Knowledge Distillation

Chia-Chi Tsai (蔡家齊)
cctsai@gs.ncku.edu.tw
AI System Lab
Department of Electrical Engineering
National Cheng Kung University

Outline

- What is Knowledge Distillation
- What to Match
- Self and Online Distillation
- Distillation for Different Tasks
- Network Augmentation, a Training Technique for Tiny Machine Learning Models

Outline

- What is Knowledge Distillation
- What to Match
- Self and Online Distillation
- Distillation for Different Tasks
- Network Augmentation, a Training Technique for Tiny Machine Learning Models

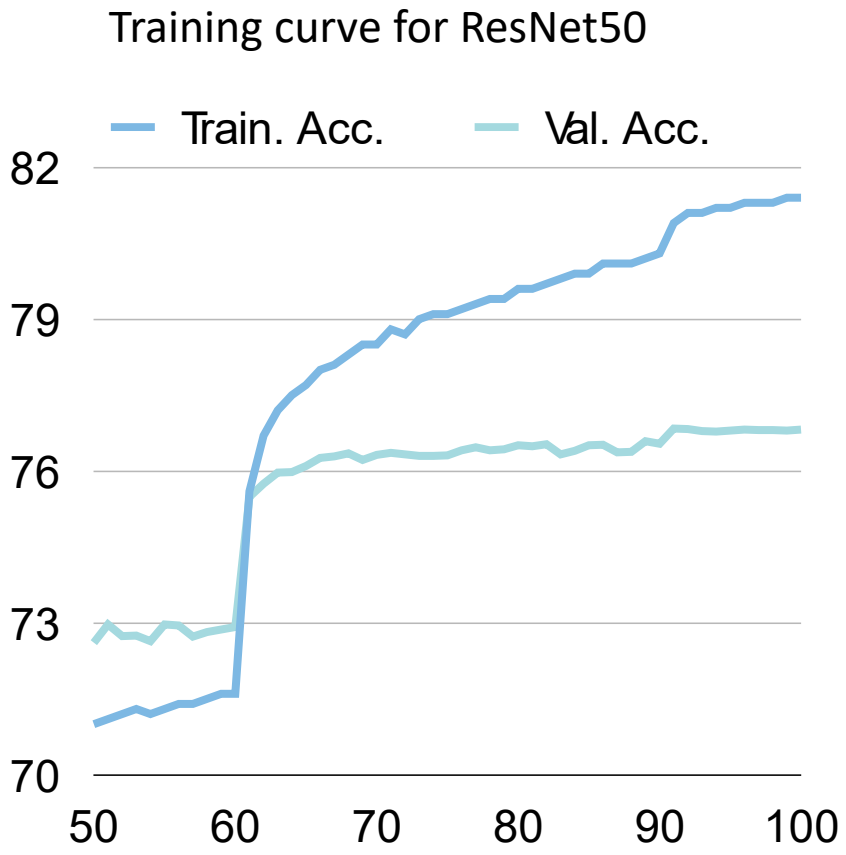
AI is Quickly Coming to the Edge

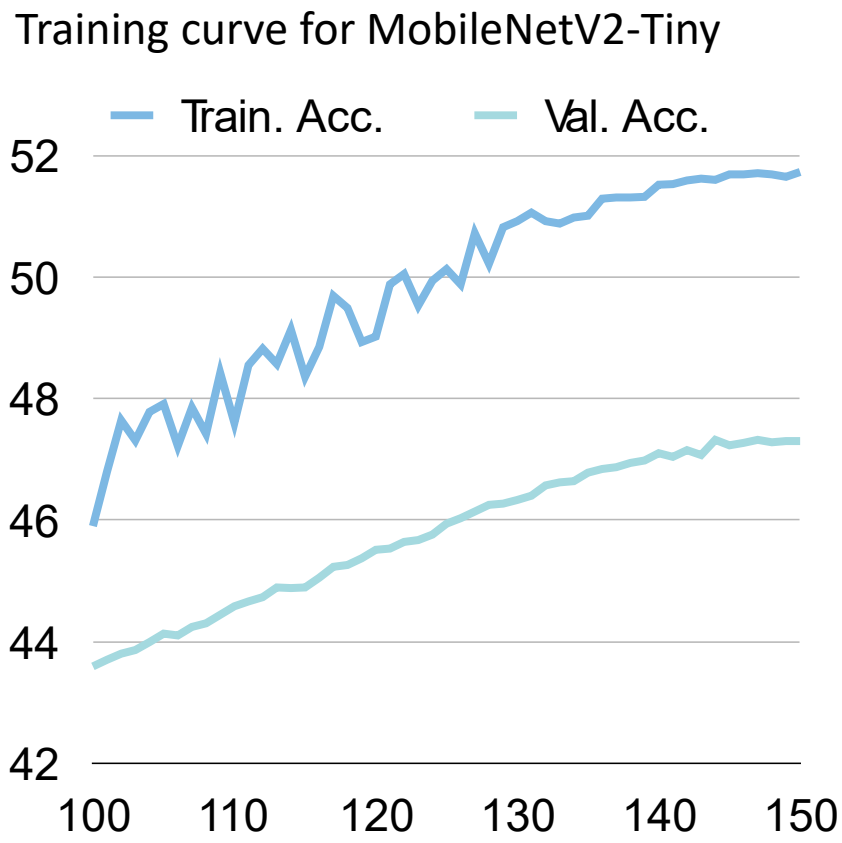
	Cloud AI	Mobile AI	Tiny AI
Computation	XX TOPs	X TOPs	XX~XXX MOPs
Memory(Activation)	32G	4GB	320KB
Storage(Weights)	~TB/PB	256GB	1mb

Neural network must be tiny to run efficiently on tiny edge devices

Tiny Models Are Hard to Train

- Tiny models underfit large datasets





Can we help the training of tiny models with large models?

Distilling the Knowledge in a Neural Network

Distilling the Knowledge in a Neural Network

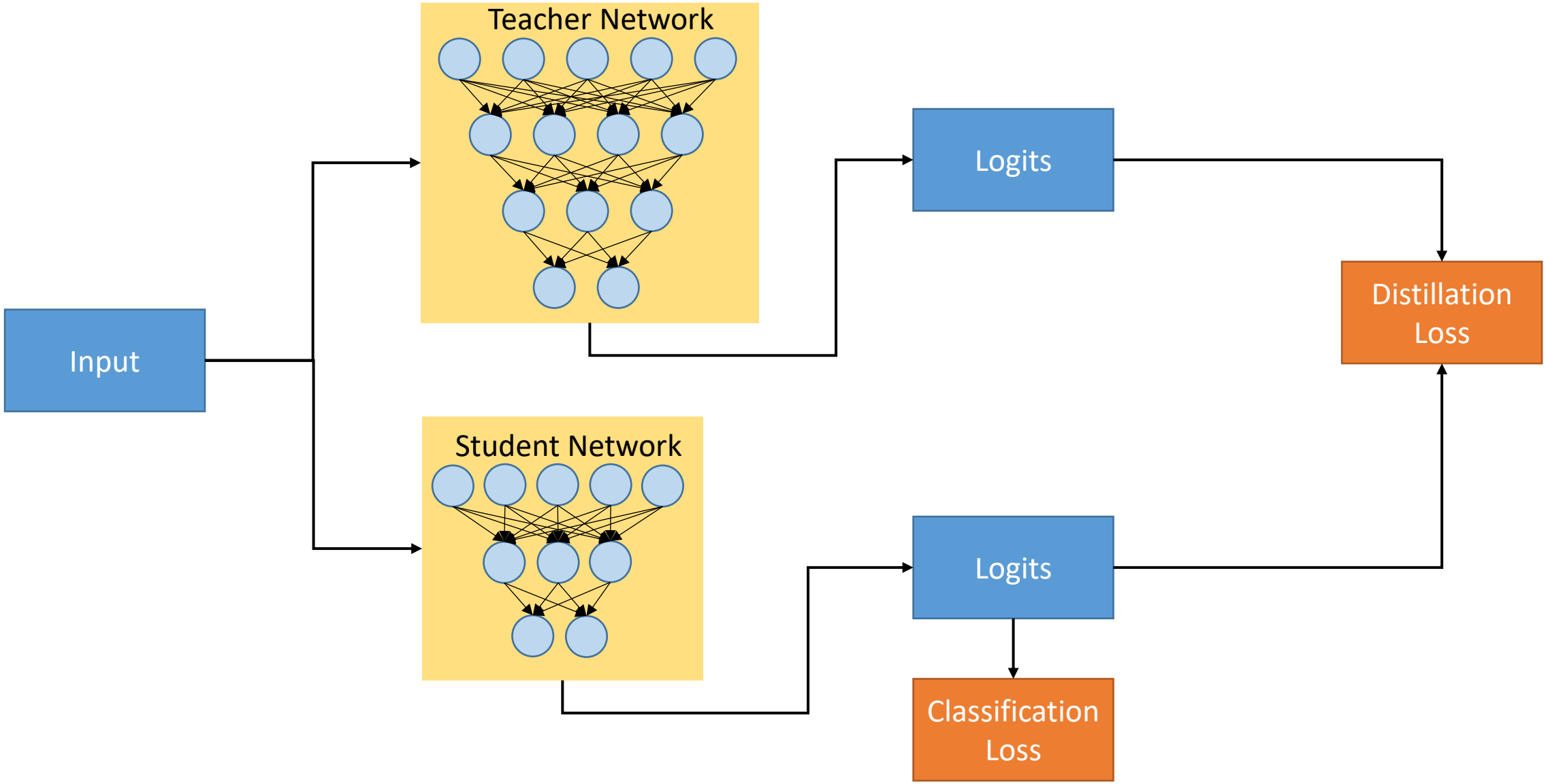
Geoffrey Hinton[*†]
Google Inc.
Mountain View
geoffhinton@google.com

Oriol Vinyals[†]
Google Inc.
Mountain View
vinyals@google.com

Jeff Dean
Google Inc.
Mountain View
jeff@google.com

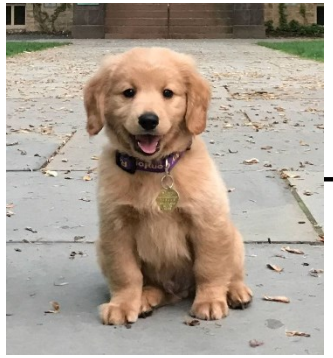
Hinton, G., Vinyals, O., & Dean, J. (2015). Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, *2*(7).

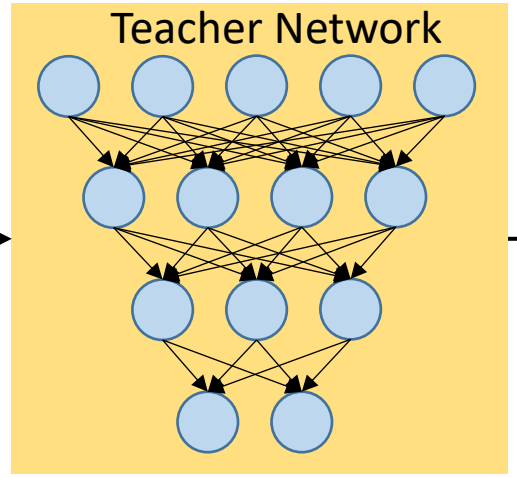
Illustration of Knowledge Distillation



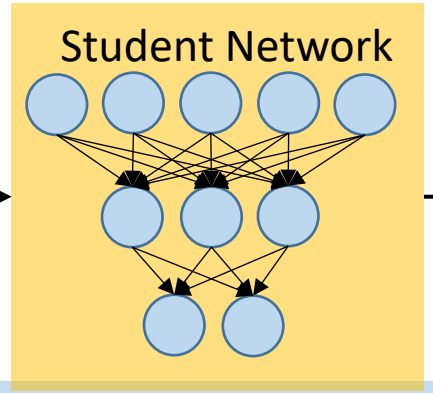
Intuition of Knowledge Distillation

- Matching prediction probabilities between teacher and student





	Logits	Probabilities
Cat	1	$\frac{e^1}{e^5+e^1} = 0.017$
Dog	5	$\frac{e^5}{e^5+e^1} = 0.982$

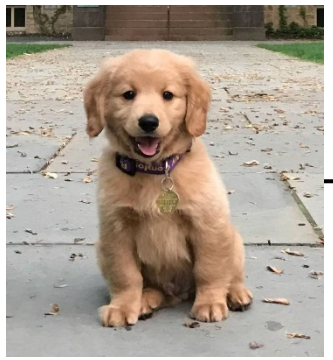


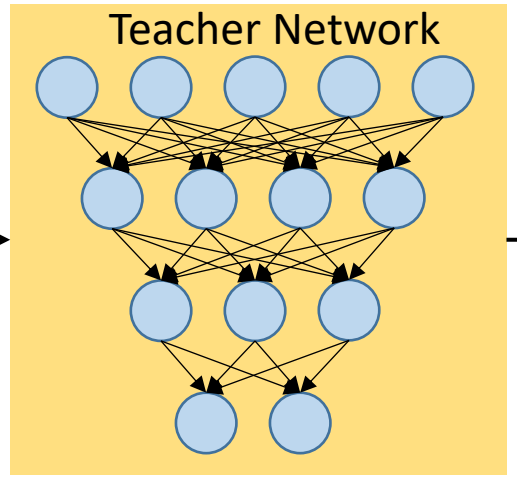
	Logits	Probabilities
Cat	2	0.269 ↓
Dog	3	0.731 ↑

The student model is less confident

Concept of Temperature

- A larger temperature smooths the output probability distribution





	Logits	Probabilities (T=1)	Probabilities (T=10)
Cat	1	$\frac{e^{\frac{1}{1}}}{e^{\frac{5}{1}}+e^{\frac{1}{1}}} = 0.017$	$\frac{e^{\frac{1}{10}}}{e^{\frac{5}{10}}+e^{\frac{1}{10}}} = 0.401$
Dog	5	$\frac{e^{\frac{5}{1}}}{e^{\frac{5}{1}}+e^{\frac{1}{1}}} = 0.982$	$\frac{e^{\frac{5}{10}}}{e^{\frac{5}{10}}+e^{\frac{1}{10}}} = 0.599$

Formal Definition of KD

- Neural networks typically use a softmax function to generate the **logits** z_i to class **probabilities** $p(z_i, T)$
  - $i, j = 0,1,2 \ldots C-1$
  - Where T is the temperature, which is normally set to 1

