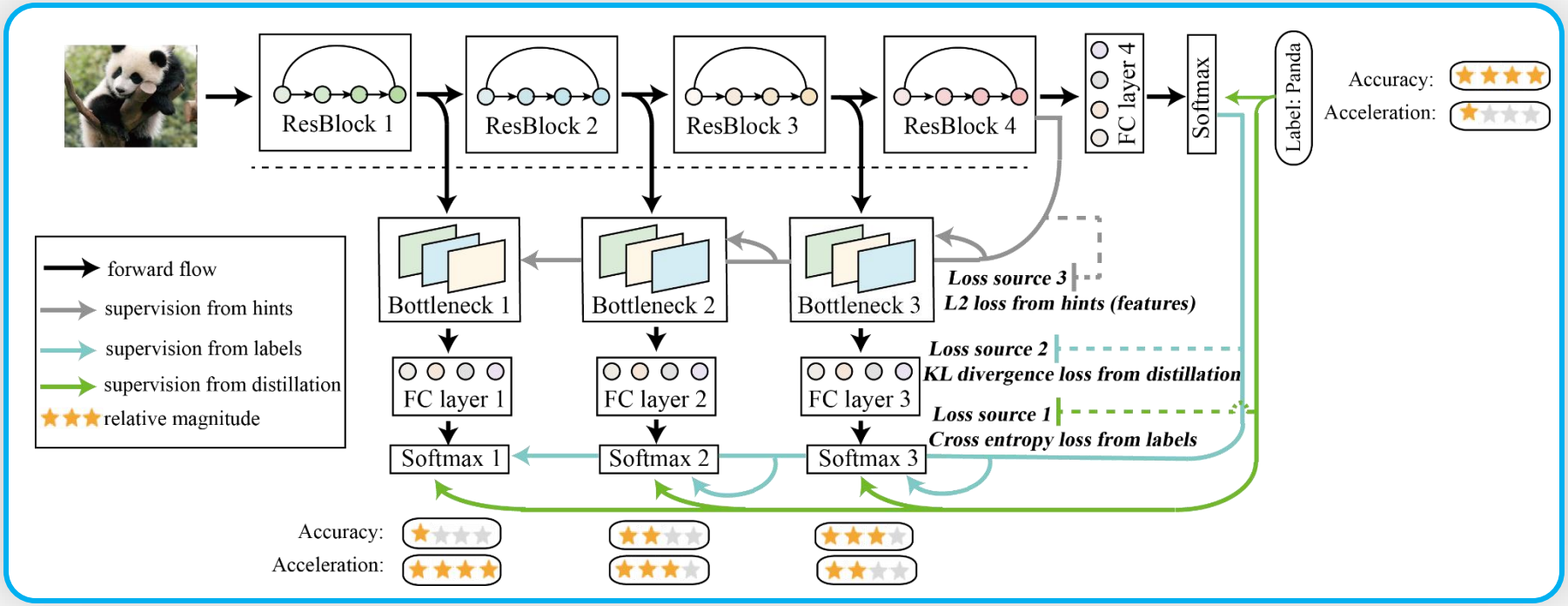
Introduction to Distillation ❖



- But we need a large network as Teacher Model, Can we remove it?

❖Hinton, Geoffrey, Oriol Vinyals, and Jeff Dean. "Distilling the knowledge in a neural network." arXiv preprint arXiv:1503.02531 (2015).

Introducing Self-Distillation – Deeper Ones Teach Shallower Ones



Methods

- **Multi-Exits Neural Network**: Attach additional shallow classifier.
- **Self Distillation:** Regard the deepest classifier as teacher model and distill it to all the shallow classifiers with KL divergence and hint loss.
- **Self Ensemble**: Ensemble both shallow and deep classifiers.

❖ Be Your Own Teacher: Improve the Performance of Convolutional Neural Networks via Self Distillation, ICCV2019, Linfeng Zhang, Jiebo Song, Anni Gao, Jingwei Chen, Chenglong Bao, Kaisheng Ma.

Self Distillation: Loss Function

$$loss = \sum_{i}^{C} loss_i$$

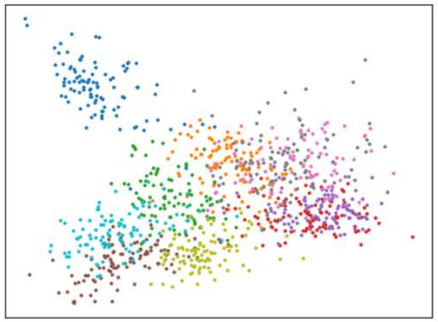
$$= \sum_{i}^{C} (CrossEntropy\,(q_i\,, \mathrm{y}) + \mathrm{KL}(q_i\,, q_C) + \beta \cdot (F_i\,, F_C))$$

$$= \sum_{i}^{C} (\sum_{j} y_i \cdot \log\left(\frac{1}{q_i^j}\right) + \sum_{j} q_i^j \cdot \log\frac{q_i^j}{q_C^j} + \beta \cdot |F_i - F_C|^2)$$

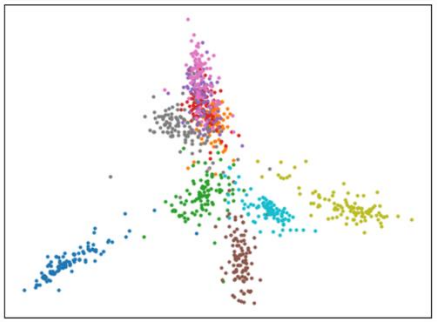
Loss function

- **From labels：** Cross Entropy between labels and outputs of softmax.
- **From teachers：** KL divergence between teachers and students.
- **From hints：** L2 distance of feature maps between teachers and students.
- One-hot Code Problems: NO variance within one class, NO distance between classes.

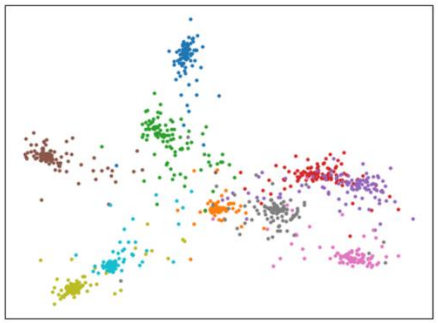
Why Does It Work: More Discriminative Features



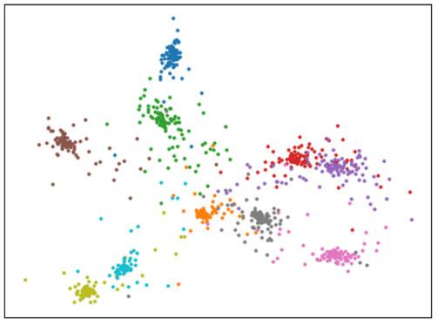
(a) Classifier 1/4



(b) Classifier 2/4



(c) Classifier 3/4



(d) Classifier 4/4

SSE	SSB	**SSE/SSB**	Acc
20.85	1.08	**19.21**	71.21
8.69	1.15	**7.54**	80.86
11.42	1.87	**6.08**	81.58
11.74	2.05	**5.73**	81.59

- SSE: Sum of squares due to error.
- SSB: Sum of squares between groups

Observation

- The Deeper the classifier is, the more discriminative the features are.
- Ratio of SSE/SSB decrease as the classifier goes deeper.

Experiments: Accuracy Compared with No-distillation Baseline.

Table1: Experiments results of accuracy (%) on CIFAR100

Neural Networks	Baseline	Classifier 1/4	Classifier 2/4	Classifier3/4	Classifier 4/4	Ensemble
VGG19(BN)	64.47	63.59	67.04	68.03	67.73	68.54
ResNet18	77.09	67.85	74.57	78.23	78.64	79.67
ResNet50	77.68	68.23	74.21	75.23	80.56	81.04
ResNet101	77.98	69.45	77.29	81.17	81.23	82.03
ResNet152	79.21	68.84	78.72	81.43	81.61	82.29
ResNeXt29-8	81.29	71.15	79.00	81.48	81.51	81.90
WideResNet20-8	79.76	68.85	78.15	80.98	80.92	81.38
WideResNet44-8	79.93	72.54	81.15	81.96	82.09	82.61
WideResNet28-12	80.07	71.21	80.86	81.58	81.59	82.09
PyramidNet101-240	81.12	69.23	78.15	80.98	82.30	83.51

Observation

- As a compression & acceleration method, a ResNet18 equipped with self distillation outperform ResNet152, achieving **5.33** compression and **6.27** acceleration.
- As a method to boost accuracy, self distillation bring **2.65%** increment on average, varying from **4.07%** on VGG and **0.61%** on ResNeXt.
- Deeper neural networks benefit more from self distillation.

Experiments: Compared to Traditional Distillation

Table2: Accuracy (%) comparison with traditional distillation on CIFAR100

Teacher Model	Student Model	Baseline	KD [15]	FitNet [32]	AT [42]	DML [43]	Our approach
ResNet152	ResNet18	77.09	77.79	78.21	78.54	77.54	78.64
ResNet152	ResNet50	77.68	79.33	80.13	79.35	78.31	80.56
WideResNet44-8	WideResNet20-8	79.76	79.80	80.48	80.65	79.91	80.92
WideResNet44-8	WideResNet28-12	80.07	80.95	80.53	81.46	80.43	81.58

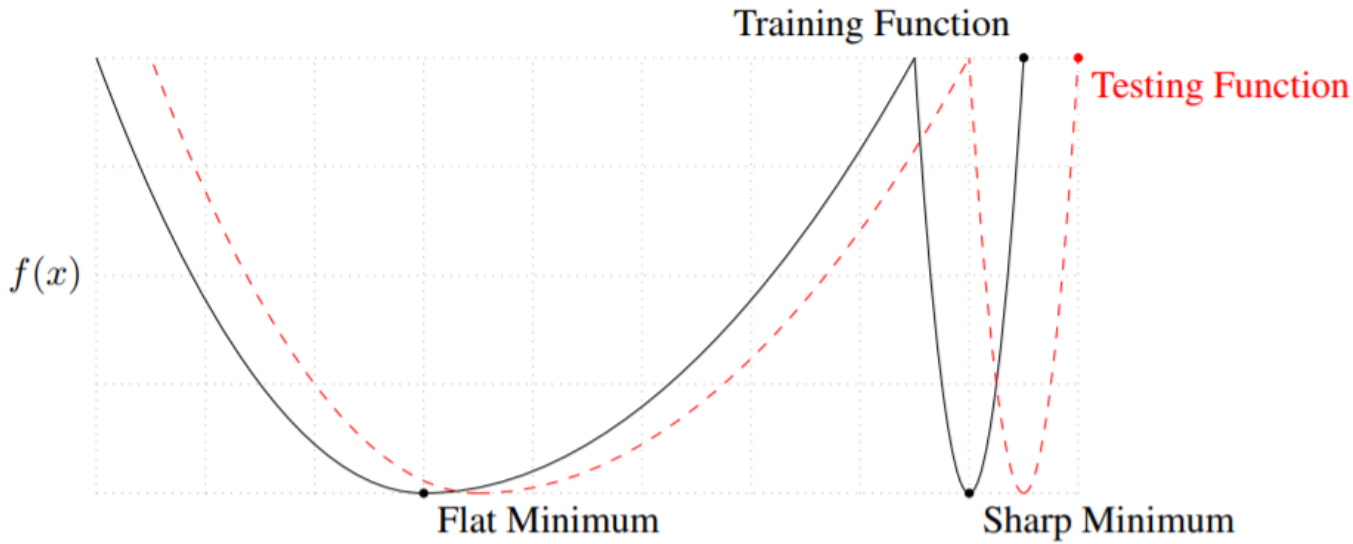
Table3: Accuracy (%) comparison with deeply supervised net on CIFAR100.