- The goal of knowledge distillation is to **align the class probability distributions from teacher and student networks**

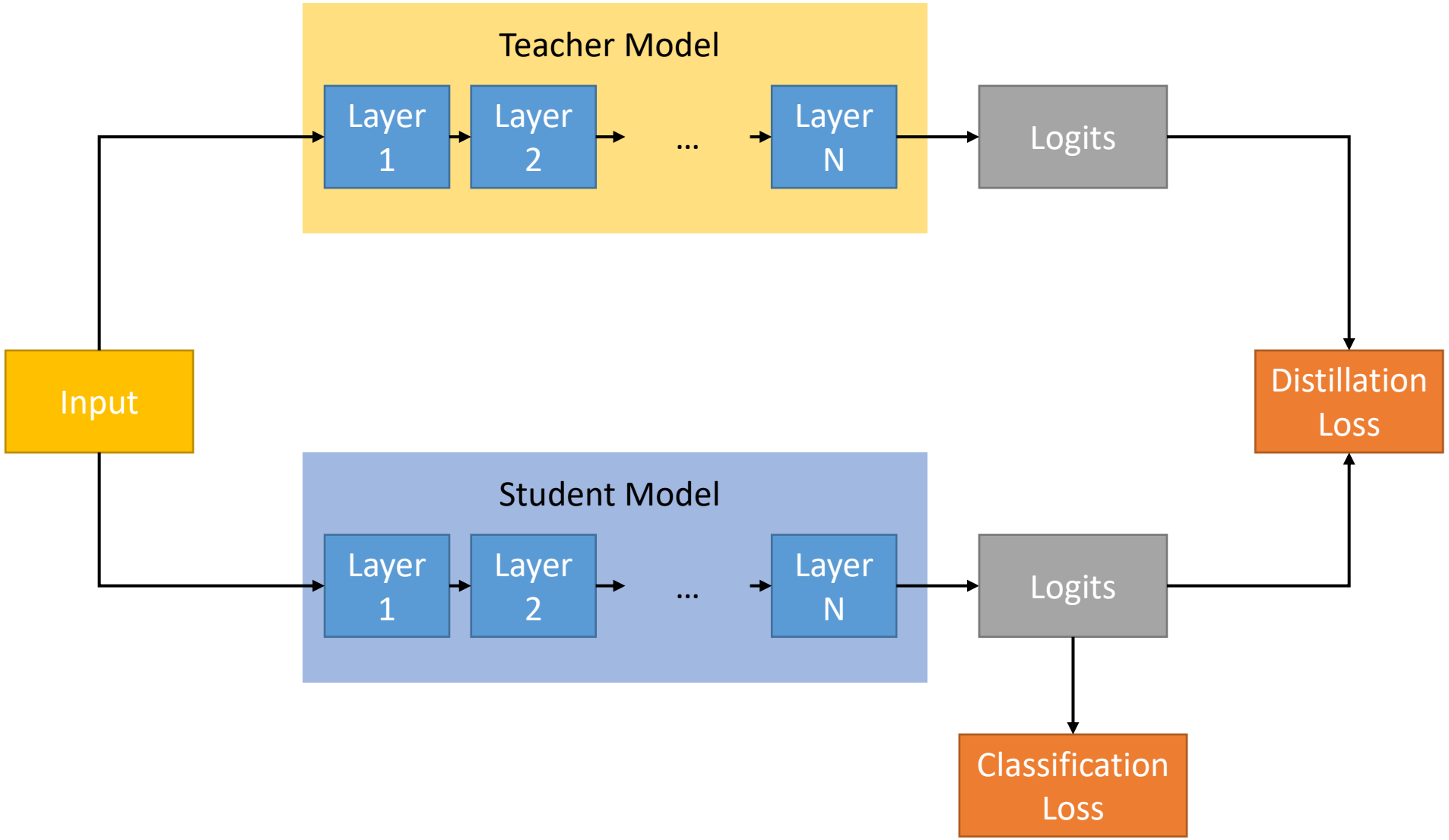
Outline

- What is Knowledge Distillation
- What to Match
- Self and Online Distillation
- Distillation for Different Tasks
- Network Augmentation, a Training Technique for Tiny Machine Learning Models

What to Match?

- Output logits
- Intermediate weights
- Intermediate features
- Gradients
- Sparsity patterns
- Relational information

Matching Output Logits



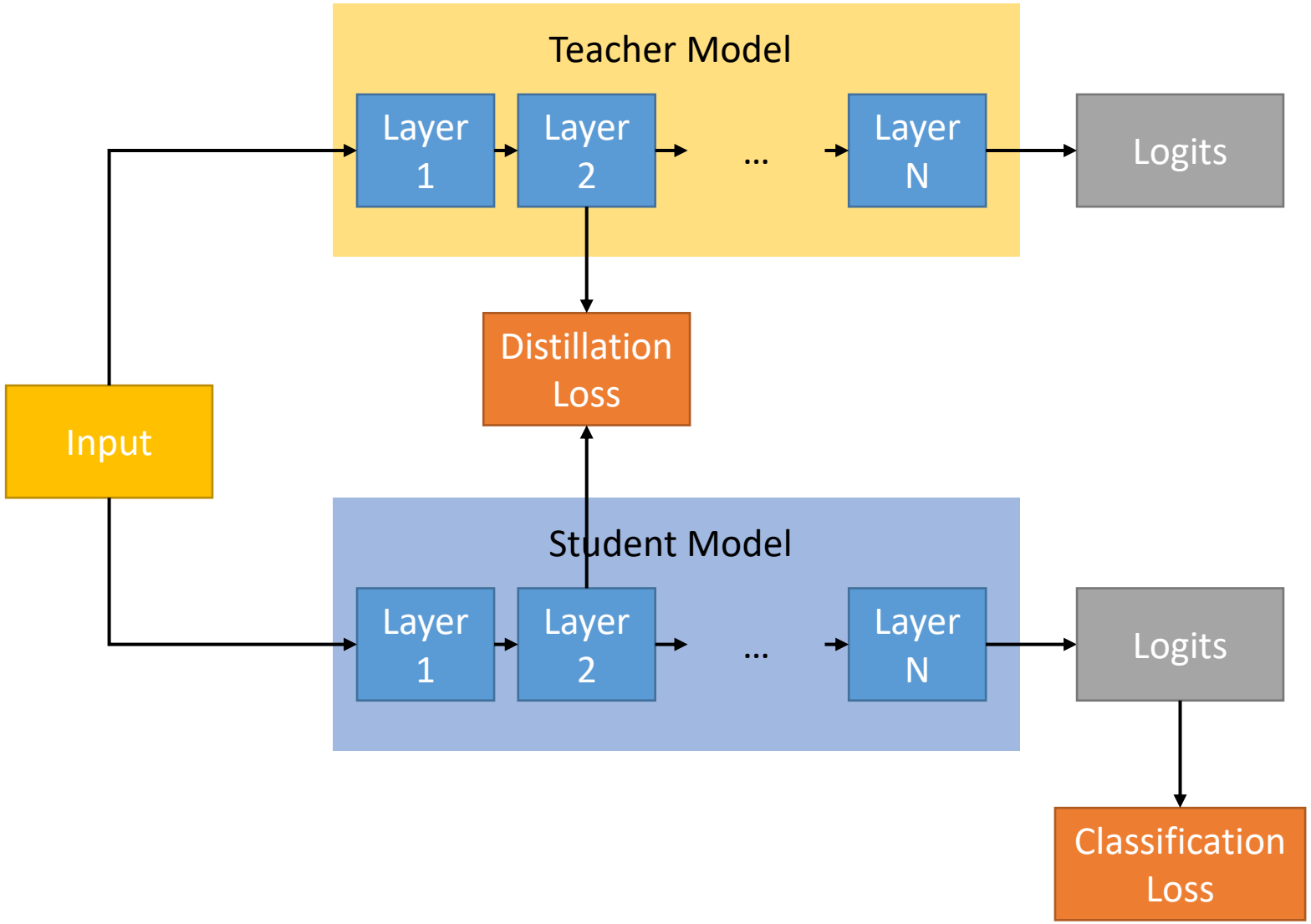
Cross entropy loss

$E(-p_t \log p_s)$

L2 loss

$E(\|p_t - p_s\|_2^2)$

Matching Intermediate Weights



Cross entropy loss

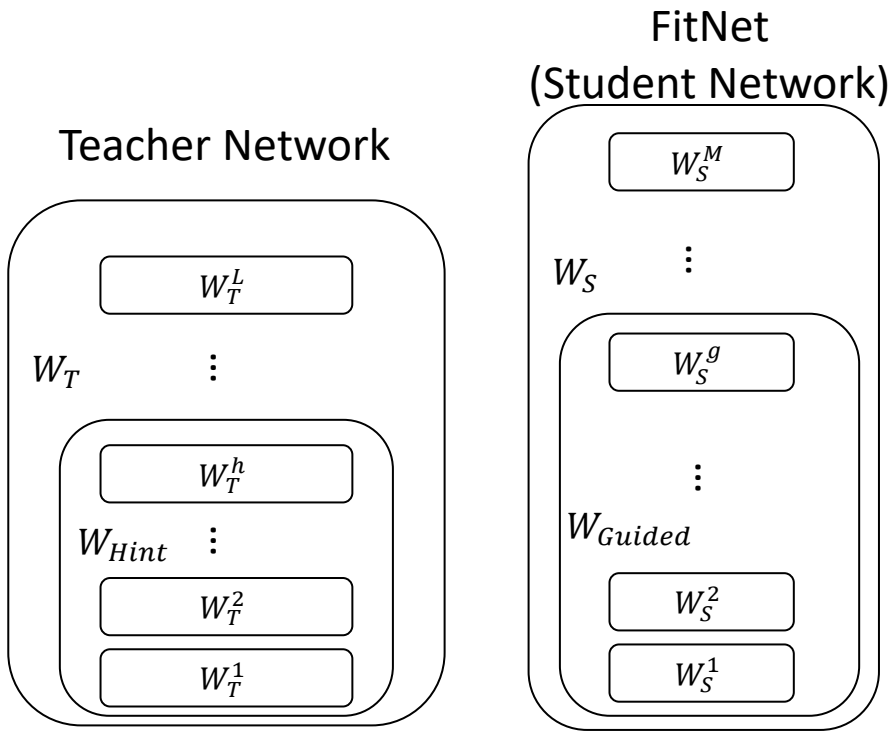
$E(-p_t \log p_s)$

L2 loss

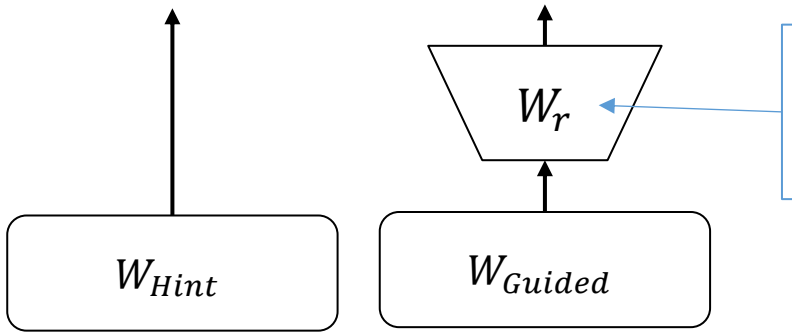
$E(\|p_t - p_s\|_2^2)$

Example – FitNets

- Other than the cross-entropy distillation loss, also add a L2 loss between teacher weights and student weights
  - Linear transformation is applied to match the dimensionalities



$$W^*_{Guided} = \arg\min_{W_{Guided}} \mathcal{L}_{HT}(W_{guided}, W_r)$$

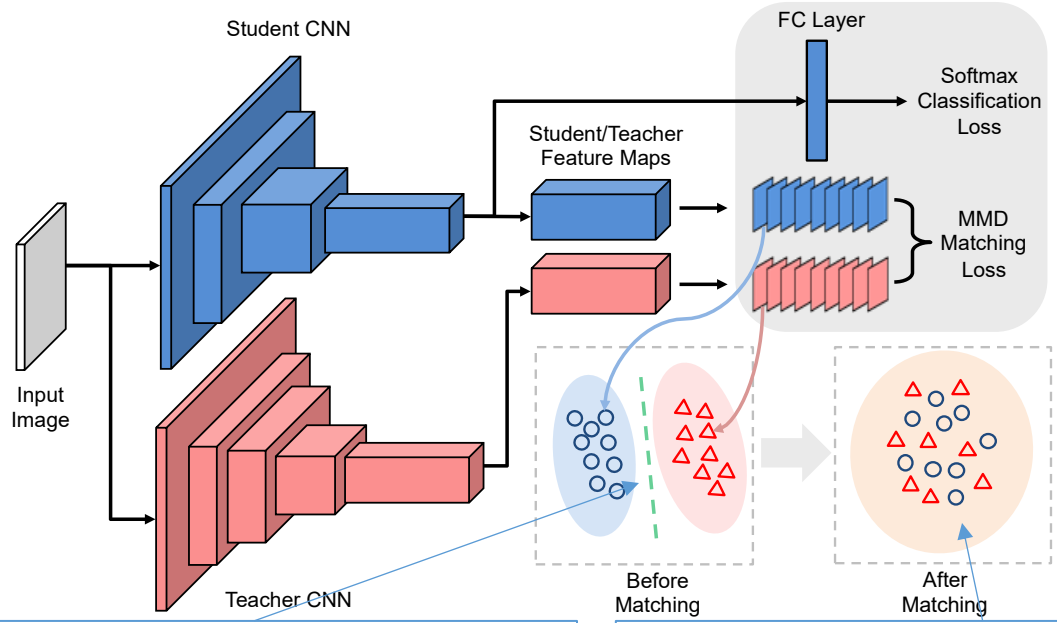


Hint Training

Romero, A., Ballas, N., Kahou, S. E., Chassang, A., Gatta, C., & Bengio, Y. (2015). Fitnets: Hints for thin deep nets. In ICLR.

Matching Intermediate Features

- Minimizing maximum mean discrepancy between feature maps
- Example - Neuron selectivity transfer
  - Intuition
    - Teacher and student networks should have **similar** feature distributions
    - Not just output probability distributions



Teacher and student have very different feature distributions without distillation

With the distillation objective, teacher and student feature distributions are similar

Use maximum mean discrepancy(MMD) as an objective, k is the kernel function, and its simplest form is the dot product

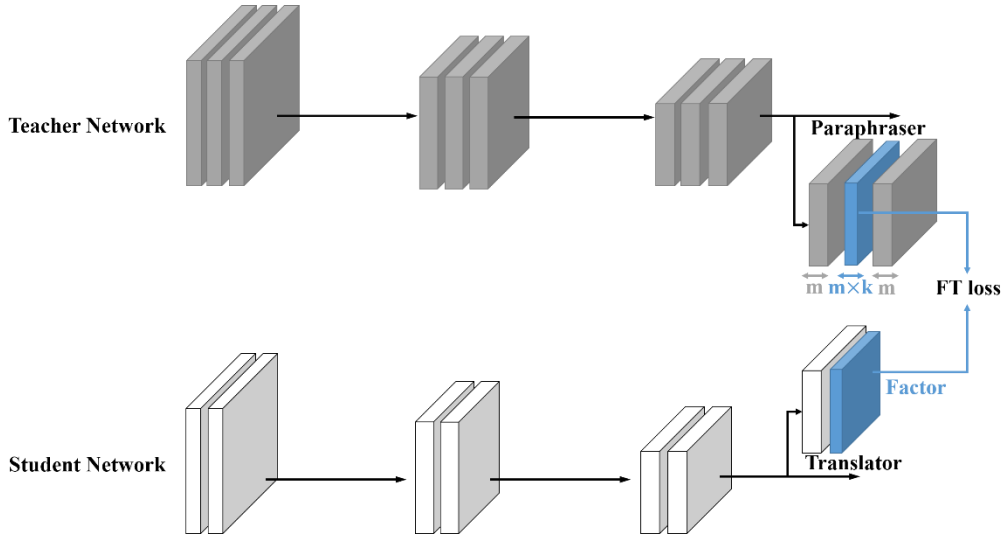
$$L_{MMD^2}(F_T, F_S) = \frac{1}{C_T^2}\sum_{i=1}^{C_T}\sum_{i'=1}^{C_T} k\left(\frac{f_T^{i\cdot}}{\left\|f_T^{i\cdot}\right\|_2}, \frac{f_T^{i'\cdot}}{\left\|f_T^{i'\cdot}\right\|_2}\right) + \frac{1}{C_S^2}\sum_{j=1}^{C_T}\sum_{j'=1}^{C_T} k\left(\frac{f_S^{j\cdot}}{\left\|f_S^{j\cdot}\right\|_2}, \frac{f_S^{j'\cdot}}{\left\|f_S^{j'\cdot}\right\|_2}\right) - \frac{2}{C_T C_S}\sum_{i=1}^{C_T}\sum_{j=1}^{C_S} k\left(\frac{f_T^{i\cdot}}{\left\|f_T^{i\cdot}\right\|_2}, \frac{f_S^{j'\cdot}}{\left\|f_S^{j'\cdot}\right\|_2}\right)$$

Cosine of angle between teacher/student feature vector

Huang, Z., & Wang, N. (2017). Like what you like: Knowledge distill via neuron selectivity transfer. *arXiv preprint arXiv:1707.01219*.

Matching Intermediate Features

- Minimizing the L2 distance between feature maps
- Example - Factor Transfer
  - The paraphraser shrinks the output teacher feature map from dimensions to $m \times k$ dimensions (called **factor**, typically k=0.5) and then expands the dimensionality back to m
  - The output of paraphraser is supervised with a **reconstruction loss** against the original m-dimensional output
  - The student uses one layer of MLP to obtain a **factor** with the same dimensionality of $m \times k$
  - FT minimizes the distance between teacher and student factors



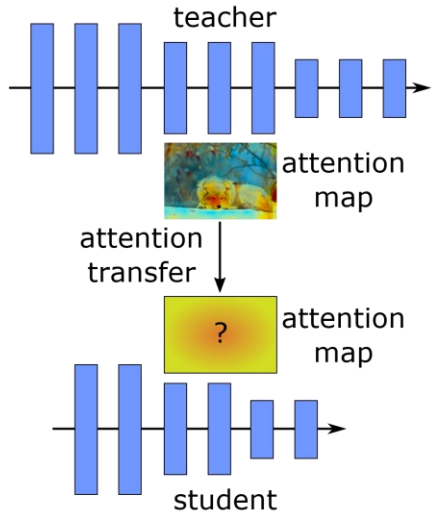
Kim, J., Park, S., & Kwak, N. (2018). Paraphrasing complex network: Network compression via factor transfer. *Advances in neural information processing systems*, *31*.

Matching Gradients

- Intermediate attention maps
  - Gradients of feature maps are used to characterize **attention** of DNNs
- The attention of a CNN feature map is defined as $\frac{\partial L}{\partial x}$
  - L: learning objective
- Example - Paying more attention to attention
  - Intuition
    - If $\frac{\partial L}{\partial x_{i,j}}$ is large, a small perturbation at i, j will significantly impact the final output.
    - As a result, the network is putting more attention on position i, j

input image

attention map



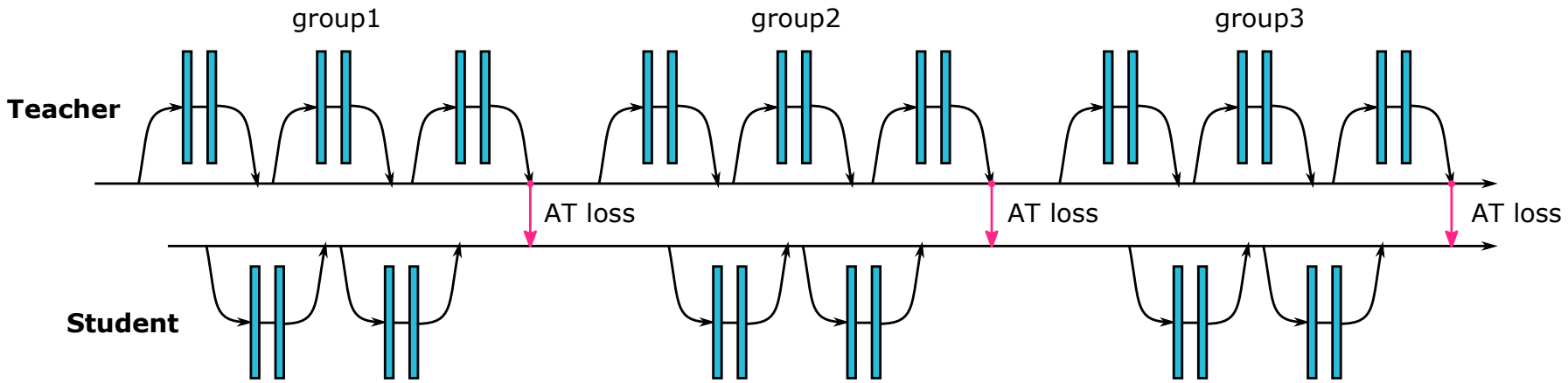
S. Zagoruyko and N. Komodakis, "Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer," in Proc. Int. Conf. Learn. Representations, 2017.

Matching Intermediate Attention Maps

- The attention transfer objective is defined as

$$\frac{\beta}{2}\|J_S - J_T\|_2^2$$

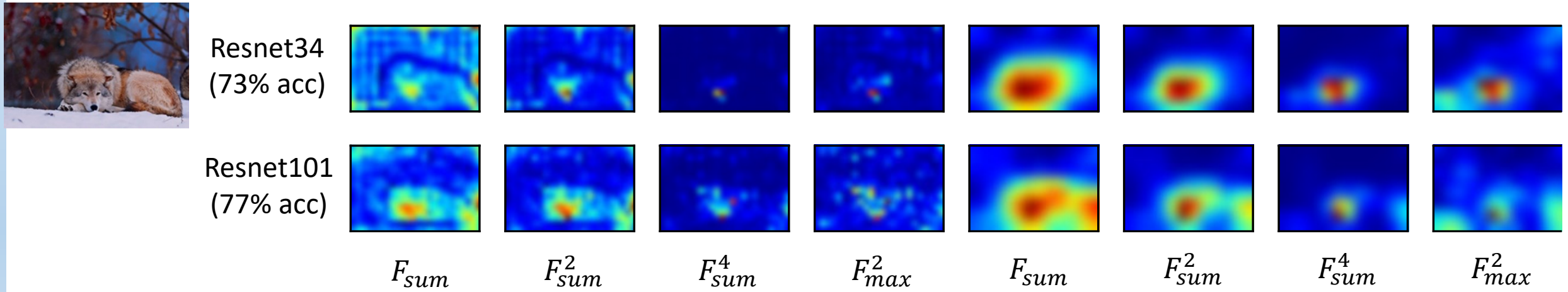
  - J_s: the student attention map (gradient of student feature map)
  - J_T: the teacher attention map.
  - β: a constant



Intermediate Attention Maps

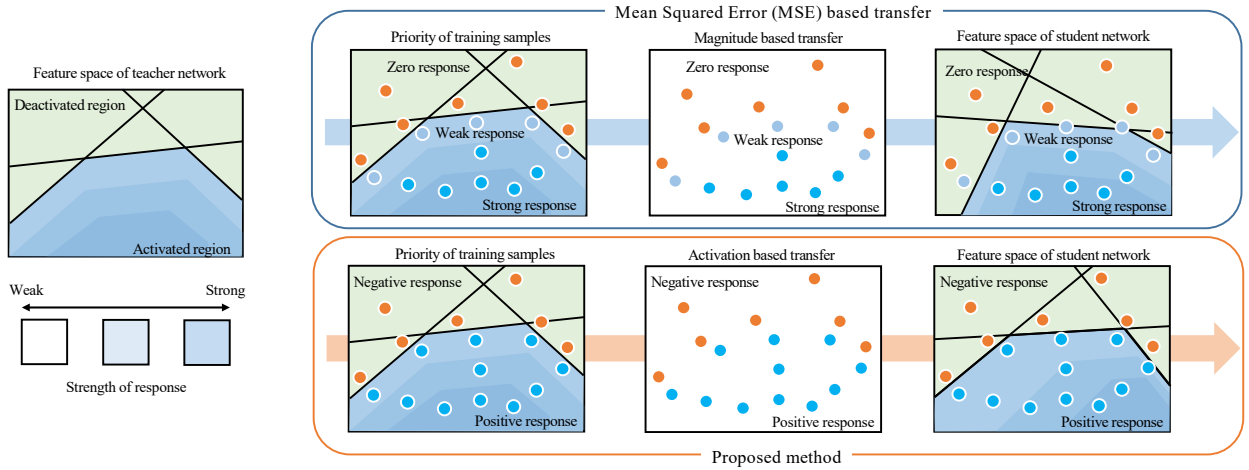
- Performant models have similar attention maps
  - Attention maps of performant ImageNet models (ResNets) are indeed similar to each other
  - However, the less performant model (NIN) has quite different attention maps

$$F_{sum}(A) = \sum_{i=1}^{C} |A_i| \qquad F_{sum}^{p}(A) = \sum_{i=1}^{C} |A_i|^p \qquad F_{max}^{p}(A) = \max_{i=1,C} |A_i|^p$$



Matching Sparsity Patterns

- Intuition
  - The teacher and student networks should have similar sparsity patterns after the ReLU activation
    - A neuron is **activated** after ReLU if its value is larger than 0, denoted by the indicator function $\rho(x) = 1[x > 0]$
- We want to minimize $\mathcal{L}(I) = \left\| \rho\left(T(I)\right) - \rho\left(S(I)\right) \right\|_1$
  - Where S and T corresponds to student and teacher networks, respectively



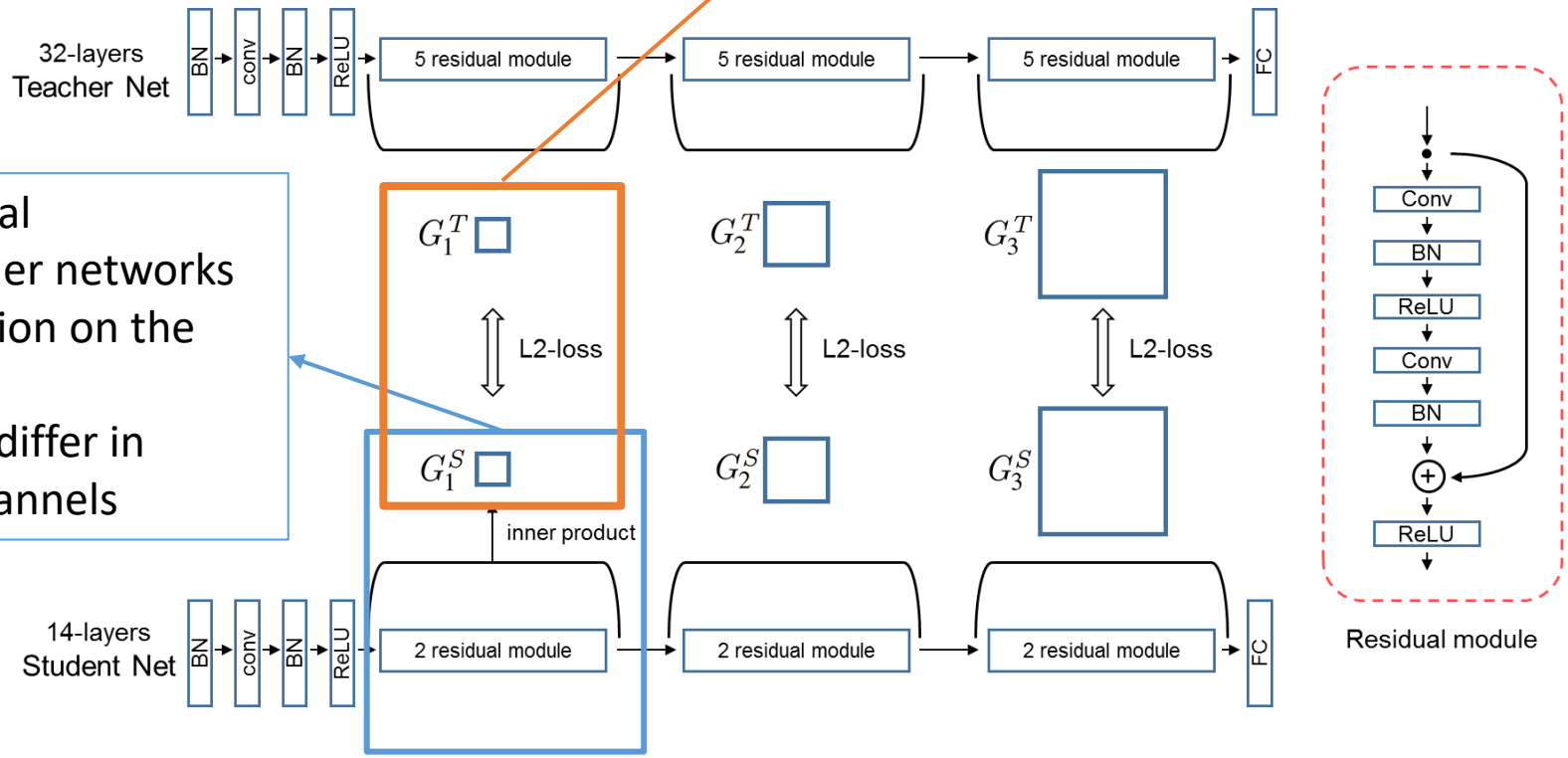
Heo, B., Lee, M., Yun, S., & Choi, J. Y. (2019, July). Knowledge transfer via distillation of activation boundaries formed by hidden neurons. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 33, No. 01, pp. 3779-3787).