Neural Networks	Method	Classifier 1/4	Classifier 2/4	Classifier3/4	Classifier 4/4	Ensemble
ResNet18	DSN	67.23	73.80	77.75	78.38	79.27
	Our approach	67.85	74.57	78.23	78.64	79.67
ResNet50	DSN	67.87	73.80	74.54	80.27	80.67
	Our approach	68.23	74.21	75.23	80.56	81.04
ResNet101	DSN	68.17	75.43	80.98	81.01	81.72
	Our approach	69.45	77.29	81.17	81.23	82.03
ResNet152	DSN	67.60	77.04	81.06	81.35	81.83
	Our approach	68.84	78.72	81.43	81.61	82.29

Observation

- Self distillation outperform prior distillation method with less training time (**4.6X** faster).
- Compared with prior multi-classifiers algorithm, self distillation works better on all the classifiers.

Why Does It Work: Flat Minima and Better Generalization

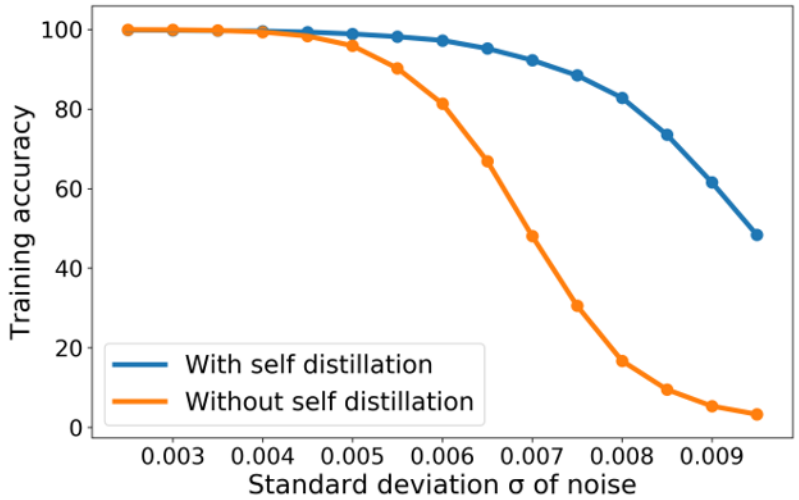


Observation

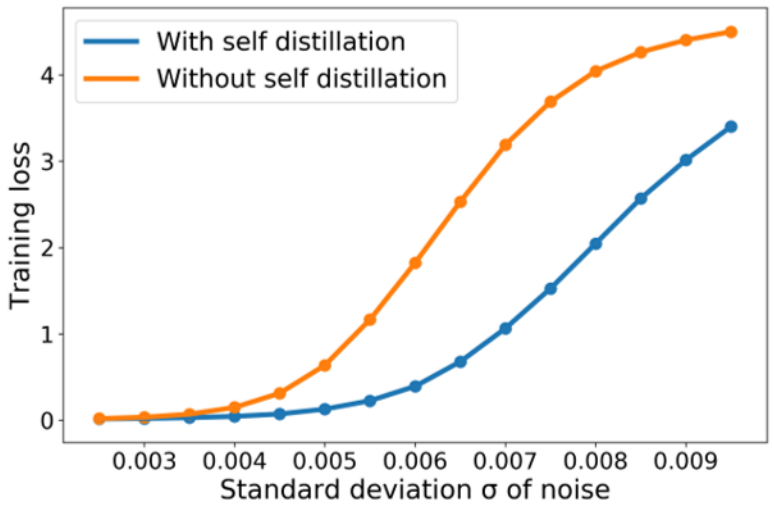
- Flat minima shows more robustness to bias from datasets.
- After adding Gaussian noise to models parameters, models' accuracy and loss changed as depicted in figure a and b.
- Experiments show the model trained with self distillation show more robustness to the noise.

❖ N.S.Keskar *et al.* On large batch training for deep learning: Generalization gap and sharp minima.

Why Does It Work: Flat Minima and Better Generalization



(a) Training accuracy

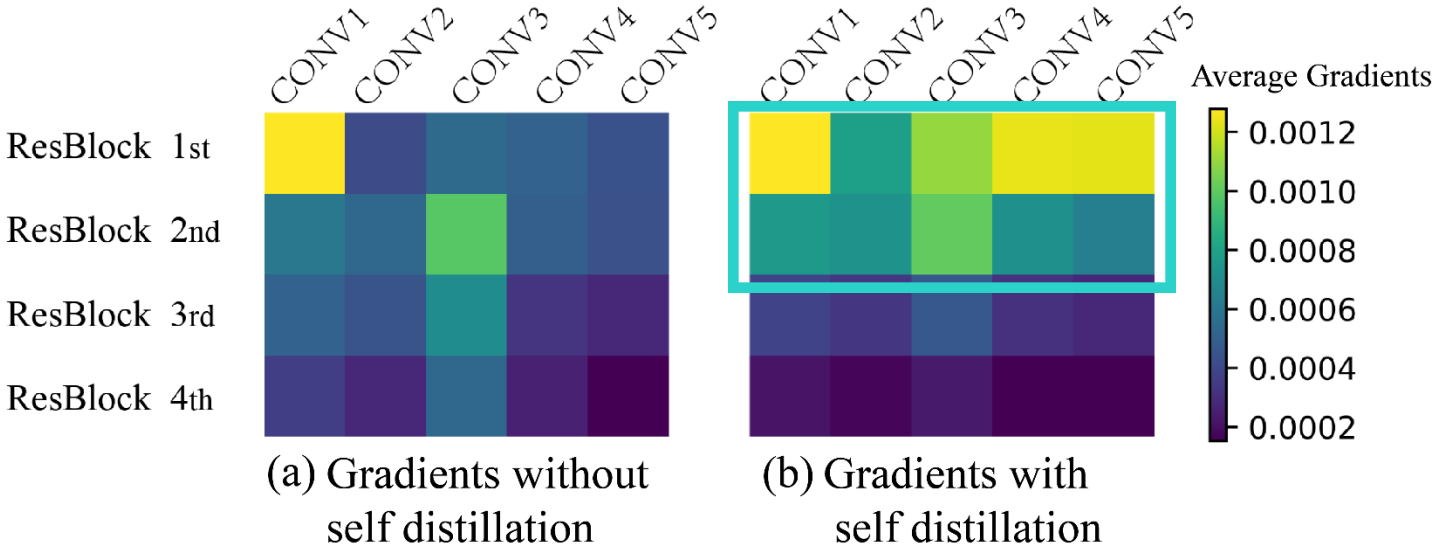


(b) Training loss

Observation

- Flat minima shows more robustness to bias from datasets.
- After adding Gaussian noise to models parameters, models' accuracy and loss changed as depicted in figure a and b.
- Experiments show the model trained with self distillation show more robustness to the noise.

Why Does It Work: Better Gradients



(a) Gradients without self distillation

(b) Gradients with self distillation

Observation

- Deep neural networks always suffer from the gradients vanishing problem.
- Neural Networks trained with self distillation has larger magnitude of gradients.

Towards Accurate, Efficient and Robust Models

WX → Y : Self-Distillation, ICCV2019

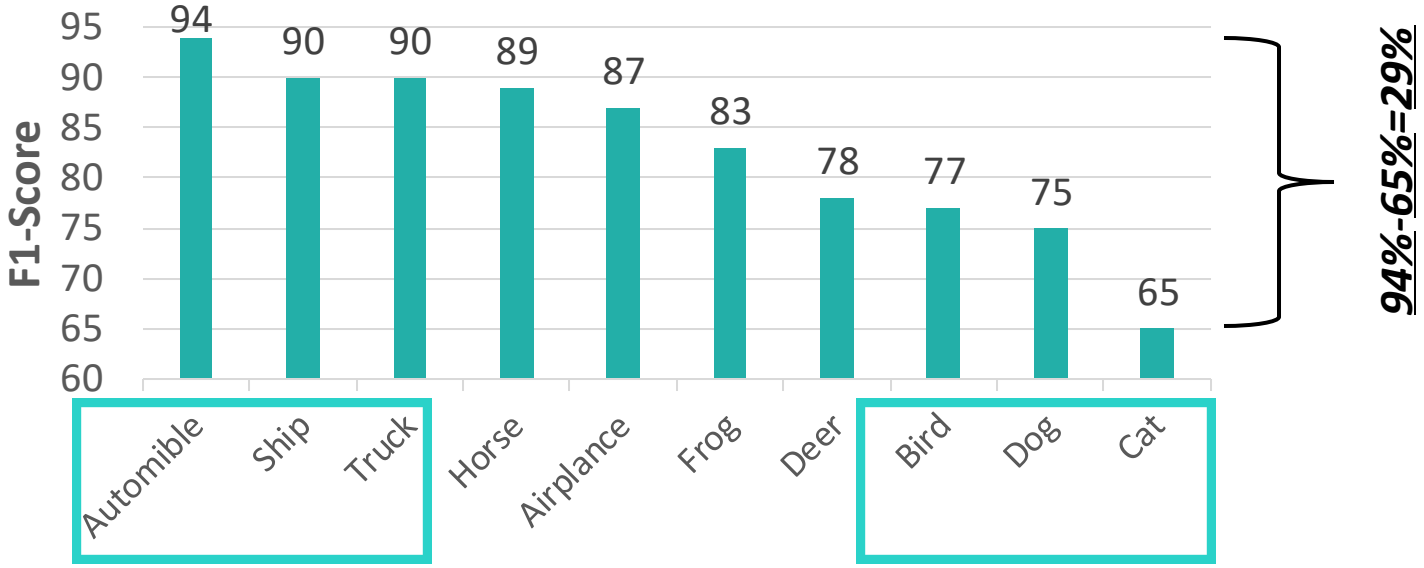
WX → Y : Scalable Neural Networks, NeuralPS 2019

WX → Y : ADA: Augment Data Augmentation

WX ----> Y : Auxiliary Training

WX → Y : Data Calibrator

Towards Accurate, Efficient and Robust Models



✓ Inference Stage Observation

- Difference between the largest and the least F1-score is ***29%.***
- Difference between the top-3 and the min-3 F1-score is ***19%.***
- Conclusion: There is much different among difference classes !