Matching Relational Information

- Relations between different layers

2. Then match the resulting dot products between teacher and student networks (G_1^T, G_1^S)

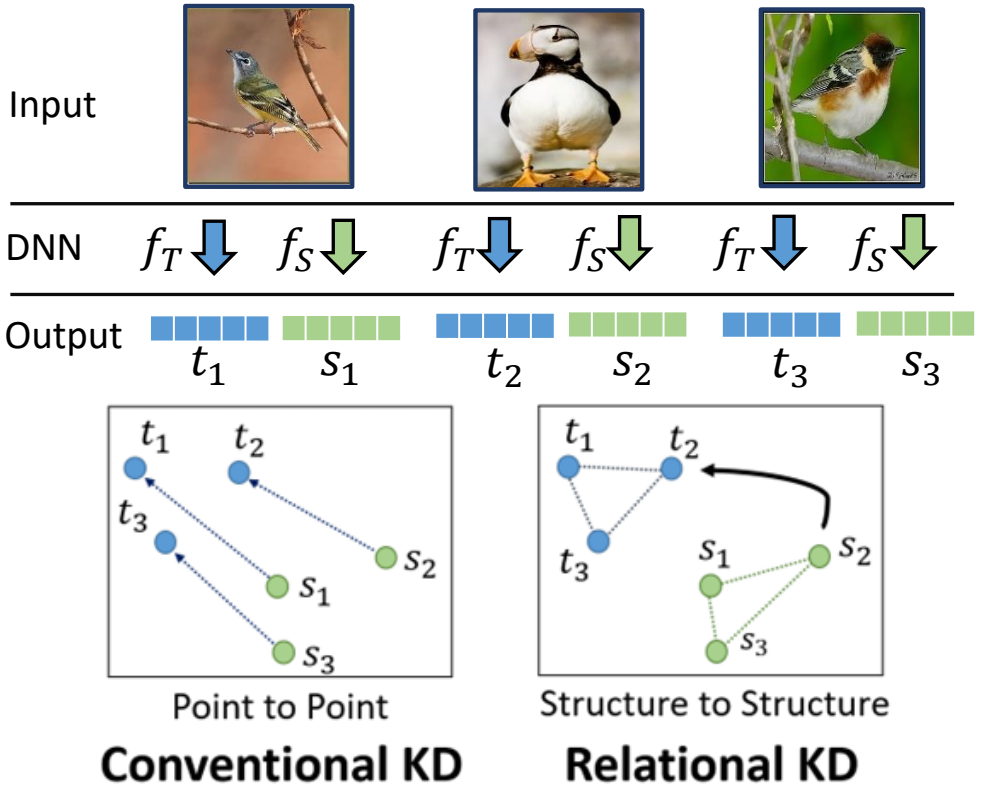
1. Use inner product to extract relational information for both student and teacher networks
   - A matrix of shape $C_{in} \times C_{out}$ reduction on the spatial dimensions
   - Student and teacher networks only differ in number of layers, not number of channels



Yim, J., Joo, D., Bae, J., & Kim, J. (2017). 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).

Matching Relational Information

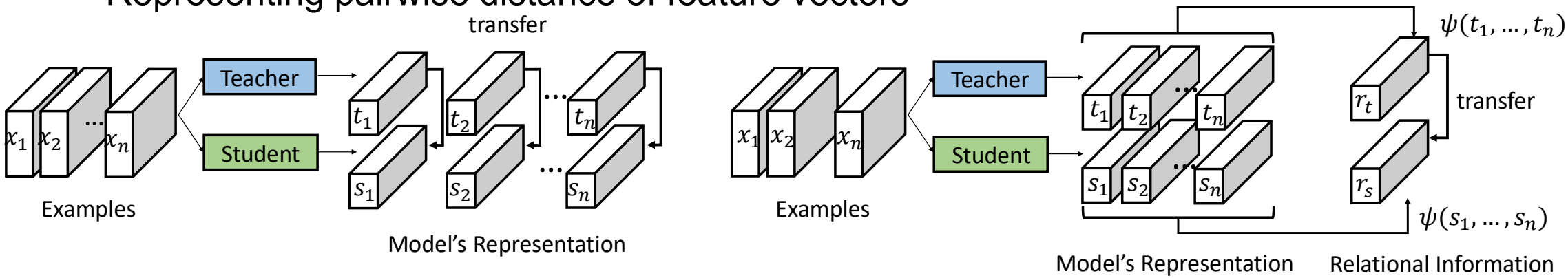
- Relations between different samples
  - Conventional KD focuses on matching feature /logit for **one input**
  - Relational KD looks at the **relation** between intermediate features from **multiple inputs**



Park, W., Kim, D., Lu, Y., & Cho, M. (2019). Relational knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 3967-3976).

Matching Relational Information

- Relations between different samples
  - $\psi(s_1, s_2, \dots, s_n) = (\|s_1 - s_2\|_2^2, \|s_1 - s_3\|_2^2, \dots, \|s_1 - s_n\|_2^2, \dots, \|s_{n-1} - s_n\|_2^2)$
    - A vector of length $\frac{n(n-1)}{2}$
    - Representing pairwise distance of feature vectors



Individual Knowledge Distillation

Representative method: FSP(Flow of Solution Procedure)
Relation **within** the model, calculated **separately** for each input

Relational Knowledge Distillation

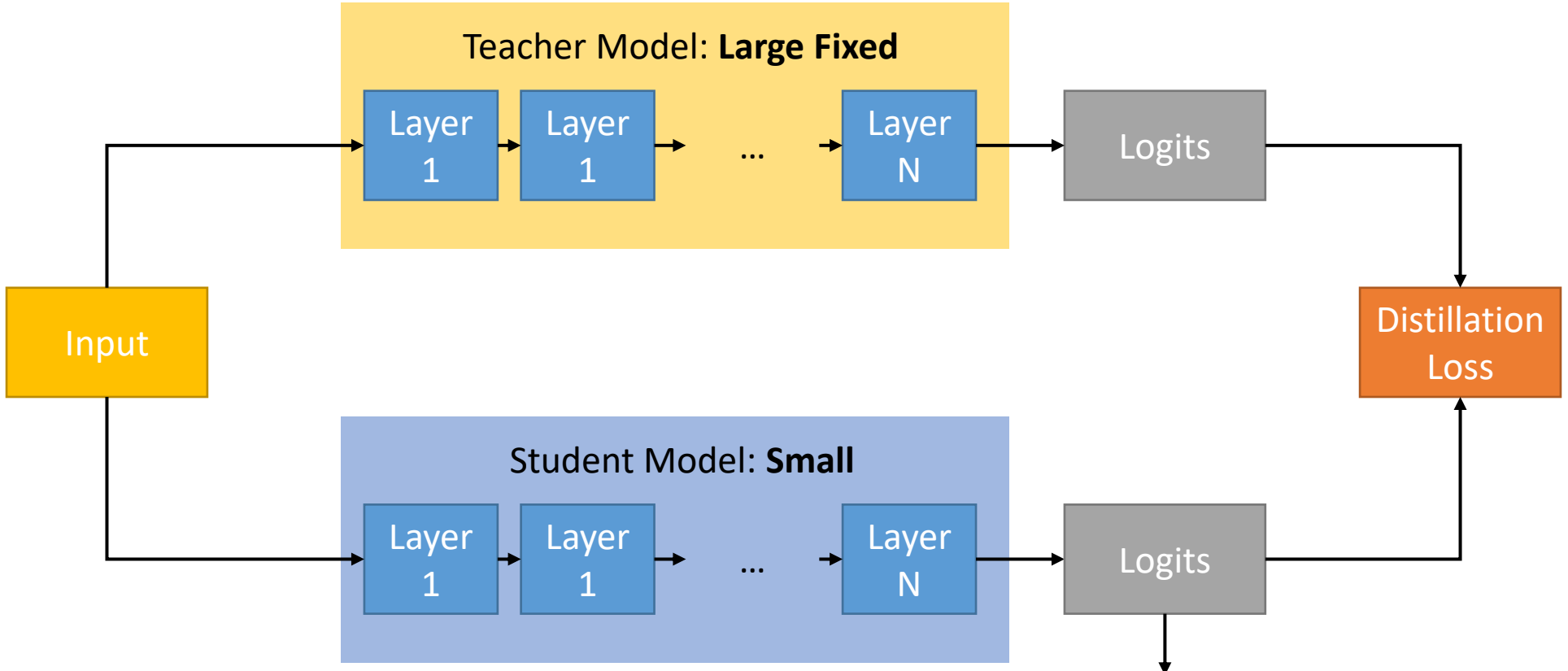
Representative method: RKD
Relation **across different samples**

Outline

- What is Knowledge Distillation
- What to Match
- Self and Online Distillation
- Distillation for Different Tasks
- Network Augmentation, a Training Technique for Tiny Machine Learning Models

Overview of Knowledge Distillation

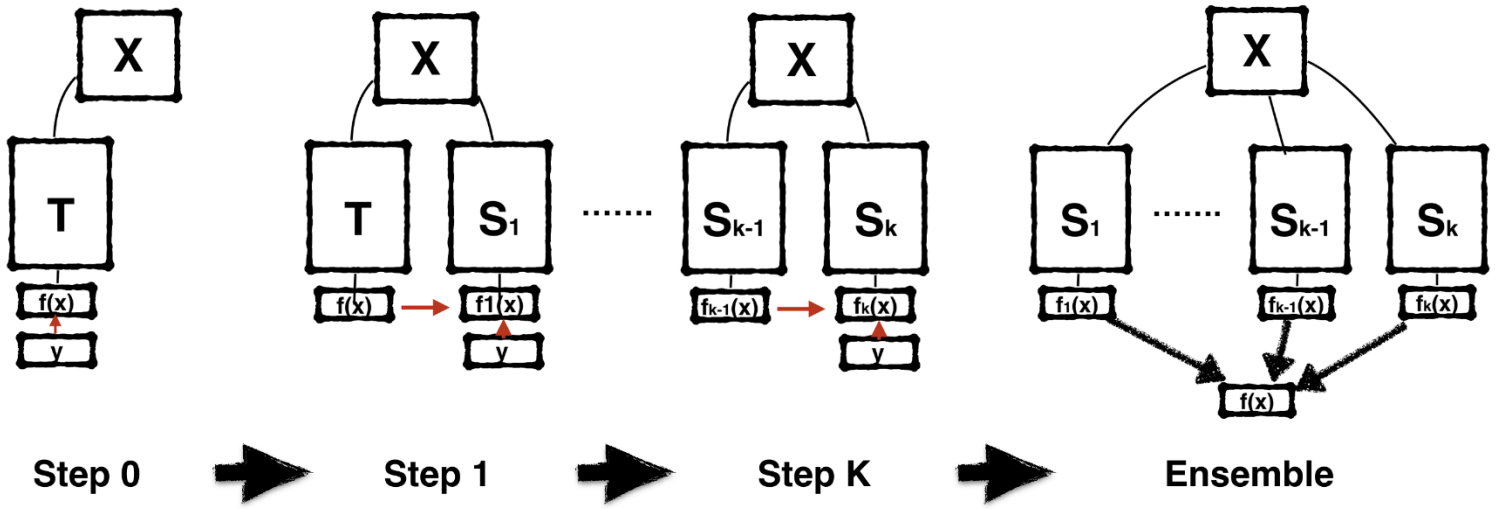
- Teacher model is usually larger than the student model and is fixed



What is the disadvantage of fixed large teachers?
Does it have to be the case that we need a fixed large teacher in KD?

Self-Distillation with Born-Again NNs

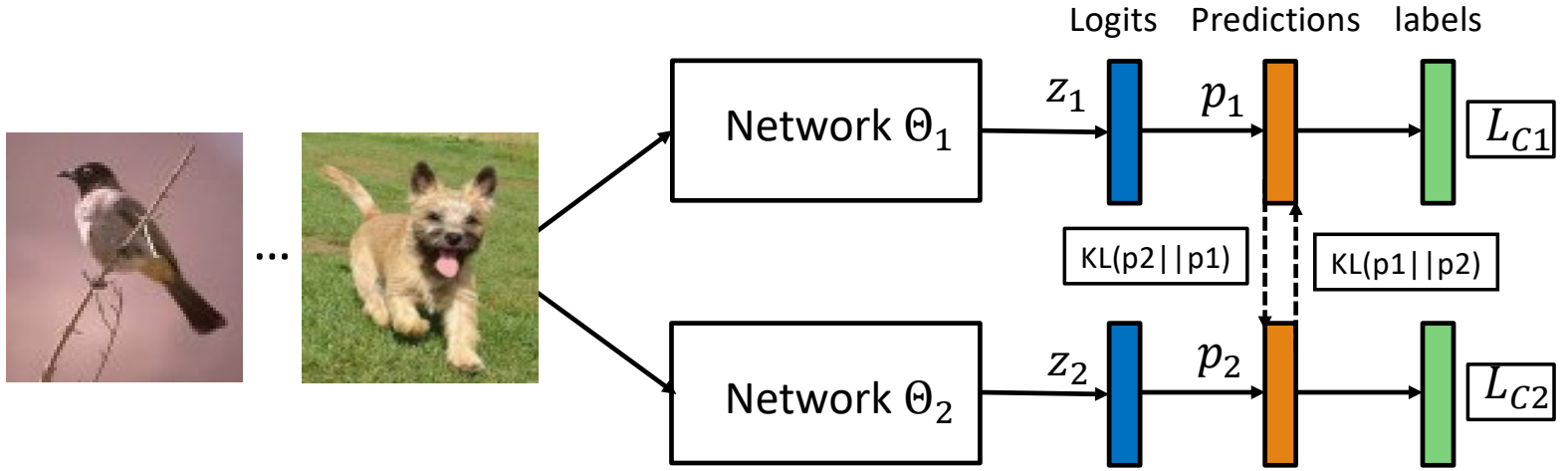
- Born-Again Networks generalizes defensive distillation by
  - Adding iterative training stages
  - Using both classification objective and distillation objective in subsequent stages
- Network architecture $T = S_1 = S_2 = \cdots = S_k$
- Network accuracy$T < S_1 < S_2 < \cdots < S_k$
- Can alternatively ensemble $T, S_1, S_2, \ldots, S_k$ to get even better performance



Furlanello, T., Lipton, Z., Tschannen, M., Itti, L., & Anandkumar, A. (2018, July). Born again neural networks. In *International Conference on Machine Learning* (pp. 1607-1616). PMLR.