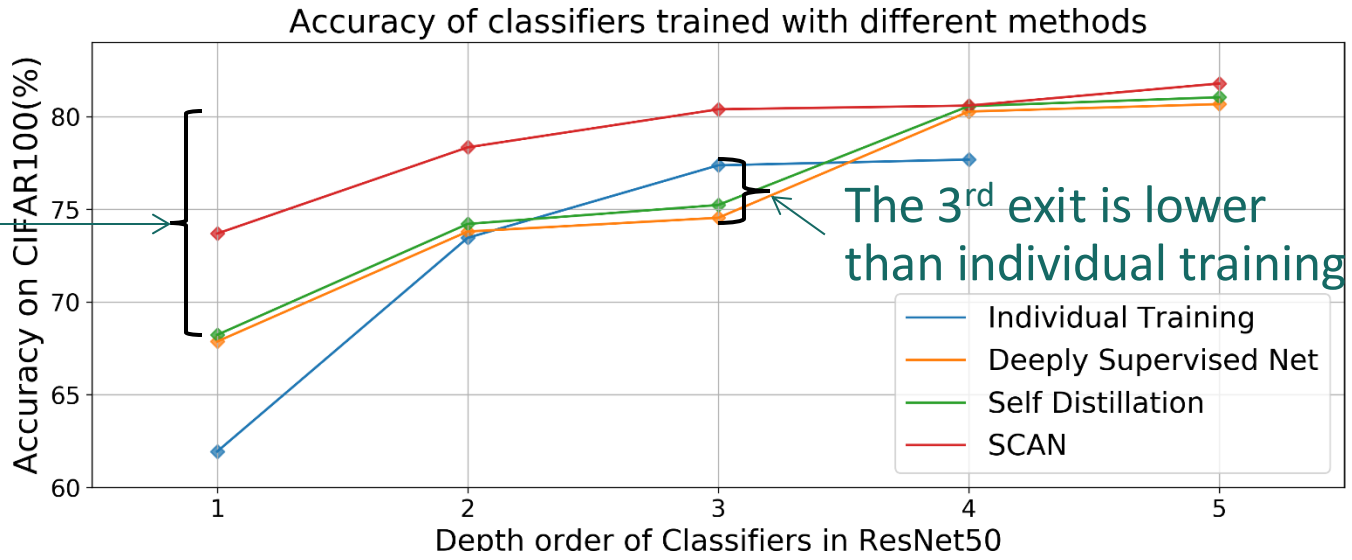
WX → Y

➢ Why must we spend same efforts for those easy classes?

❖AlexNet on CIFAR10.

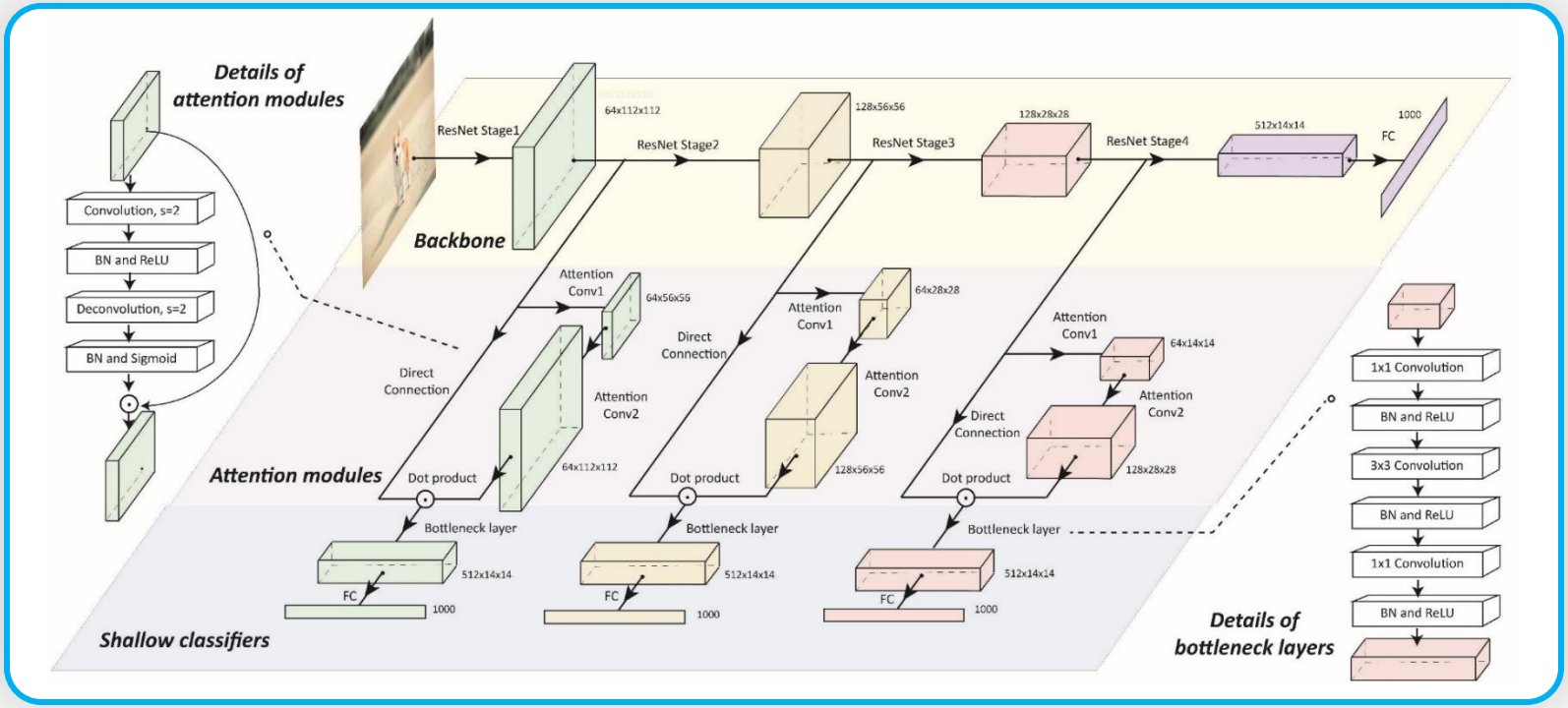
Motivation for Attention Modules in SCAN

Accuracy drops seriously on shallow exits in self-distilation.



- As is shown in the figure, nearly **12% and 7%** accuracy drop can be observed on the first and second classifiers in self distillation.
- The 3rd shallow classifier of self distillation is lower than individually training, indicating that **the training of deep classifiers may influence the training of shallow classifiers**.

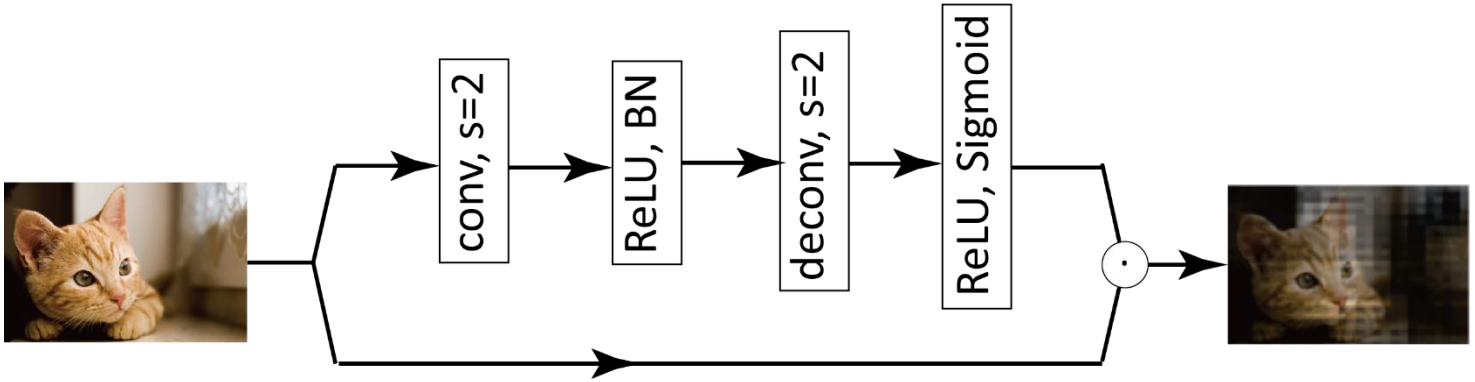
Proposed Method: Scalable Networks (SCAN)



Method

- Adding additional **attention** modules in shallow classifiers to enable they to obtain specific features from backbone.

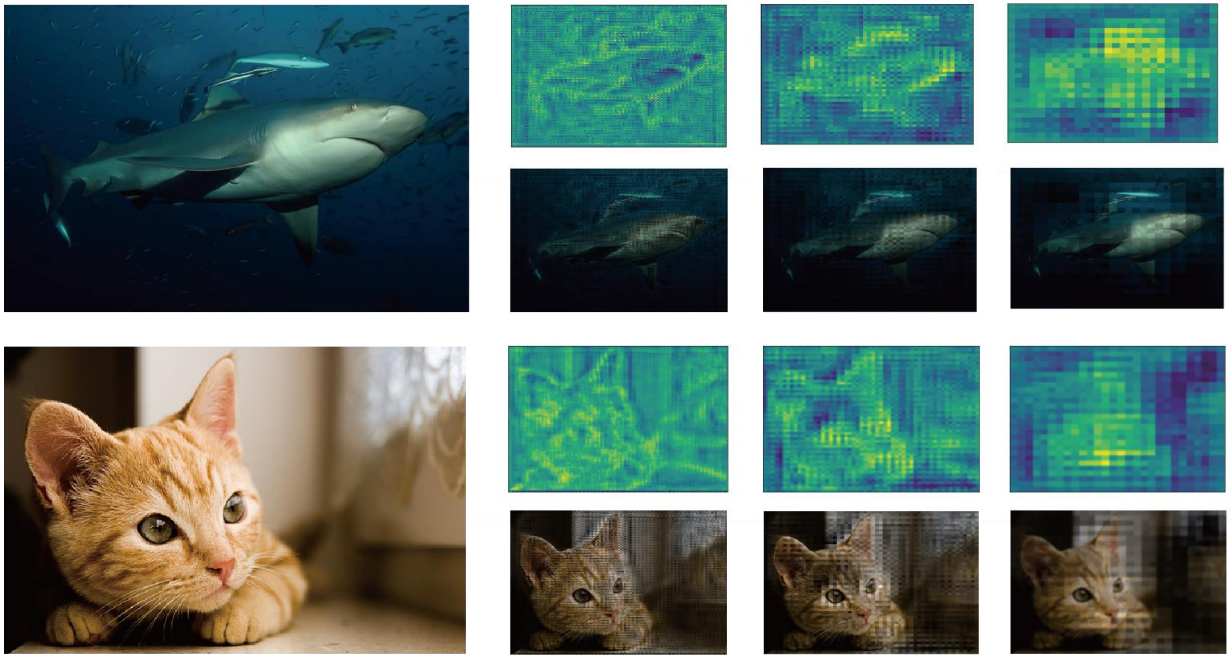
Attention Modules in SCAN



- Propose attention modules in SCAN is composed of one convolution layer for downsampling and one deconvolution for upsampling.
- A sigmoid layer is attached after them to obtain attention factor from 0 to 1. Then, a dot production is brought about to get the attention enhanced features.
- The whole process can be written as

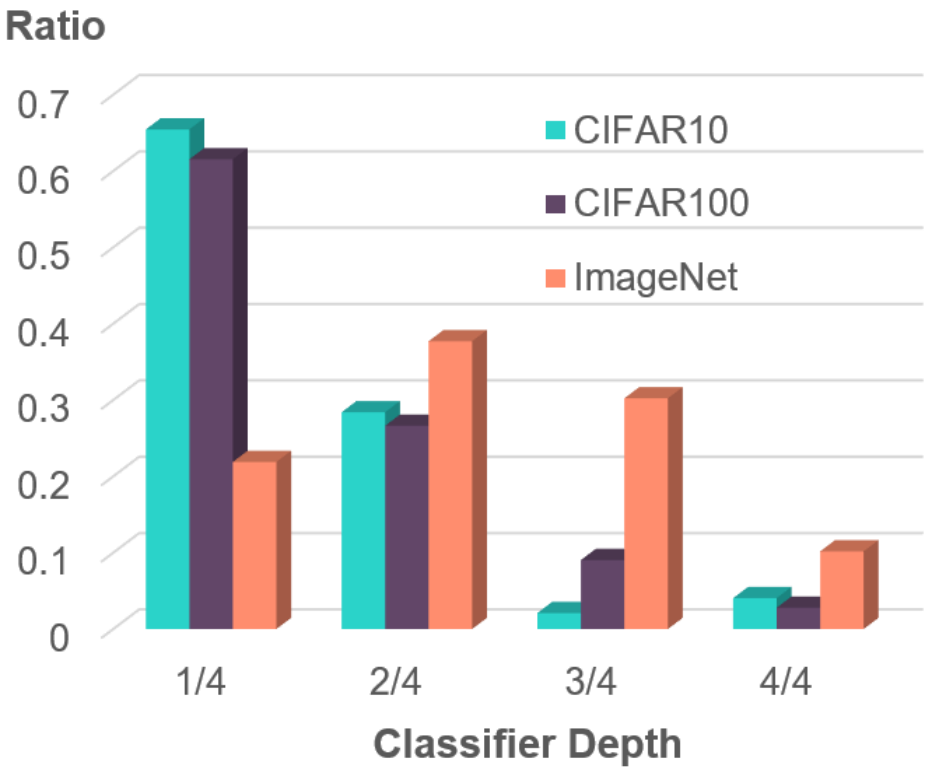
$$Attention\ Maps(W_{conv}, W_{deconv}, F) = \sigma(\phi(\psi(F, W_{conv})), W_{deconv})$$

Discussion: What have attention modules learned



- All the classifiers pay their attention on the same spatial position - the bodies of a shark and a cat, while ignoring the backgrounds.
- Shallow classifiers pay more attention to the local and high frequency features, while deep classifiers pay more attention to global and low frequency features.