Online Distillation

- Example - Deep Mutual Learning
  - For both teacher and student networks, we want to add a distillation objective that minimizes the output distribution of the other party
  - $\mathcal{L}(S) = CrossEntropy(S(I), y) + KL\big(S(I), T(I)\big)$
  - $\mathcal{L}(T) = CrossEntropy(T(I), y) + KL\big(T(I), S(I)\big)$
  - Note: it is not necessary to pretrain T, and $S = T$ is allowed



Θ_1, Θ_2 can be the same or different, and they are trained from scratch

Zhang, Y., Xiang, T., Hospedales, T. M., & Lu, H. (2018). Deep mutual learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4320-4328).

Online Distillation

- Deep Mutual Learning (DML)
  - Deep mutual learning can improve both student (net 2) and teacher (net 1) models

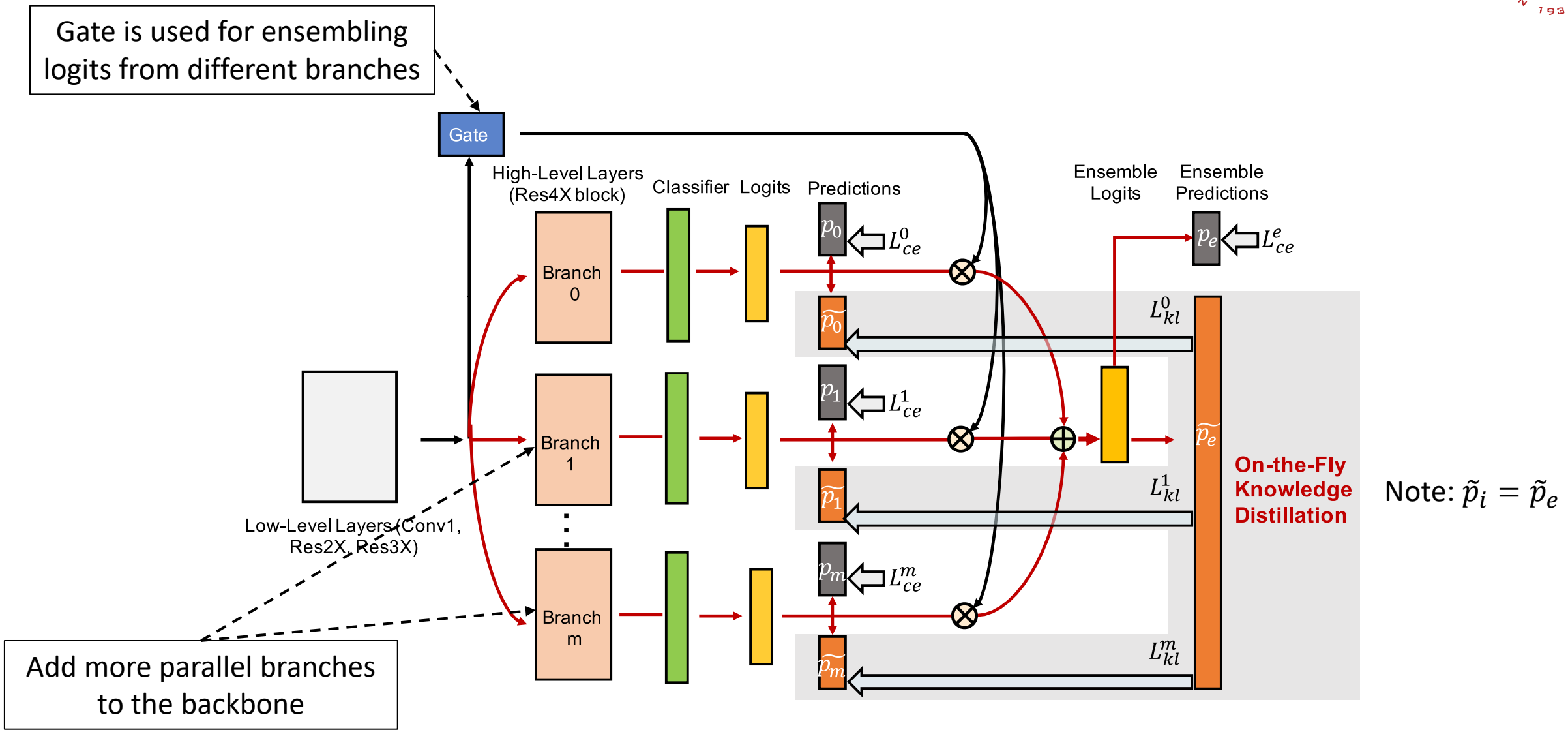
Network Types		CIFAR-10						CIFAR-100					
		Independent		DML		DML-Ind		Independent		DML		DML-Ind	
Net 1	Net 2	Net 1	Net 2	Net 1	Net 2	Net 1	Net 2	Net 1	Net 2	Net 1	Net 2	Net 1	Net 2
Resnet-32	Resnet-32	92.47	92.47	92.68	92.80	0.21	0.33	68.99	68.99	71.19	70.75	2.20	1.76
WRN-28-10	Resnet-32	95.01	92.47	95.75	93.18	0.74	0.71	78.69	68.99	78.96	70.73	0.27	1.74
MobileNet	Resnet-32	93.59	92.47	94.24	93.32	0.65	0.85	73.65	68.99	76.13	71.10	2.48	2.11
MobileNet	MobileNet	93.59	93.59	94.10	94.30	0.51	0.71	73.65	73.65	76.21	76.10	2.56	2.45
WRN-28-10	MobileNet	95.01	93.59	95.73	94.37	0.72	0.78	78.69	73.65	80.28	77.39	1.59	3.74
WRN-28-10	WRN-28-10	95.01	95.01	95.66	95.63	0.65	0.62	78.69	78.69	80.28	80.08	1.59	1.39

Combining Online and Self-Distillation

- Example – ONE: **O**n-the-Fly **N**ative **E**nsemble as the teacher network
  - Idea
    - Generating multiple output probability distributions and ensemble them as the target distribution for knowledge distillation
  - Similar to DML, ONE allows the teacher model to be exactly the same as the student model
    - It does not require pretraining the teacher network first
    - It is also not necessary to train two models as in DML

Zhu, X., & Gong, S. (2018). Knowledge distillation by on-the-fly native ensemble. *Advances in neural information processing systems*, *31*.

ONE



Note: $\tilde{p}_i = \tilde{p}_e$

ONE – Result

Error rates on ImageNet

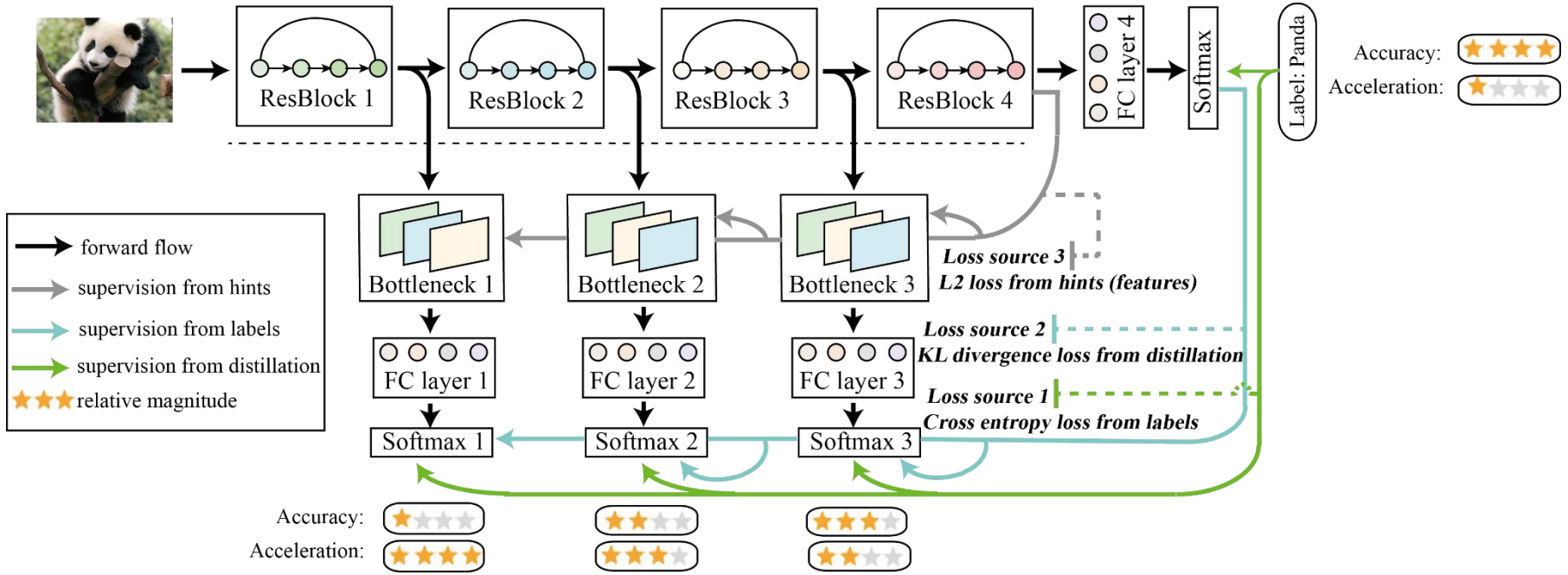
Method	Top-1	Top-5
ResNet-18 [4]	30.48	10.98
ResNet-18 + **ONE**	**29.45±0.23**	**10.41±0.12**
ResNeXt-50 [23]	22.62	6.29
ResNeXt-50 + **ONE**	**21.85±0.07**	**5.90±0.05**
SeNet-ResNet-18 [31]	29.85	10.72
SeNet-ResNet-18 + **ONE**	**29.02±0.17**	**10.13±0.12**

Comparison with DML
(TrCost: training cost, TeCost: testing cost)

Target Network	ResNet-32			ResNet-110		
Metric	Error (%)	TrCost	TeCost	Error (%)	TrCost	TeCost
KD [10]	**28.83**	6.43	1.38	N/A	N/A	N/A
DML [17]	29.03±0.22*	**2.76**	1.38	**24.10±0.72**	**10.10**	5.05
ONE	**26.61±0.06**	**2.28**	1.38	**21.62±0.26**	**8.29**	5.05

Combining Online and Self-Distillation

- Example - Be Your Own Teacher: deep supervision + distillation
  - Use deeper layer to distill shallower layers
  - **Intuition**
    - Labels at later stages are more reliable, so the author use them to supervise the prediction from the previous stages



Zhang, L., Song, J., Gao, A., Chen, J., Bao, C., & Ma, K. (2019). Be your own teacher: Improve the performance of convolutional neural networks via self distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 3713-3722).

Be Your Own Teacher – Result

- Result on CIFAR100 shows consistent performance improvements over the baseline
- Predictions from intermediate classifiers (1/4, 2/4, 3/4) can sometimes outperform the baseline
  - Inference efficiency can be improved.

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

Outline

- What is Knowledge Distillation
- What to Match
- Self and Online Distillation
- Distillation for Different Tasks
- Network Augmentation, a Training Technique for Tiny Machine Learning Models

KD for Object Detection

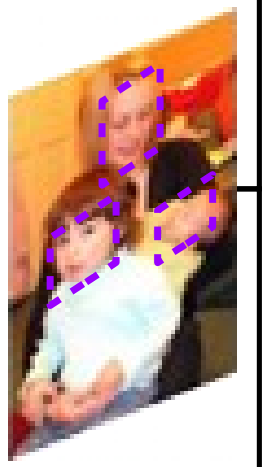
- Feature Imitation

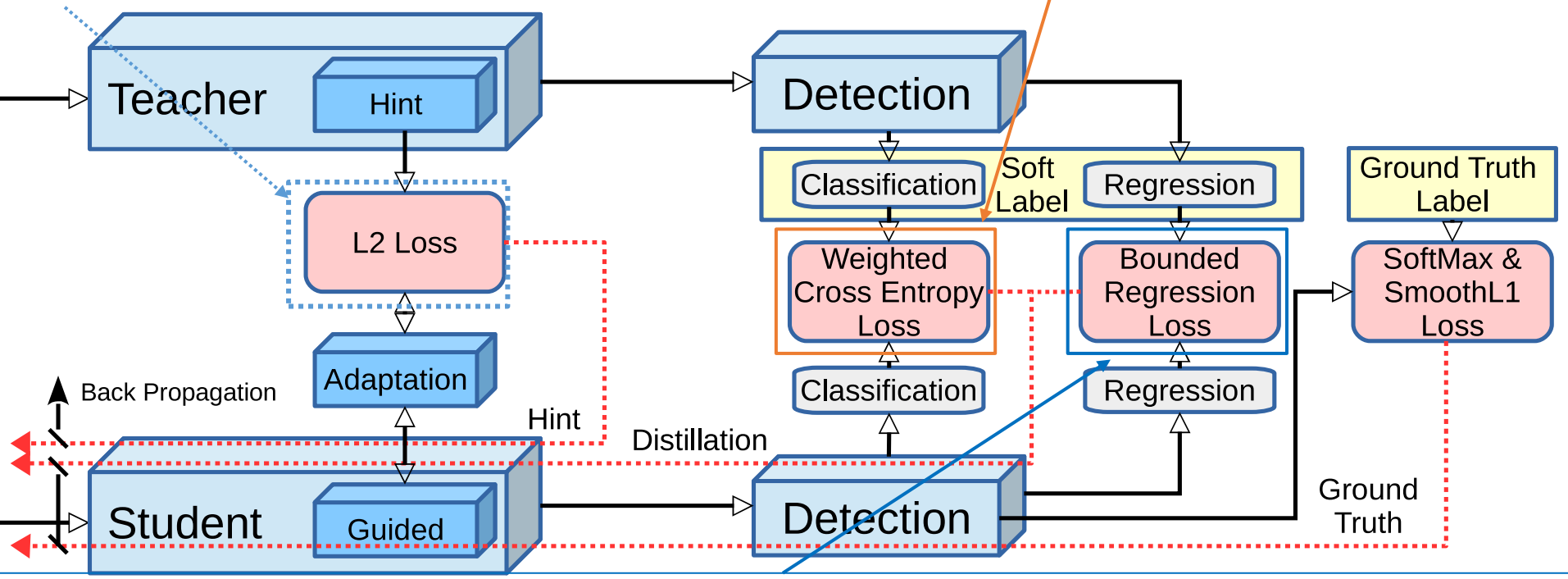
$$L_{Hint}(V, Z) = \|V - Z\|_2^2$$

Add a 1x1 convolution layer to match the shape

$$L_{soft}(P_s, P_t) = \sum w_c P_t \log P_s$$

Use different weights for foreground and background classes to handle the class imbalance problem





$$L_b(R_s, R_t, y) = \begin{cases} \|R_s - y\|_2^2, & if \|R_s - y\|_2^2 + m > \|R_t - y\|_2^2 \\ 0, & otherwise \end{cases}$$

Exploit teacher's prediction as an upper bound for the student to achieve. Once the quality of the student surpasses that of the teacher with a certain margin, the loss becomes zero

Chen, G., Choi, W., Yu, X., Han, T., & Chandraker, M. (2017). Learning efficient object detection models with knowledge distillation. *Advances in neural information processing systems*, *30*.

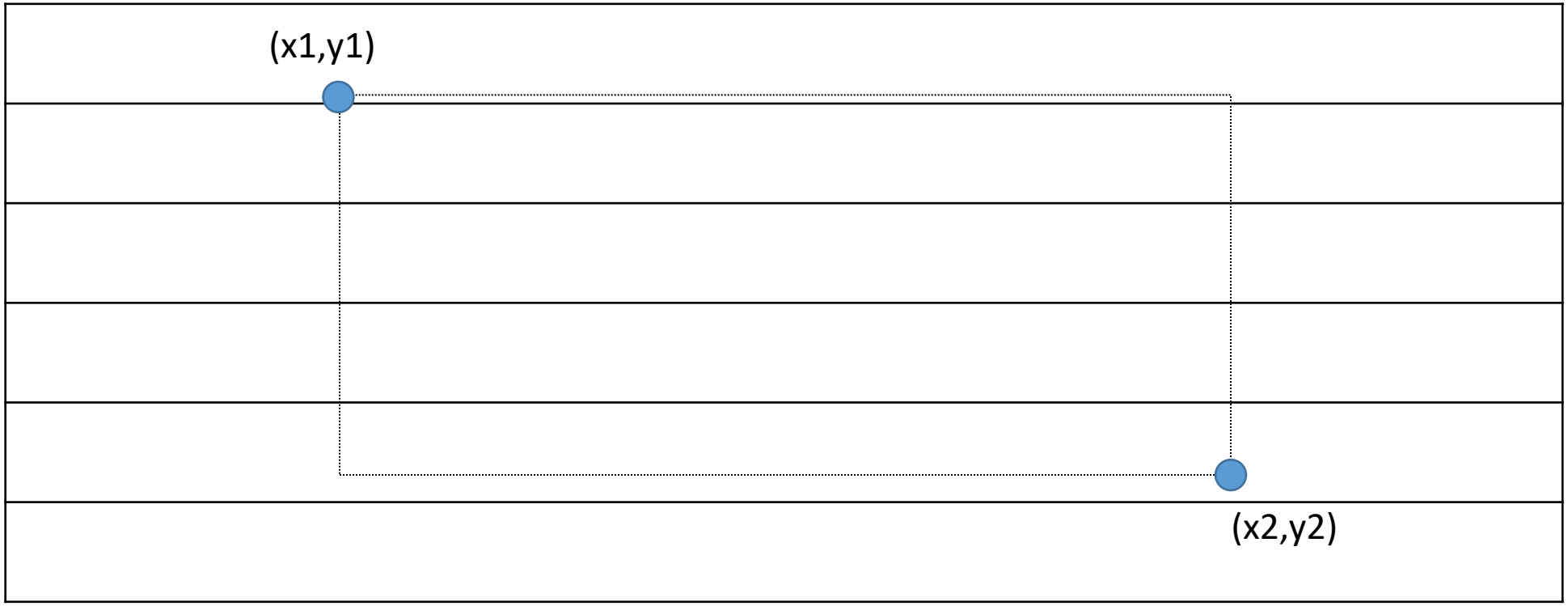
KD for Object Detection

- Convert bounding box regression to classification problem

Zheng, Z., Ye, R., Wang, P., Ren, D., Zuo, W., Hou, Q., & Cheng, M. M. (2022). Localization Distillation for Dense Object Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 9407-9416).

KD for Object Detection

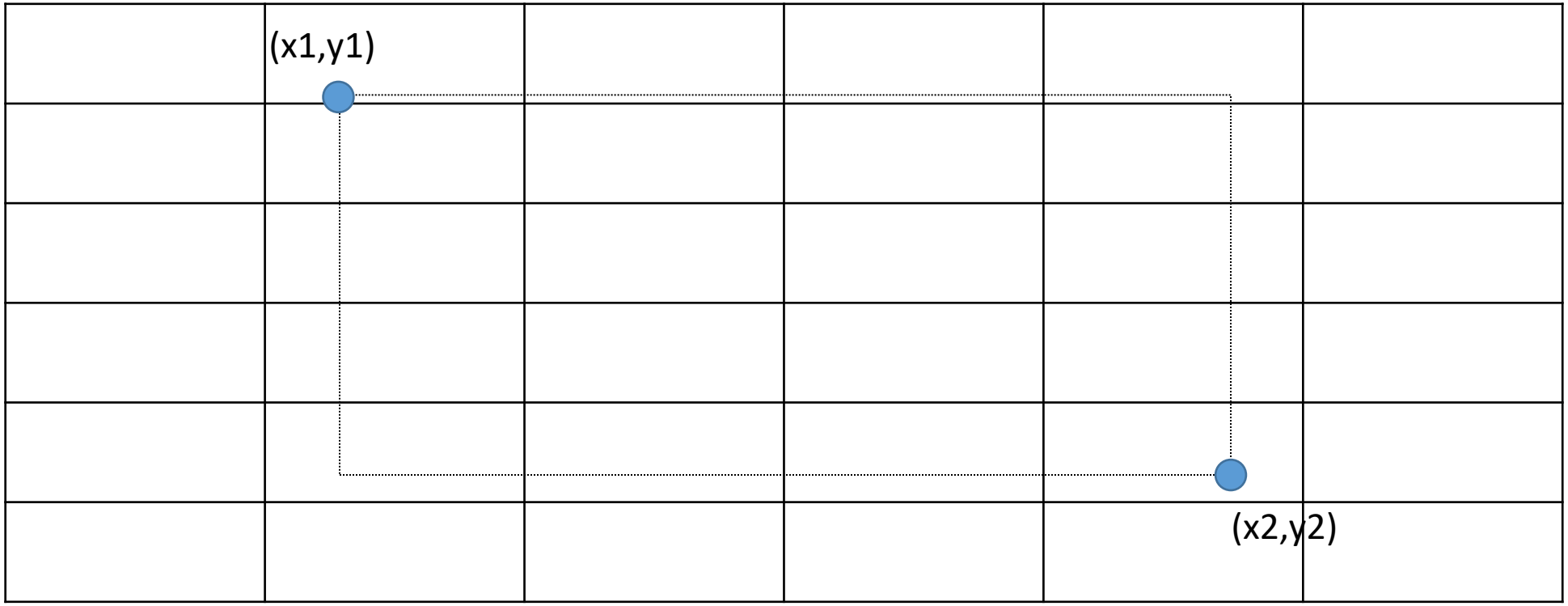
- Convert bounding box regression to classification problem
  - Divide the y-axis into 6 bins



Zheng, Z., Ye, R., Wang, P., Ren, D., Zuo, W., Hou, Q., & Cheng, M. M. (2022). Localization Distillation for Dense Object Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 9407-9416).

KD for Object Detection

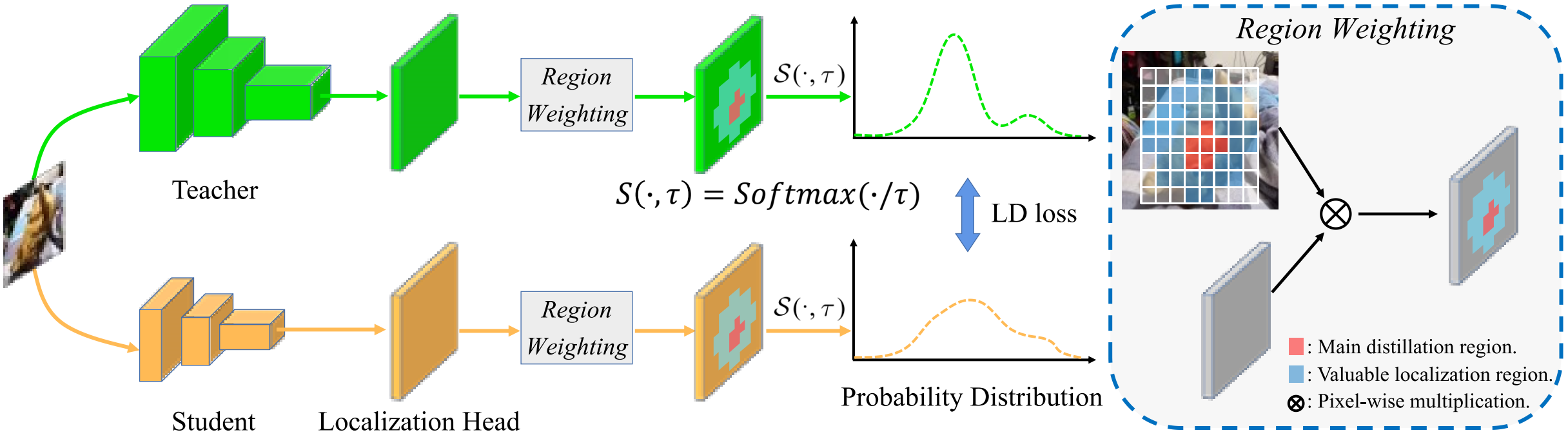
- Convert bounding box regression to classification problem
  - Divide the y-axis into 6 bins
  - Divide the x-axis into 6 bins



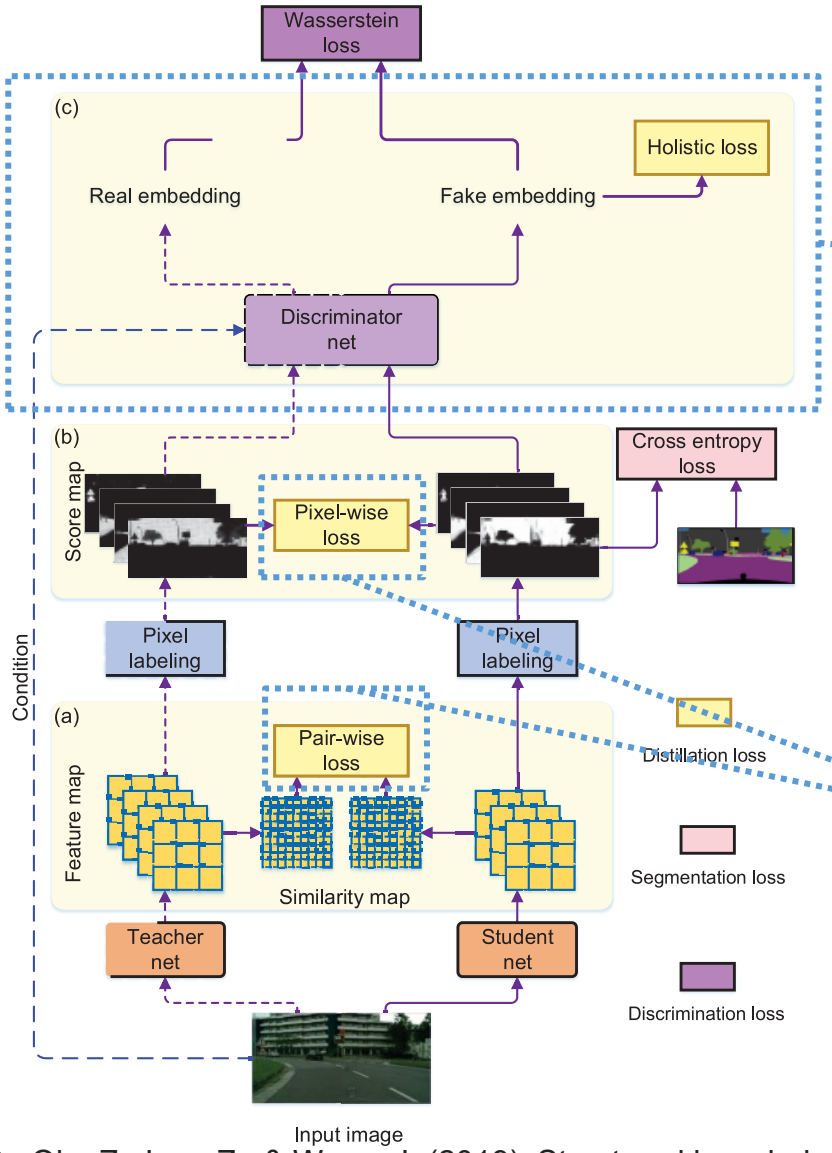
Zheng, Z., Ye, R., Wang, P., Ren, D., Zuo, W., Hou, Q., & Cheng, M. M. (2022). Localization Distillation for Dense Object Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 9407-9416).