Discussion: Ratio of Samples Predicted by Each Classifier



- This figure shows the statistics of samples predicted by each classifier of three datasets with the same thresholds.
- More than half samples in CIFAR can be classified in the shallowest classifier.
- In ImageNet, more samples have to be predicted in the last two classifiers

Experiments Results on CIFAR100 and ImageNet

Table 1: Experiments results of accuracy (%) on CIFAR100.

Models	Baseline	Classifier1/4	Classifier2/4	Classifier3/4	Classifier4/4	Ensemble
VGG16(BN)	72.46	71.29	74.92	75.18	75.29	76.80
VGG19(BN)	72.25	71.52	74.02	74.15	74.43	75.43
ResNet18	77.09	71.84	77.74	78.62	79.13	80.46
ResNet50	77.68	73.69	78.34	80.39	80.45	81.78
ResNet101	77.98	72.26	79.26	80.95	81.12	82.06
ResNet152	79.21	73.14	80.40	81.73	81.62	82.94
WRN20-8	74.61	74.52	78.17	79.25	/	80.04
WRN44-8	76.22	76.02	78.74	79.67	/	80.35

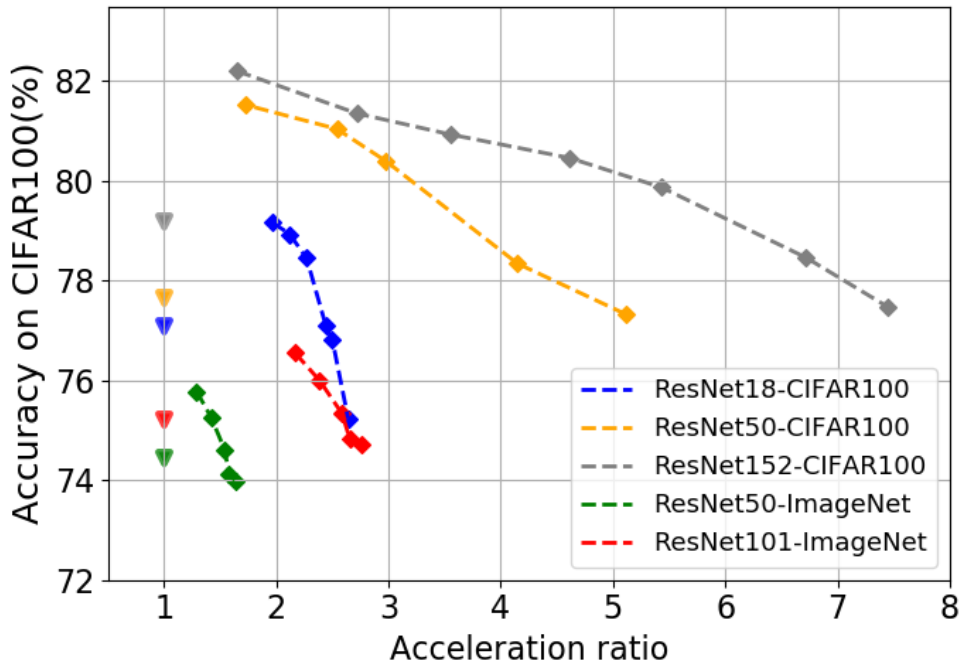
Table 2: Experiments results of accuracy (%) on ImageNet.

Models	Baseline	Classifier1/4	Classifier2/4	Classifier3/4	Classifier4/4
ResNet18	68.02	48.25	58.00	65.32	69.32
ResNet50	74.47	53.86	66.54	73.57	75.88
ResNet101	75.24	52.32	65.33	74.51	76.32

Experiments

- In all the situations, the classifier2/4 equipped with SCAN outperforms its baseline. **2.17X** acceleration and **3.20X** compression have been achieved on average with no accuracy drop.

Experiments Results of Scalable Inference



X axis: acceleration ratio compared to its baseline.

Y axis: top 1 accuracy evaluated on CIFAR100 and ImageNet.

Squares connected in the same line denotes the results of different

Triangles on x=1 denote baselines of different neural networks.

Methods & Results

- Thresholds based dynamitic inference with shallow classifiers.
- Apply gene algorithm to search the proper thresholds.
- On average, **4.41X** and **1.99X** acceleration can be achieved with no accuracy drop on CIFAR100 and ImageNet, respectively.

Towards Accurate, Efficient and Robust Models

WX → Y : Self-Distillation, ICCV2019

WX → Y : Scalable Neural Networks, NeuralPS 2019

WX → Y : ADA: Augment Data Augmentation

WX ----> Y : Auxiliary Training

WX → Y : Data Calibrator

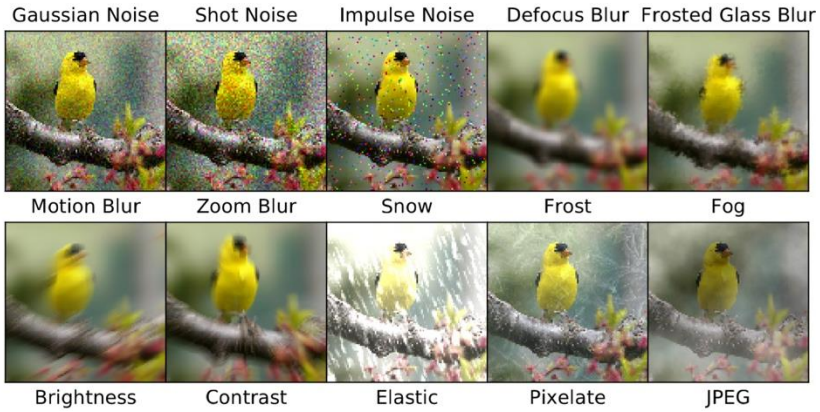
Towards Accurate, Efficient and Robust Models

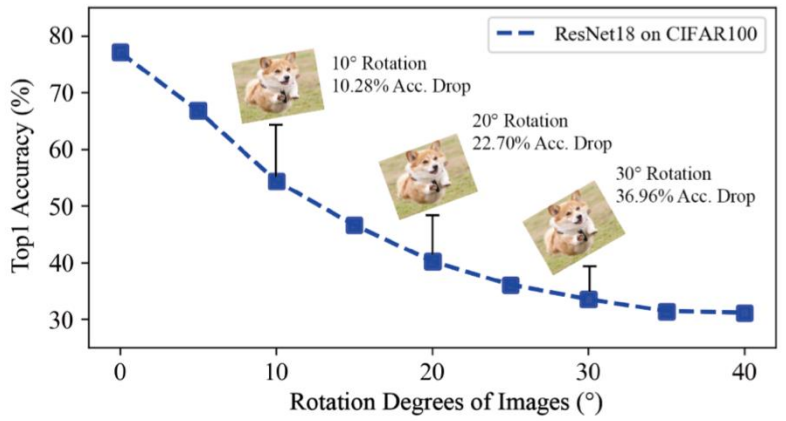
- During the stage of closing the gap, can we close them in the feature space, rather than label space?

$$WX \rightarrow Y$$

Towards Accurate, Efficient and Robust Models

- Nature corruption problems: Noise, Blurring, Rotation and so on.





- Domain shift problems: from GTA5 to real world.

Udacity

GTA5

❖Hendrycks, Dan, and Thomas Dietterich. "Benchmarking neural network robustness to common corruptions and perturbations." arXiv preprint arXiv:1903.12261 (2019).

Towards the Relation Between Clean and Corrupted Samples

✓ Motivation

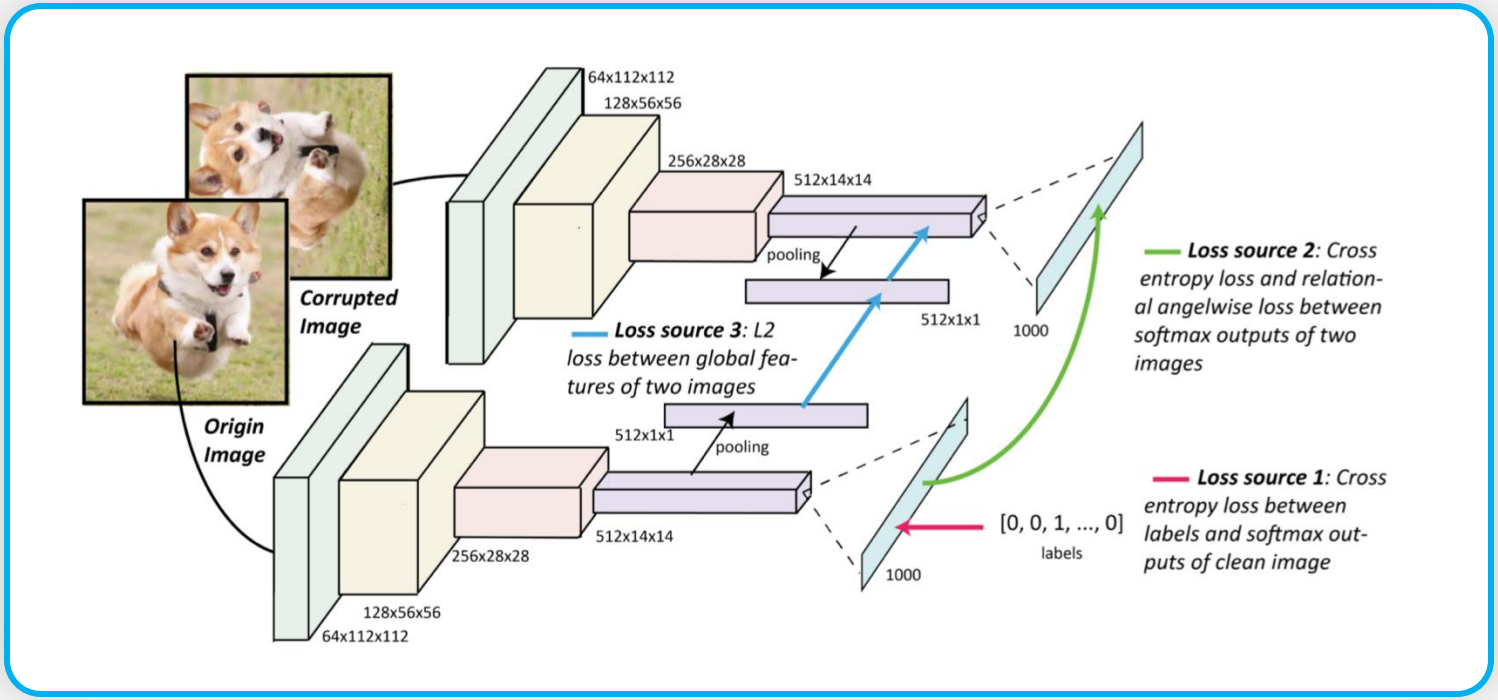
- In data augmentation training, the origin images and corrupted images have 1:N relation.