Localization Distillation

- Calculate the distillation loss between two probability distributions predicted by the teacher and the student



KD for Semantic Segmentation

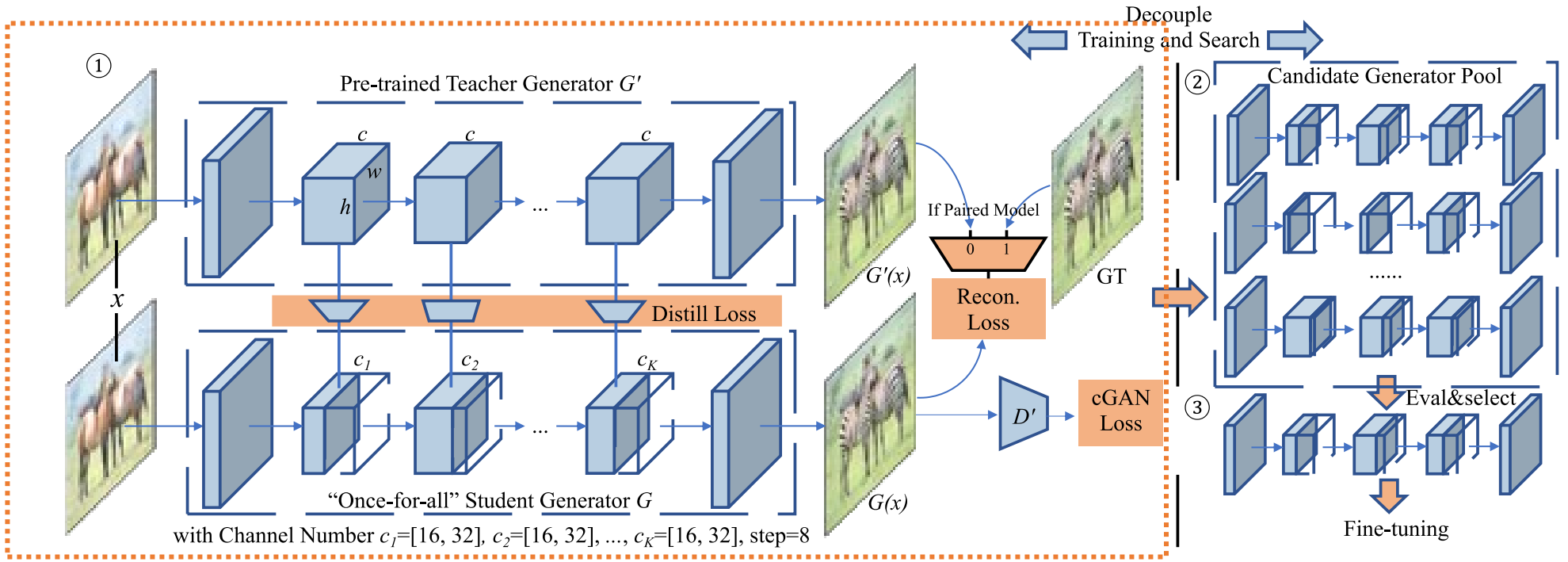


Add a discriminator network to provide adversarial loss: the student is trained to fool the discriminator network

Feature imitation is similar to classification and detection

Liu, Y., Chen, K., Liu, C., Qin, Z., Luo, Z., & Wang, J. (2019). Structured knowledge distillation for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 2604-2613).

KD for GAN



Reconstruction Loss

$$\mathcal{L}_{recon} = \begin{cases} \|G(x) - y\|, & paired\ cGAN \\ \|G(x) - G'(x)\|, & unpaired\ cGAN \end{cases}$$

Distillation Loss

$$\mathcal{L}_{distill} = \sum_{k=1}^{n} \|G_k(x) - f_k(G'(x))\|$$

cGAN Loss

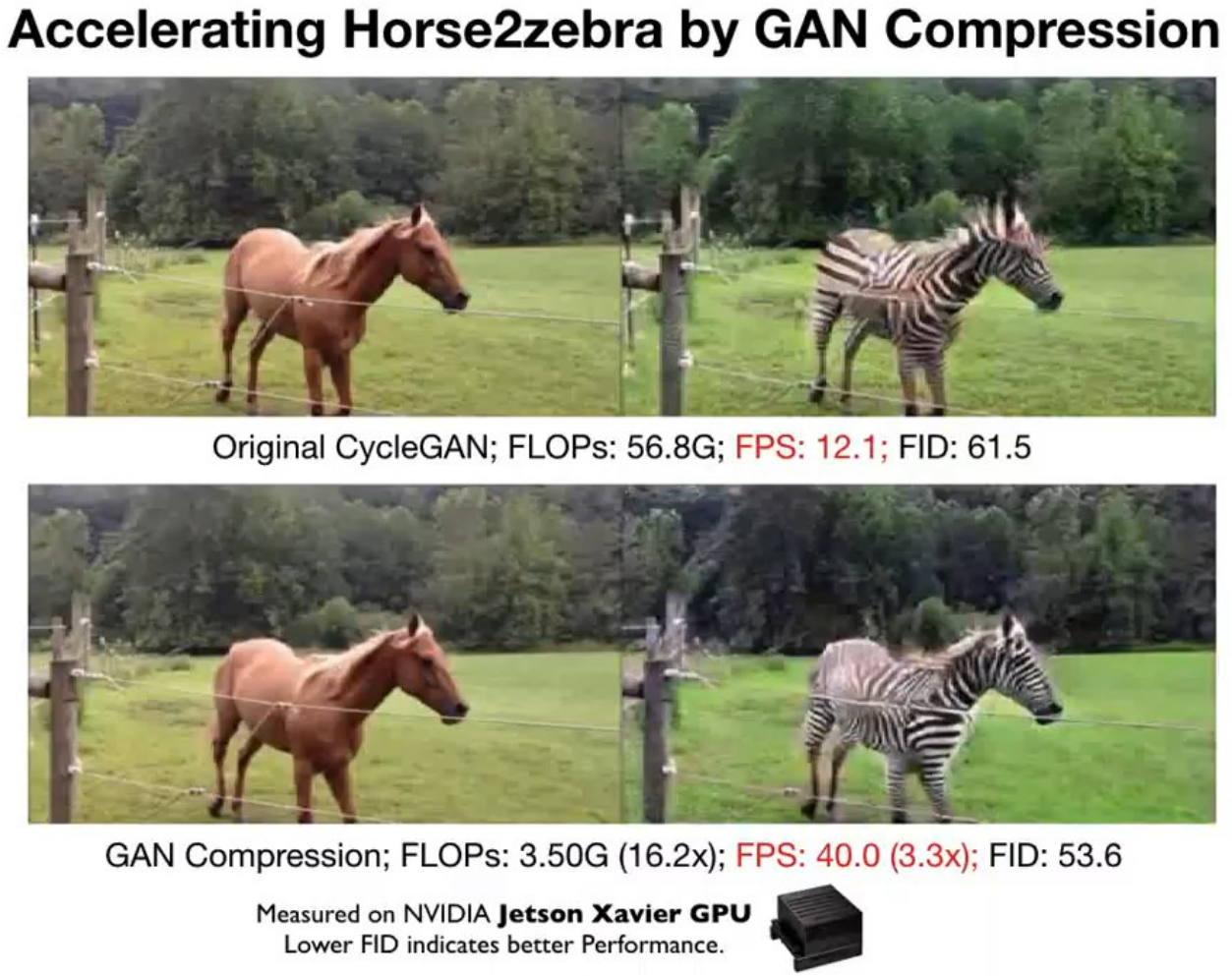
$$\mathcal{L}_{cGAN} = \mathbb{E}_{x,y}[\log D(x,y)] + \mathbb{E}\left[\log\left(1 - D\left(x, G(x)\right)\right)\right]$$

Training Objective

$$\mathcal{L} = \mathcal{L}_{cGAN}(x) + \lambda_{recon}\mathcal{L}_{recon}(x) + \lambda_{distill}\mathcal{L}_{distill}(x)$$

Li, M., Lin, J., Ding, Y., Liu, Z., Zhu, J. Y., & Han, S. (2020). Gan compression: Efficient architectures for interactive conditional gans. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 5284-5294).

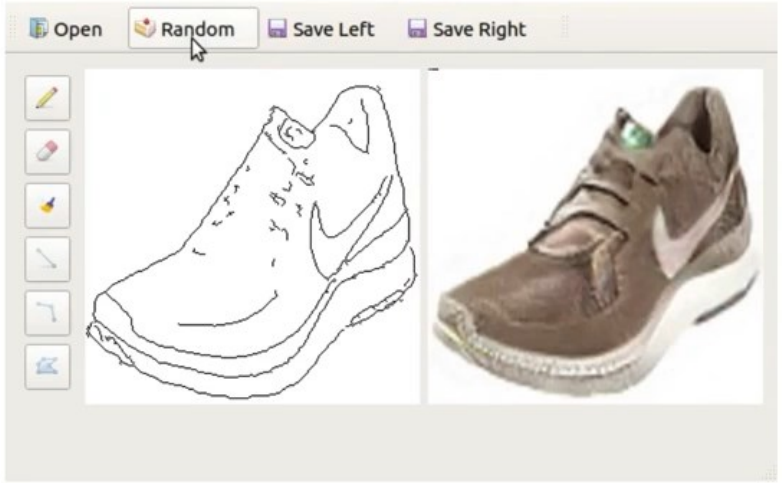
Demos on Horse2zebra Dataset



Interactive Image Editing Demo

Accelerating Edges2shoes by GAN Compression

Original Pix2pix
MACs: 56.8G FPS: 1.6 FID: 24.2

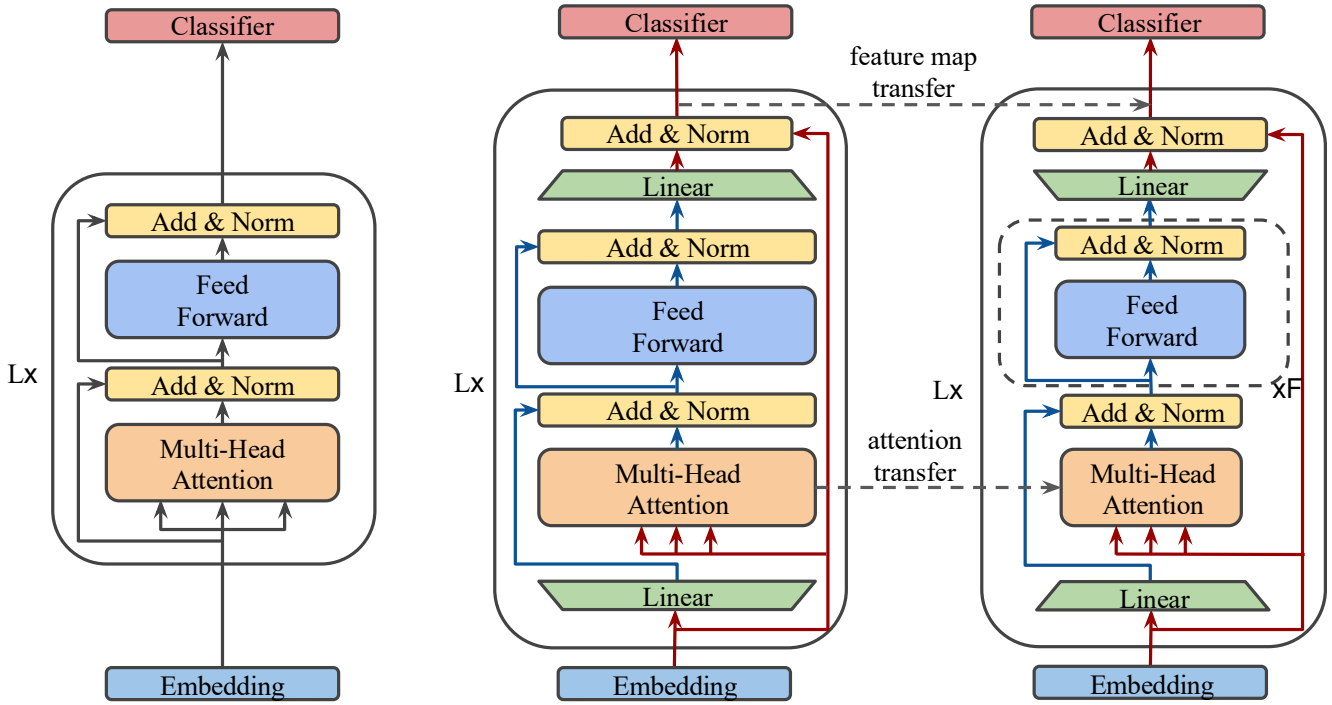


GAN Compression
MACs: 4.81G (11.8x) FPS: 3.9 (2.5x) FID: 26.6

Measured on NVIDIA **Jetson Nano GPU**
Lower FID indicates better Performance.