Rotation 90° **Origin Image** **Rotation 180°**

- However, this relation is not exposed to neural networks in the conventional data augmentation.
- **Instead of learning the corrupted images, learn their relation to clean images**.

❖Figures from Web.

ADA: Augment Data Augmentation by Relationship

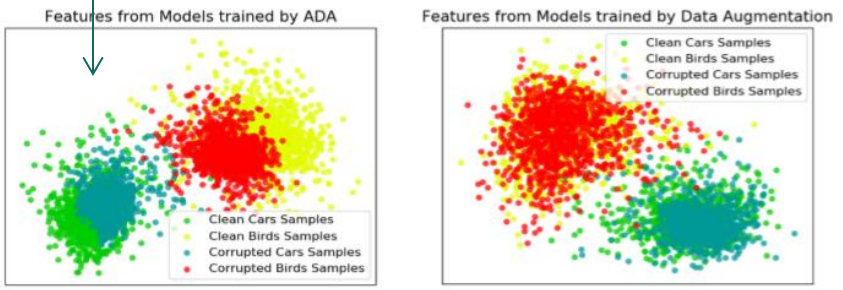


Methods

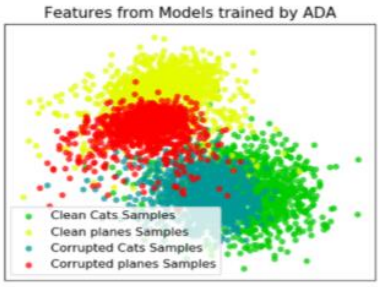
- Instead of training models on corruption images, force neural networks to have the same features from corruption images and clean images.
- Design relation based loss function to facilitate the training.

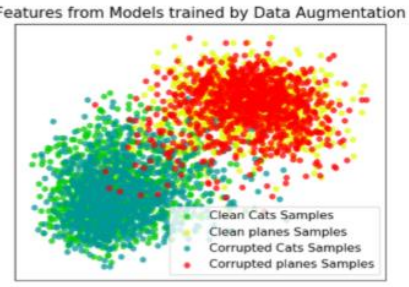
PCA Visualization of ADA and Data Augmentation

It seems like the corrupted features are a different distribution.

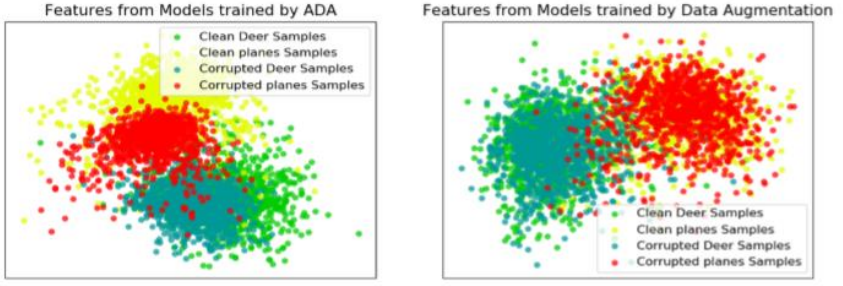


(a) Features visualization of **Cars** and **Birds** from ResNet18 trained by the proposed ADA and common data augmentation.

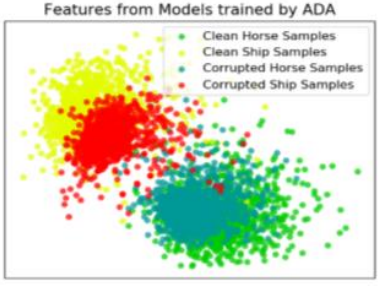


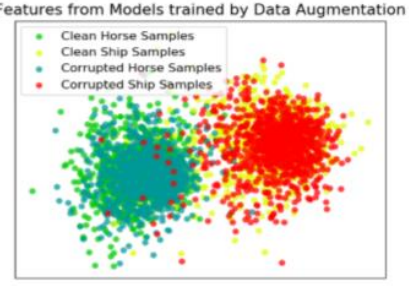


(b) Features visualization of **Cats** and **Planes** from ResNet18 trained by the proposed ADA and common data augmentation.



(c) Features visualization of **Deer** and **Planes** from ResNet18 trained by the proposed ADA and common data augmentation.





(d) Features visualization of **Horses** and **Ships** from ResNet18 trained by the proposed ADA and common data augmentation.

✓ PCA visualization of ADA and data augmentation

- The clean and corrupted images in the same classes are distributed more densely in ADA, indicating ADA enables neural networks to extract robust features.

Experiment Results on Robustness

Table 1. Experiments results on CIFAR100-C and ResNet18 for nine kinds of image corruption. Only Gaussian noise and Gaussian blur are involved in models training by data augmentation and the proposed ADA.

Training Method	Glass Blur	Shot Noise	Speckle Noise	Gaussian Blur	Motion Blur	Defocus Blur	Impluse Noise	Zoom Blur	Gaussian Noise
Baseline	24.09	46.73	47.14	51.09	54.90	59.74	37.98	53.98	40.48
Data Augmentation	58.55	57.70	58.14	60.31	59.05	62.40	62.64	61.71	55.02
Our Approach	59.12	58.40	58.71	60.62	59.71	63.44	63.41	62.06	56.14

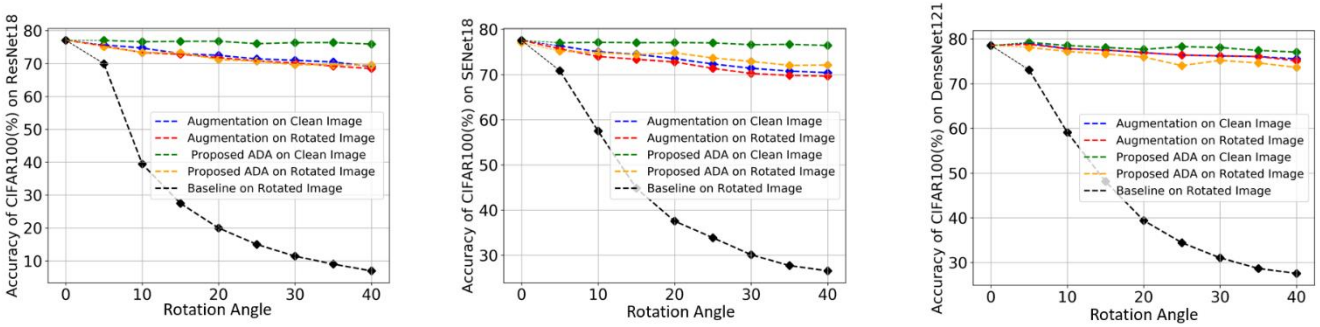
Table 2. Experiments results on CIFAR10-C and ResNet18 for nine kinds of image corruption. Only Gaussian noise and Gaussian blur are involved in models training of data augmentation and the proposed ADA.

Training Method	Glass Blur	Shot Noise	Speckle Noise	Gaussian Blur	Motion Blur	Defocus Blur	Impluse Noise	Zoom Blur	Gaussian Noise
Baseline	59.02	73.44	74.92	73.55	78.51	82.85	69.80	79.04	66.06
Data Augmentation	85.53	87.35	87.39	86.45	84.85	88.15	88.40	87.38	86.13
Our Approach	85.84	87.87	87.75	86.96	85.14	88.34	88.97	87.85	86.83

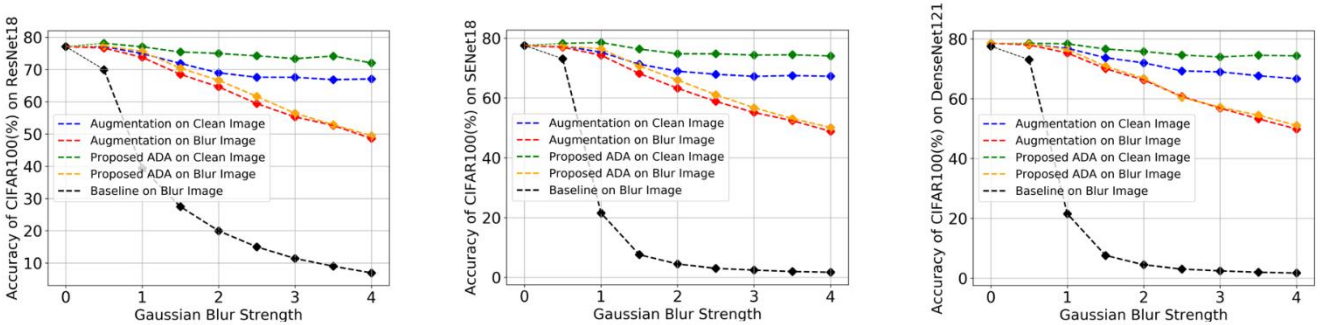
✓ Results on CIFAR-C

- CIFAR10-C and CIFAR100-C are two benchmarking datasets designed for measuring models robustness.
- The proposed ADA can improve more robustness on both two datasets than conventional data augmentation.

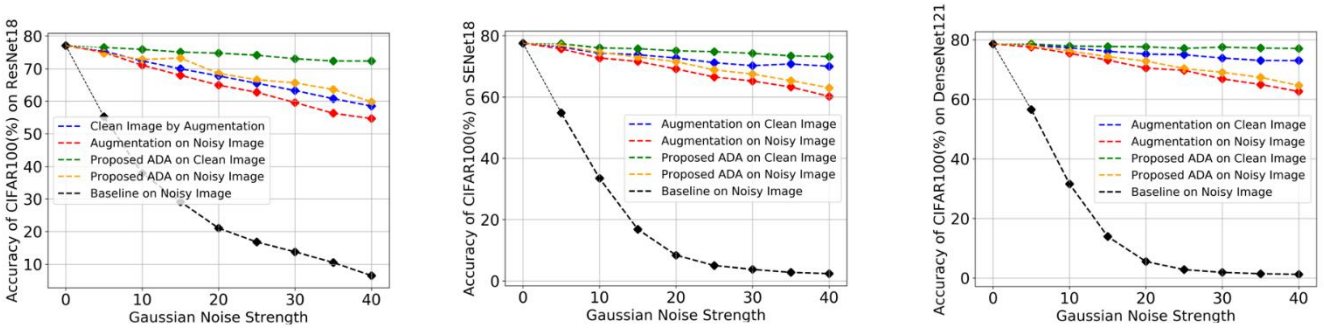
Still Better When the Noise Increases



(a) **Rotation Corruption:** Accuracy of models trained and evaluated on both clean and rotated Images.



(b) **Blur Corruption:** Accuracy of models trained and evaluated on both clean and images with Gaussian blur.



(c) **Noise Corruption:** Accuracy of models trained and evaluated on both clean and images with Gaussian noise.

Comparison with Deeper and Wider NNs

- It has nothing to do with **network volume**.

Table 3. Accuracy on two forms of images (%) with ResNet with different depth in CIFAR100.

Models Depth	Data Augmentation		Proposed ADA		Models Parameters(M)	Models FLOPs(M)
	Clean Data	Corrupted Data	Clean Data	Corrupted Data		
18	66.34	61.36	73.53	67.07	2.82	140.45
50	74.34	67.96	76.11	69.19	5.99	331.83
101	74.91	68.79	76.24	69.64	10.75	638.21
152	74.60	68.88	76.34	69.59	14.68	945.38

Table 4. Accuracy on two forms of images (%) with ResNet with different width (channels number) in CIFAR100.

Models Width	Data Augmentation		Proposed ADA		Models Parameters(M)	Models FLOPs(M)
	Clean Data	Corrupted Data	Clean Data	Corrupted Data		
0.50X	62.73	58.17	73.24	66.86	0.71	35.62
1.00X	66.34	61.36	73.53	67.07	2.82	140.45
2.00X	69.44	63.00	73.44	67.01	11.22	557.71
4.00X	71.73	64.80	74.60	66.69	44.76	2223.34

✓ Comparison with Deeper and Wider NNs

- Neural networks with more layers and channels have better nature corruption robustness.
- The proposed ADA can improve models robustness at no expense of parameters and computation in inference.

Towards Accurate, Efficient and Robust Models

WX → Y : Self-Distillation, ICCV2019

WX → Y : Scalable Neural Networks, NeuralPS 2019

WX → Y : ADA: Augment Data Augmentation

WX ----> Y : Auxiliary Training

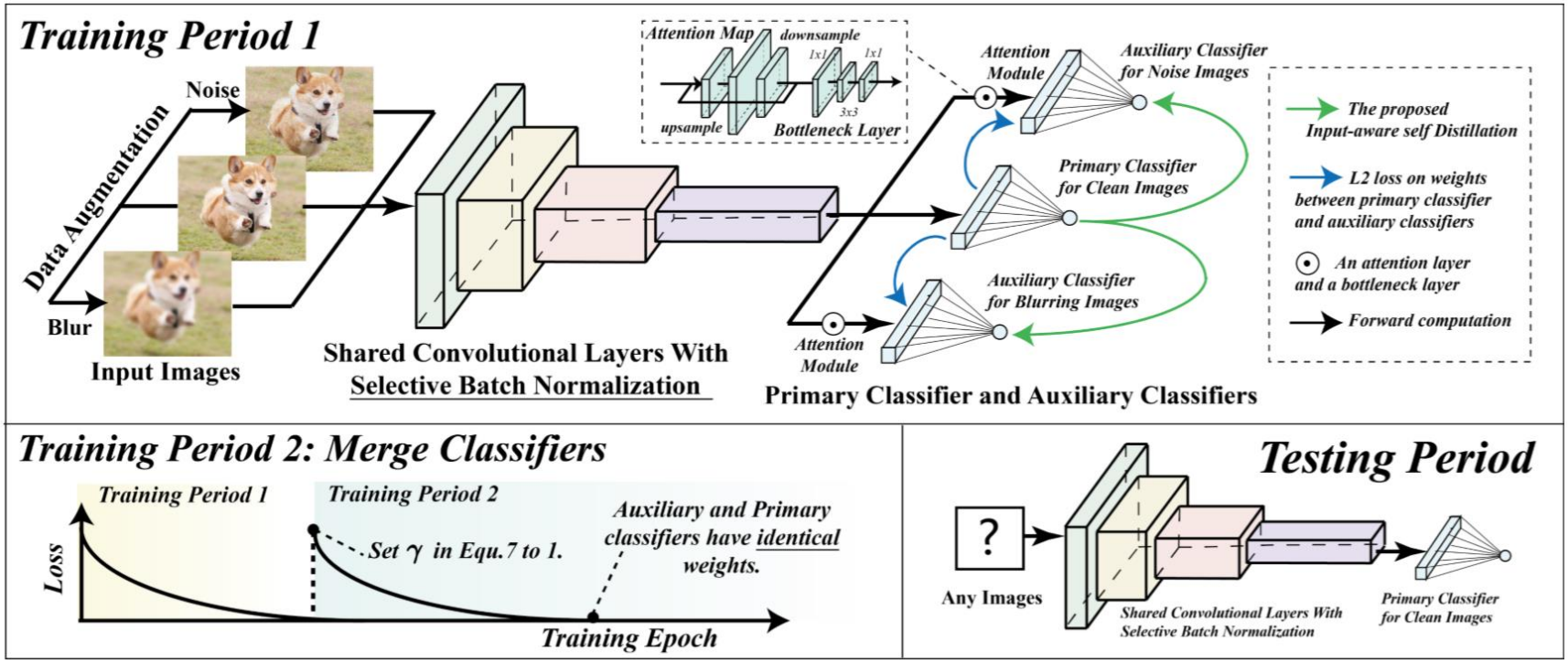
WX → Y : Data Calibrator

Auxiliary Training: Towards Accurate and Robust Models

- The bottleneck is the feature extraction space (CONVs), but rather in classifier space (FCs).

WX ----> Y

Auxiliary Training



Methods

- Neural networks learn image with different corruption with different auxiliary classifiers. All the classifiers ***converge to the same point finally.***
- Apply selective ***batch normalization*** and ***input-aware self distillation*** to enable primary classifiers lean robust information from auxiliary classifiers.

Improve Natural, Robustness and Adversarial Accuracy

Model	Our approach	Baseline	Increment
AlexNet	80.03	100.00	+19.97
ResNet18	69.34	92.21	+22.87
ResNet50	69.13	92.28	+23.15
ResNet101	66.10	88.35	+22.25
WRNet50	68.89	87.33	+18.44
ResNeXt50	69.13	92.29	+23.16

Table 4. Comparison of robustness between models trained by auxiliary training and normal training on CIFAR100-C dataset. WRN indicates Wide ResNet. Model robustness is measured by corruption error (CE) in Equation (8). **Less is better.**

Model	Our approach	Baseline	Increment
AlexNet	70.09	68.44	+1.65
ResNet18	79.47	77.09	+2.38
ResNet50	80.16	77.42	+2.74
ResNet101	80.51	77.81	+2.70
WRN50	80.84	79.08	+1.76
ResNeXt50	81.51	79.49	+2.02

Table 2. Comparison of accuracy (%) between models trained by auxiliary training and standard training on CIFAR100 dataset. WRN indicates wide ResNet.

Training Method	Clean	PGD-L_2	PGD-L_∞	BIA-L_2	BIA-L_∞	FGSM	MIA-L_2	DDN-L_2
Normal Training	94.75	23.37	4.88	24.62	6.49	18.34	24.62	1.42
Adversarial Training [31]	83.90	45.54	43.52	79.94	44.88	51.99	74.04	24.36
Auxiliary Training	85.76	49.35	46.45	82.56	47.07	54.38	76.97	26.53

Experiments

- The proposed auxiliary training achieve **2.21% accuracy improvements**, **21.46% nature robustness improvements**, and **3.17% adversarial accuracy** improvements (compared with adversarial training).

Towards Accurate, Efficient and Robust Models

WX → Y : Self-Distillation, ICCV2019

WX → Y : Scalable Neural Networks, NeuralPS 2019

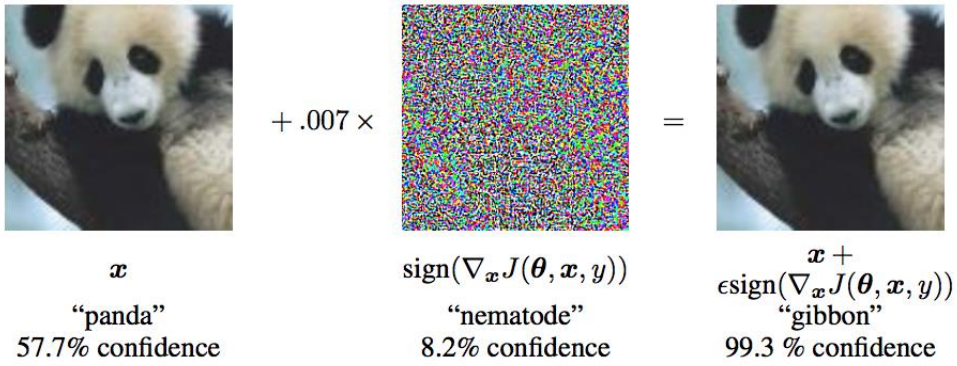
WX → Y : ADA: Augment Data Augmentation

WX ----> Y : Auxiliary Training

WX → Y : Data Calibrator

Light-weight Calibrator: A Separable Component for Unsupervised Domain Adaptation

$$WX \rightarrow Y$$

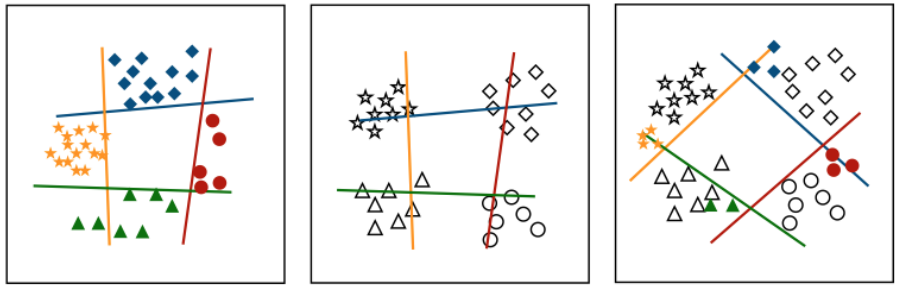


- Can we add good noises?

Light-weight Calibrator: A Separable Component for Unsupervised Domain Adaptation

- Instead of finetuning the network, we modify inputs to fit the source classifier better.

(a) GTA5 (b) GTA5 → Cityscapes (c) CityScapes



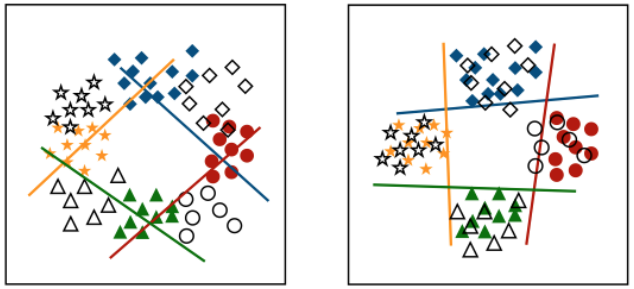
(a) (b) (c)

Transfer Learning.

Domain Adaptation.

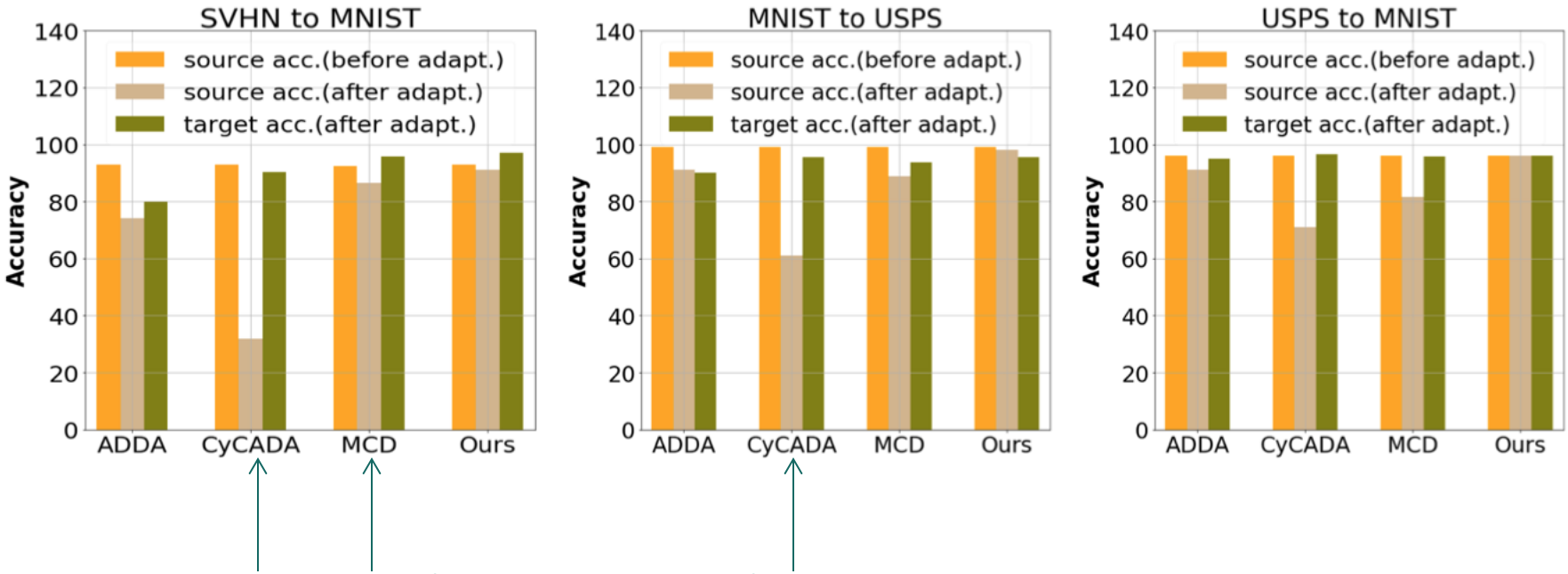
Cares only target domain.

Accuracy drops on source domain. →



(d) (e)

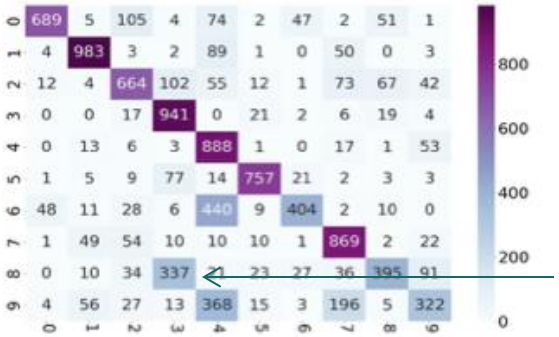
Generalization of the Calibrator Method – A Preview



- Compared to previous methods, our method achieves better target domain performance while having the best source domain performance after the adaptation.

Motivation: Mis-classify Hard Examples

(a) Source prediction at SVHN



(b) Target prediction at MNIST

3,8

4,9

7,9

(c) Our prediction at SVHN

(d) Our prediction at MNIST

- We observe that the source classifier does not lose all its classification power. From SVHN to MNIST, the domain shift causes networks mis-classify hard examples. We show in (d) that our method can correct this problem by modifying the inputs.

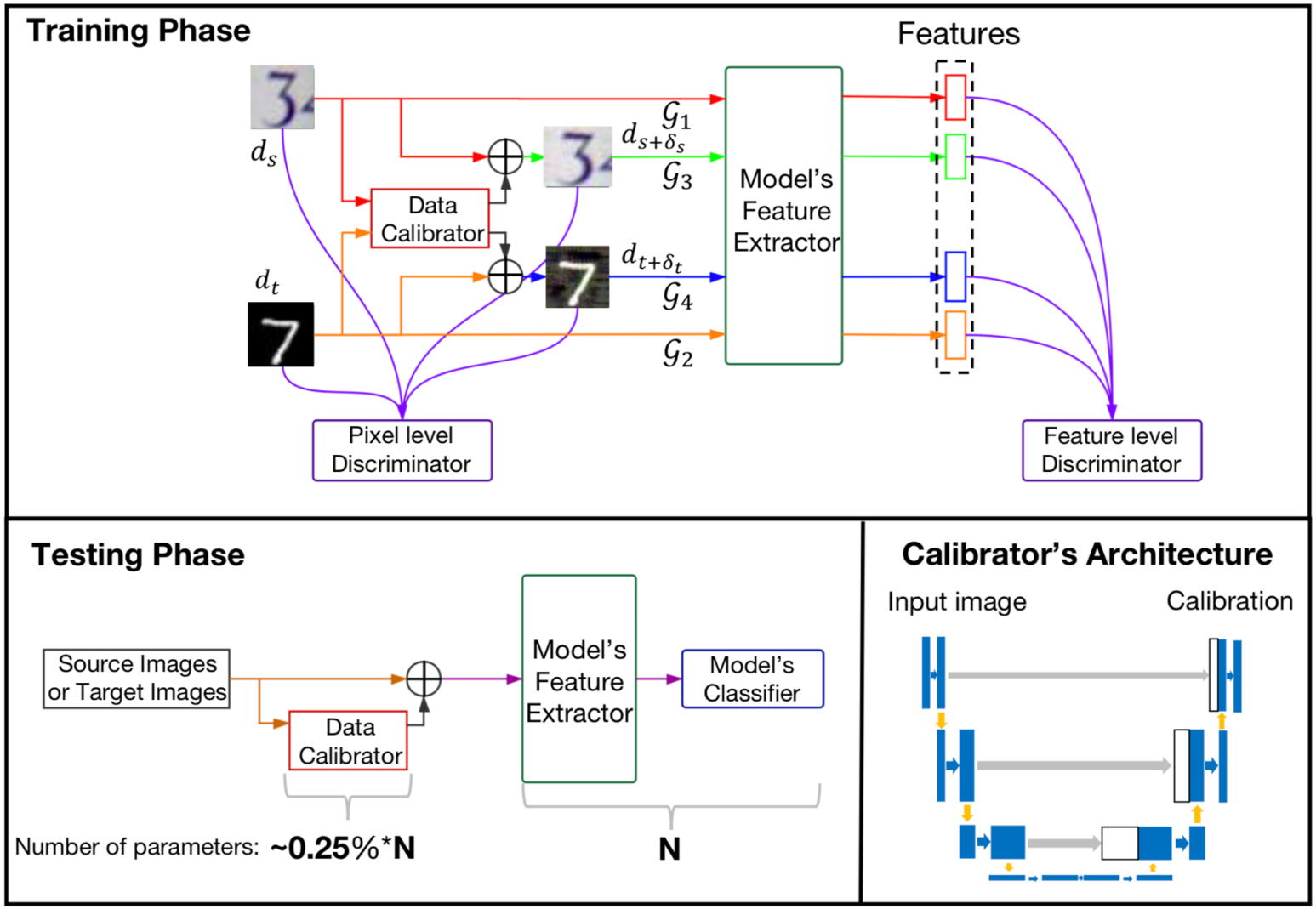
Origin of Domain/Distribution Shift: Biased Datasets

Figure 1. Name That Dataset: Given three images from twelve popular object recognition datasets, can you match the images with the dataset? (answer key below)

- The necessarily of domain adaptation is confirmed by a experiment called ”name the database”. In this game, the authors show that databases are biased can and bias can even be learnt by a neural network. In this game, human can achieve 75% accuracy while classifier can achieve 39%.

❖ *Torralba et al. “Unbiased Look at Dataset Bias.”*

Light-weight Calibrator: A Separable Component for Unsupervised Domain Adaptation



Our results

- In digits benchmark, we outperform state-of-art method by 2.5 points. This shows the effectiveness of our method.

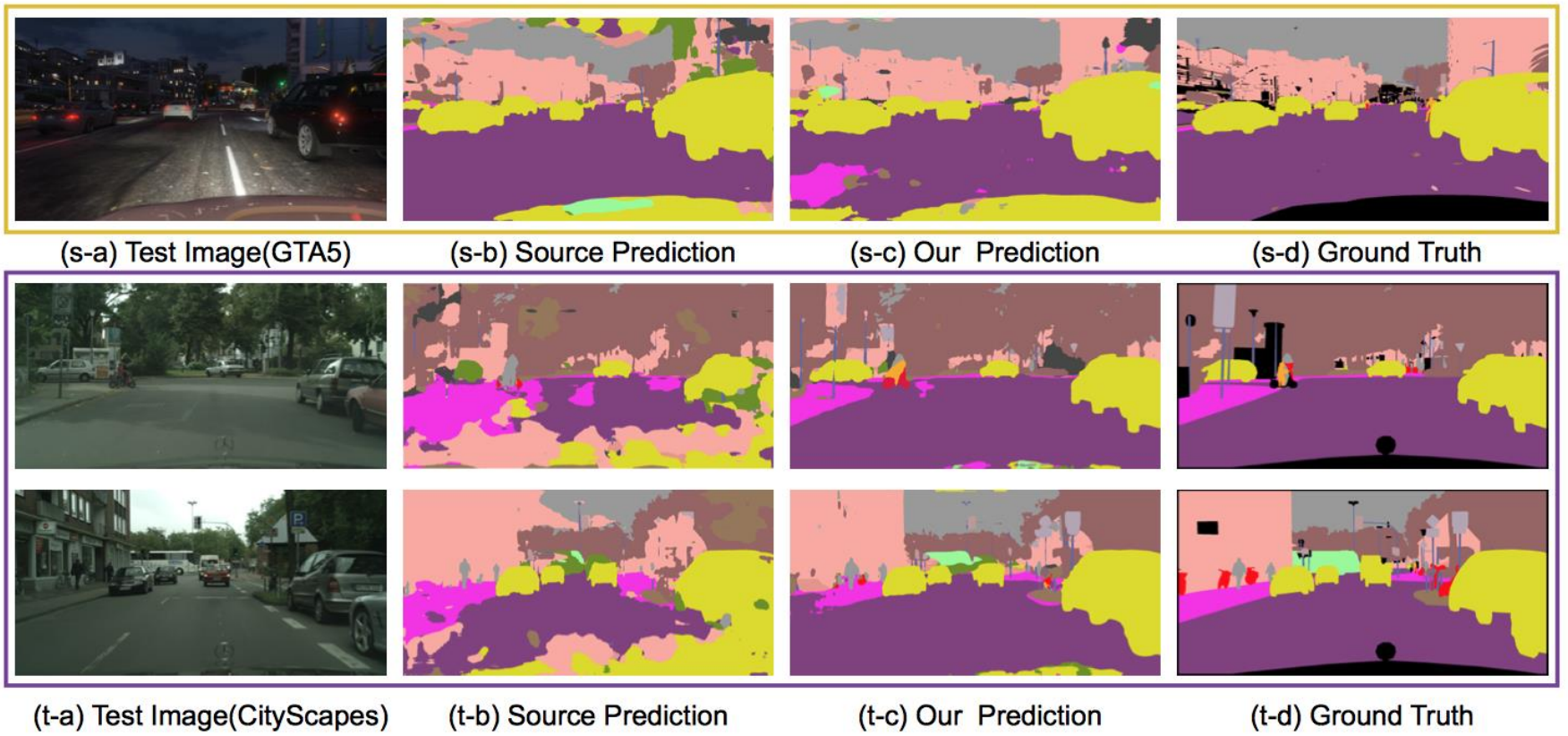
Method	MNIST to USPS	USPS to MNIST	SVHN to MNIST	Average Acc.
ADDN [34]	90.1	95.2	80.1	88.5
CoGAN [18]	91.2	89.1	-	-
SBADA [26]	**97.6**	95.0	76.1	89.6
CYCADA [12]	95.6	96.5	90.4	94.2
CDAN [20]	95.6	98.0	89.2	94.3
PFA [3]	95.0	-	93.9	-
MSTN [37]	92.9	97.6	93.3	94.6
MCD [28]	93.8	95.7	95.8	95.1
Ours	95.6	97.1	97.1	96.6
CyCleGAN+Ours	97.1	**98.3**	**97.5**	**97.6**

- GTA5 to Cityscapes, a show case is followed.

	road	sidewalk	building	wall	fence	pole	traffic light	traffic sign	vegetation	terrain	sky	person	rider	car	truck	bus	train	motorbike	bicycle	**mIoU**	**fwIoU**	**Pixel acc.**
Source only	42.7	26.3	51.7	5.5	6.8	13.8	23.6	6.9	75.5	11.5	36.8	49.3	0.9	46.7	3.4	5.0	0.0	5.0	1.4	21.7	47.4	62.5
CyCADA	79.1	33.1	77.9	23.4	17.3	32.1	33.3	31.8	81.5	26.7	69.0	62.8	14.7	74.5	20.9	25.6	6.9	18.8	20.4	39.5	72.4	82.3
Ours	**83.5**	**35.2**	**79.9**	**24.6**	16.2	**32.8**	33.1	31.8	**81.7**	**29.2**	66.3	**63.0**	14.3	**81.8**	**21.0**	**26.5**	**8.5**	16.7	**24.0**	**40.5**	**75.1**	**84.0**
Target	97.3	79.8	88.6	32.5	48.2	56.3	63.6	73.3	89.0	58.9	93.0	78.2	55.2	92.2	45.0	67.3	39.6	49.9	73.6	67.4	89.6	94.3

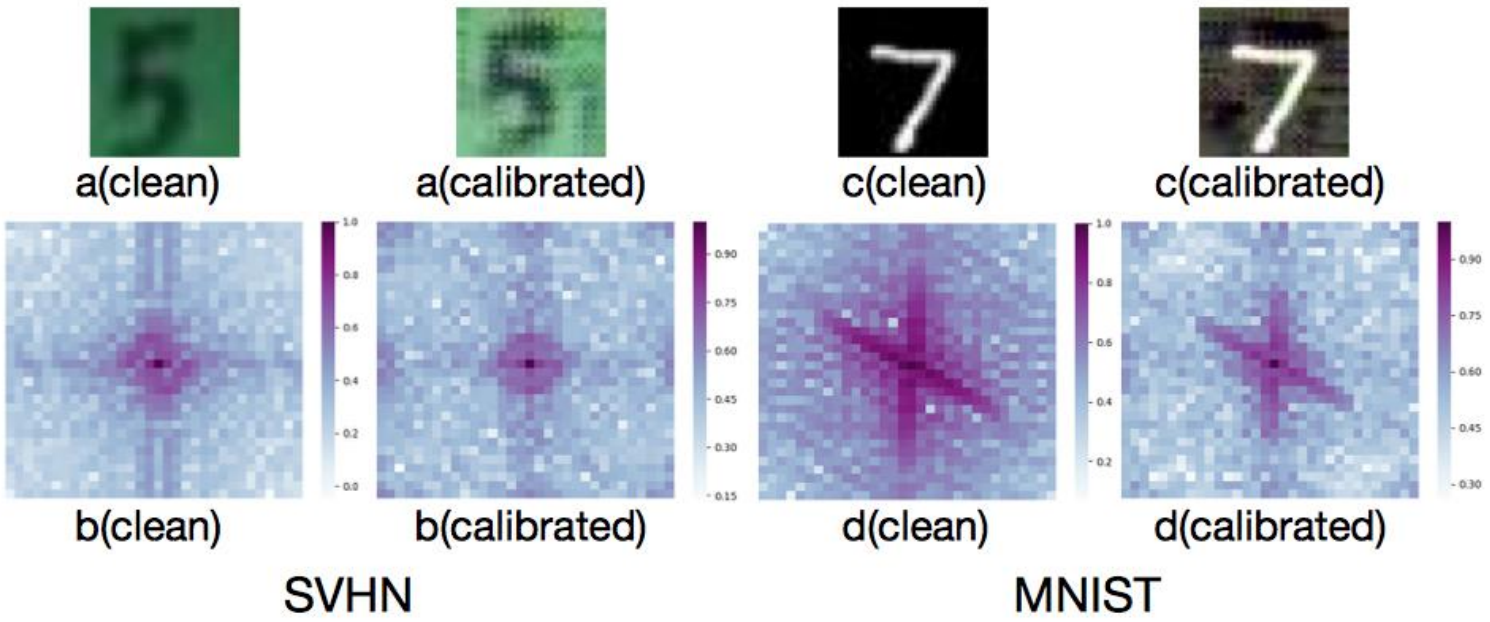
Table 2. **Adaptation between GTA5 and CityScapes**. Source only shows results of DRN-26 [43] trained in GTA5 and tested in CityScapes. Target only shows results of DRN-26 trained in CityScapes and tested in CityScapes. Our method outperforms CyCADA in mean IoU, freqency weighted IoU and pixel accuracy. In particular, our frequency weighted IoU is 2.7% better than CyCADA.

Semantic Segmentation



- The proposed method also achieves state-of-art performance in Semantic Segmentation task such as GTA5 to CityScapes, demonstrating that our method can be used in real world scenario.

Frequency Domain Interpretation



- We use Fast Fourier Transform (FFT) to visualize the images, before and after added the calibration. We show that in frequency domain, the data **calibrator reduces the high frequency information**, which is believed to be related to texture and noise.

Summary: Towards Accurate, Efficient and Robust Models

WX → Y : Self-Distillation, ICCV2019

Zhang, Linfeng, Jiebo Song, Anni Gao, Jingwei Chen, Chenglong Bao, and Kaisheng Ma. "Be Your Own Teacher: Improve the Performance of Convolutional Neural Networks via Self Distillation." *ICCV* (2019).

WX → Y : Scalable Neural Networks, NeuralIPS 2019

Zhang, Linfeng, Zhanhong Tan, Jiebo Song, Jingwei Chen, Chenglong Bao, and Kaisheng Ma. "SCAN: A Scalable Neural Networks Framework Towards Compact and Efficient Models." *NeuralIPS* (2019).

WX → Y : ADA: Augment Data Augmentation

Linfeng Zhang, Chenglong Bao, Kaisheng Ma. "ADA: Augment Data Augmentation by Relationship", *arxiv* (Under Review).

WX ----> Y : Auxiliary Training

Linfeng Zhang, Muzhou Yu, Tong Chen, Zuoqiang Shi, Chenglong Bao, Kaisheng Ma. "Auxiliary Training: Towards Accurate and Robust Models", *arxiv* (Under Review).

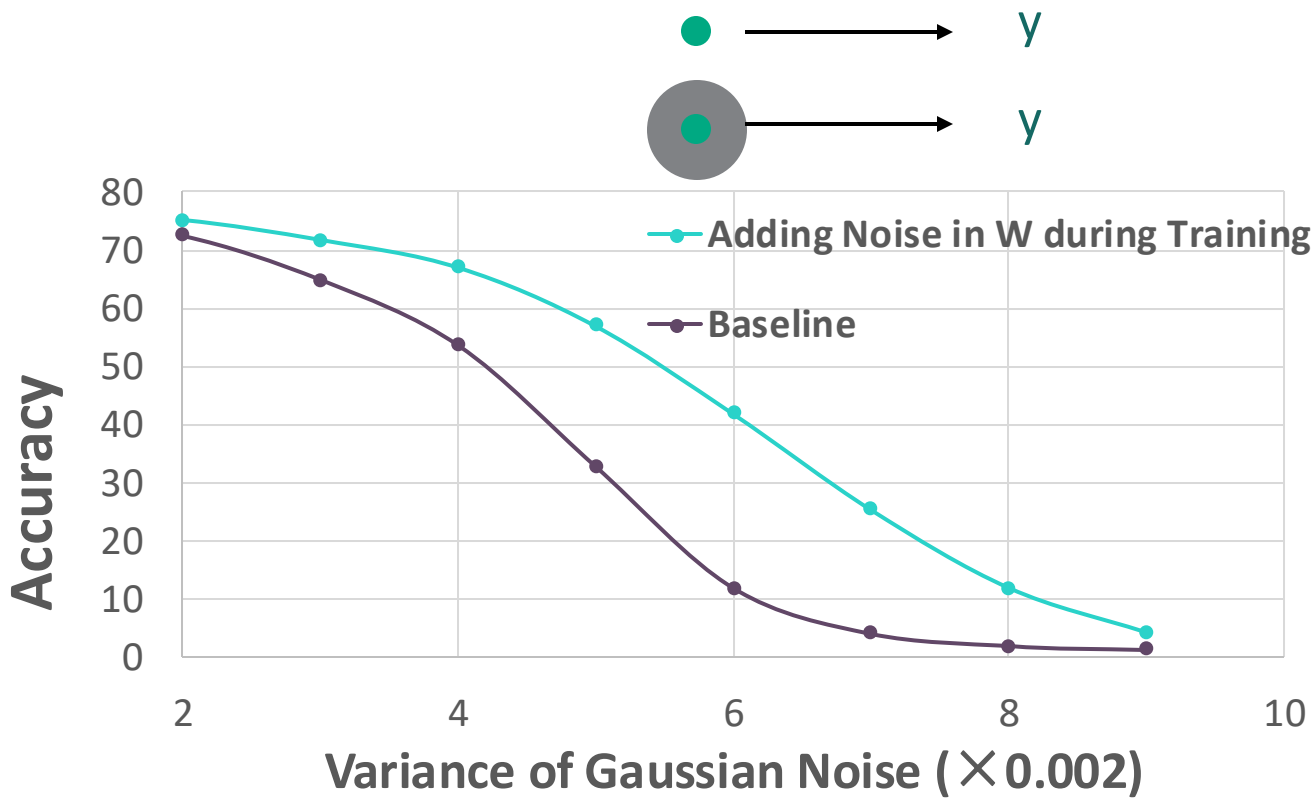
WX → Y : Data Calibrator

Shaokai Ye, Kailu Wu, Mu Zhou, Yunfei Yang, Sia Huat Tan, Jiebo Song, Kaidi Xu, Chenglong Bao, Kaisheng Ma. "Light-weight Calibrator: A Separable Component for Unsupervised Domain Adaptation", *arxiv* (Under Review).

Future Works

- Robust W.
- Good for W ONLY -> ReRAM Neuromorphic Computing, Subthreshold CMOS

$$(W+\sigma)X \rightarrow Y$$



Future Works

- Robust W.

$$(W\text{-}f(X))X \rightarrow Y$$

交叉信息研究院
Institute for Interdisciplinary Information Sciences

脑与智能实验室
ArChip Lab
Algorithm ARchitecture & Chipsets

Thanks !
Q&A

Main Contributors：
Kaisheng Ma, Linfeng Zhang, Shaokai Ye, Zhongfan Jia, Shengfa Chen @ IIIS THU
Chenglong Bao @ YMSC THU
Jingwei Chen @ HiSilicon