KD for NLP

- Attention Transfer
  - In addition to feature imitation, the student model is train to mimic teacher model's attention maps



Z. Sun et al. Mobilebert: a compact task-agnostic bert for resource limited devices. In *ACL*, pp. 2158–2170, 2020

Outline

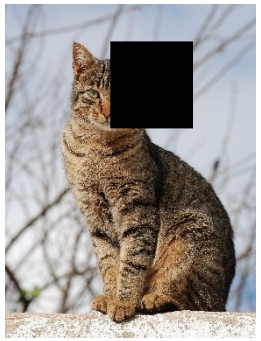
- What is Knowledge Distillation
- What to Match
- Self and Online Distillation
- Distillation for Different Tasks
- Network Augmentation, a Training Technique for Tiny Machine Learning Models

Conventional Approach

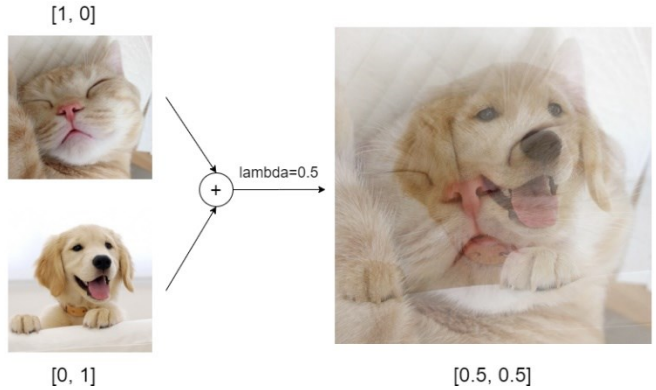
- Data augmentation during training to avoid overfitting

Data Augmentation

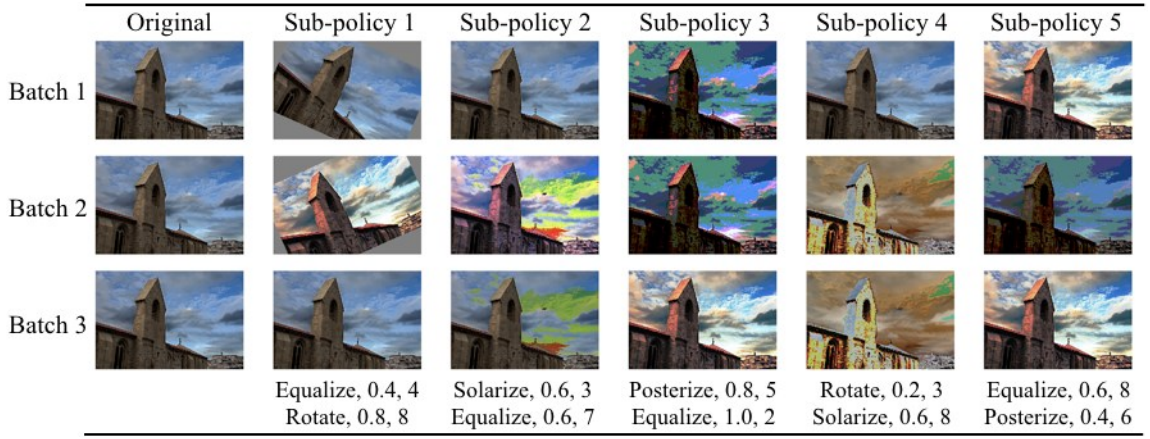
Cutout



Mixup



AutoAugment

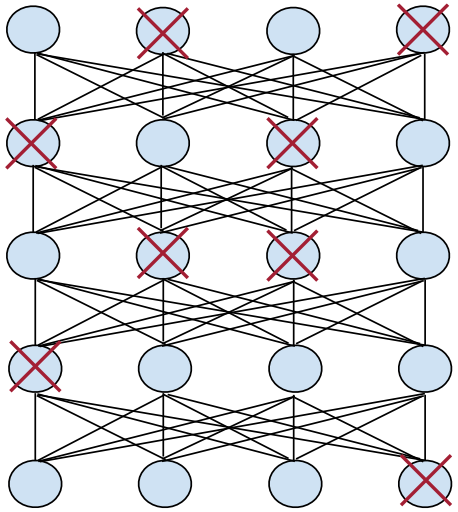


E. D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, and Q. V. Le, “AutoAugment: Learning augmentation policies from data,” Proc. Conf. Comput. Vis. Pattern Recognit., 2019, pp. 113–123.

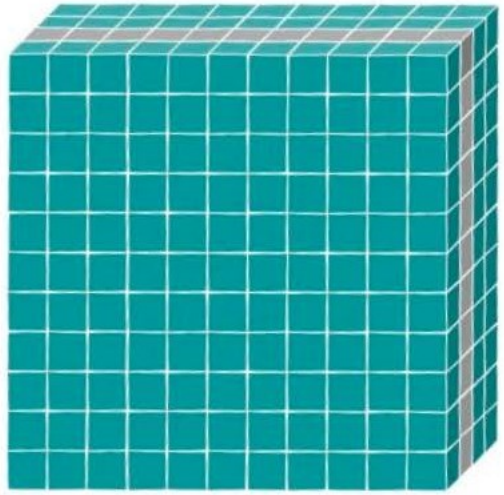
Conventional Approach

- Dropout during training to avoid overfitting

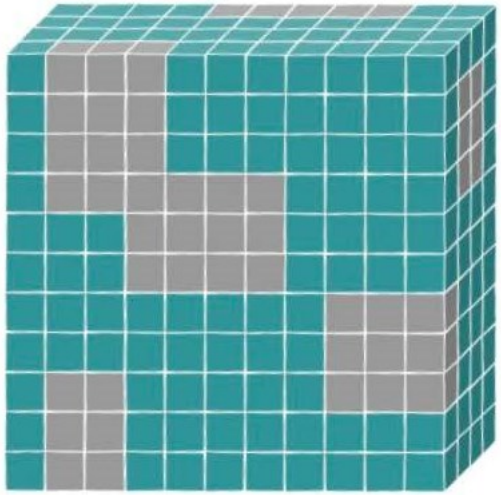
DropBlock



SpatialDropout



DropBlock

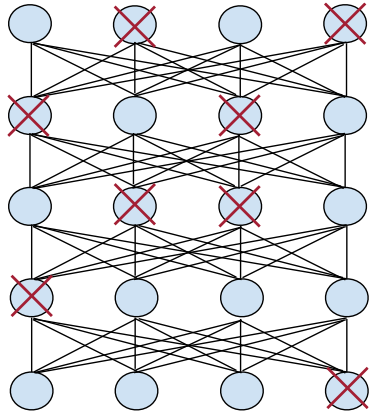


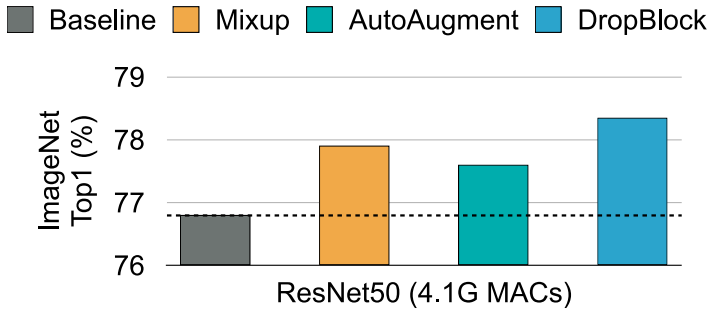
Dropout and Data augmentation on NN

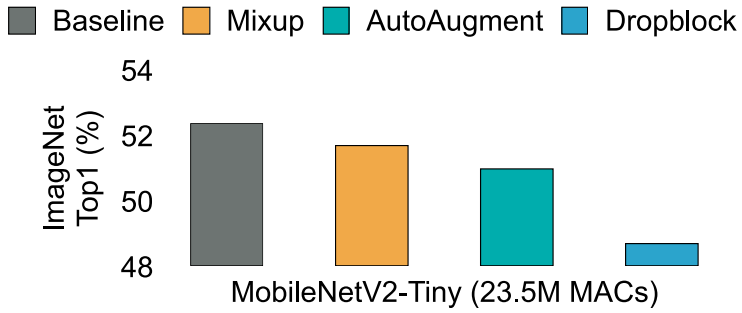
- Dropout/Data augmentation improves large neural networks' performance
  - **But hurts tiny neural networks' performance**

Data Augmentation

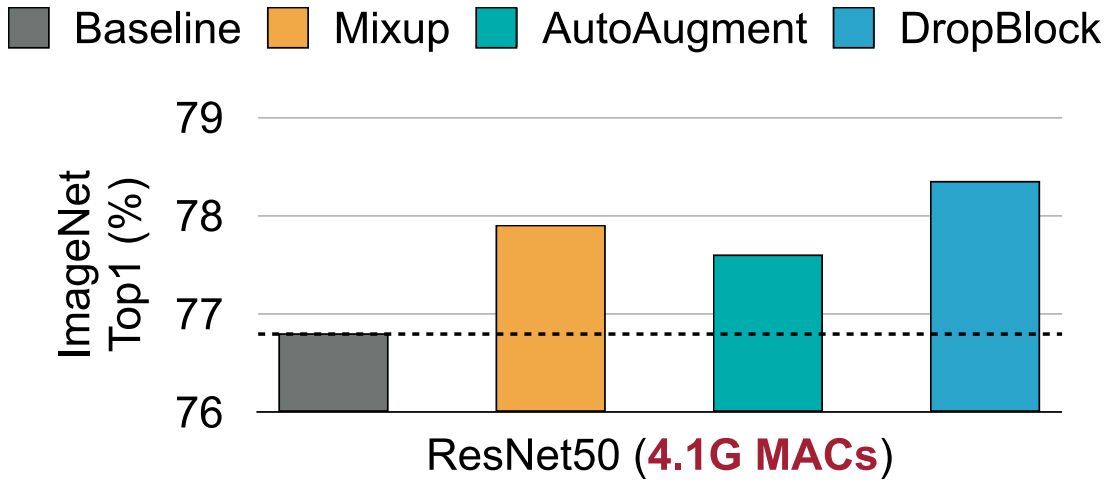
DropBlock

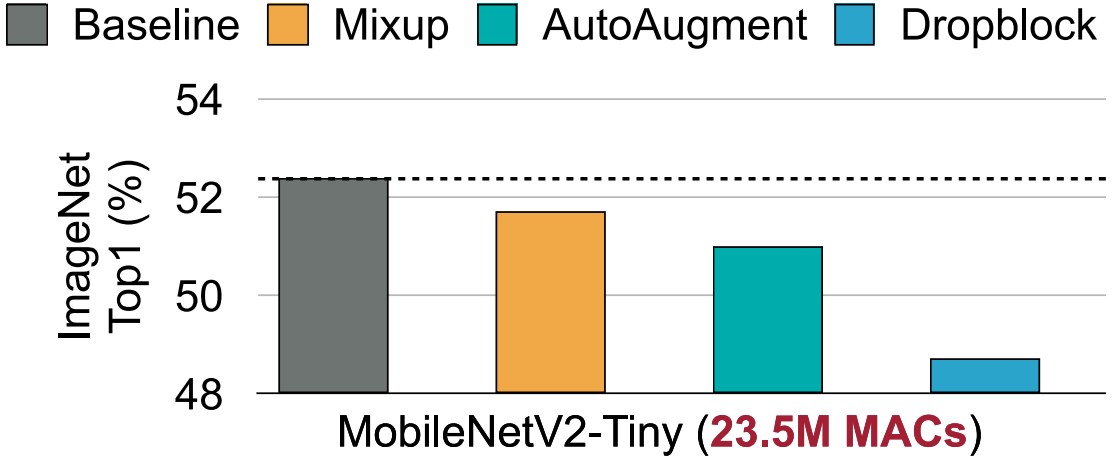




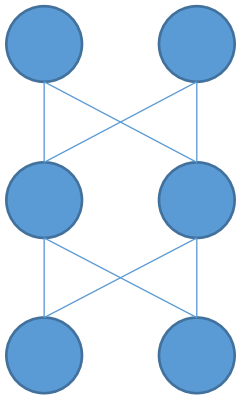


Tiny Neural Network Lacks Capacity

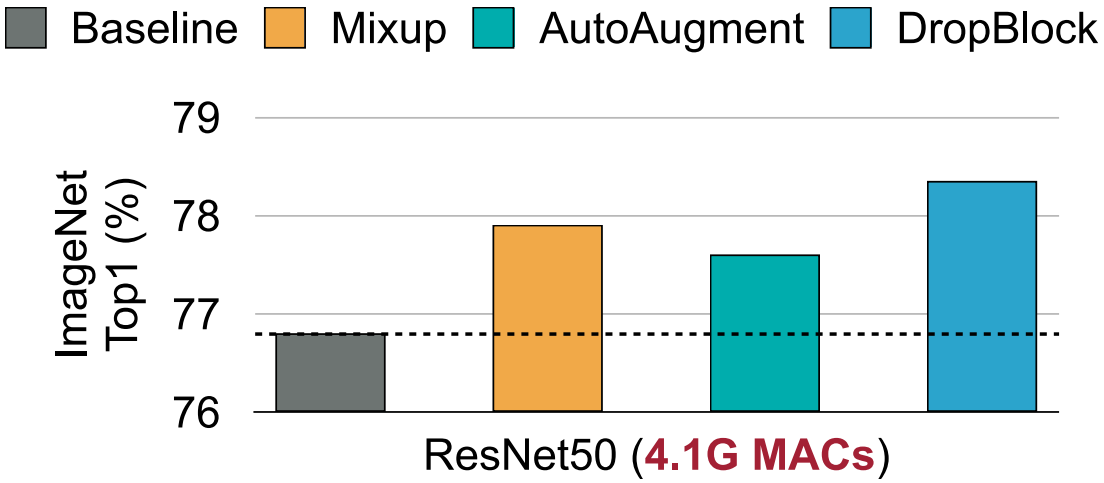


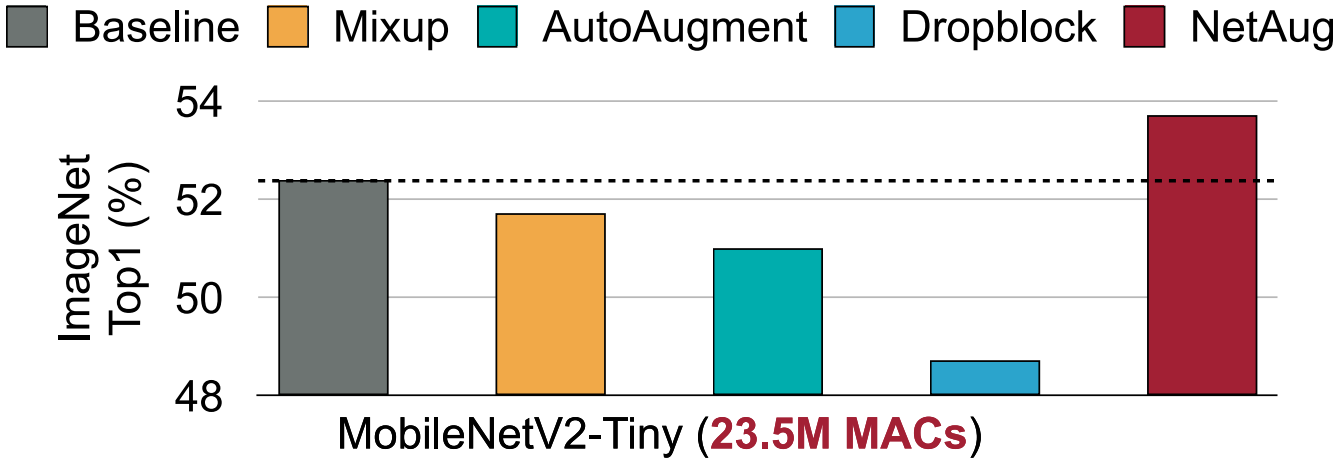


Tiny Neural Network



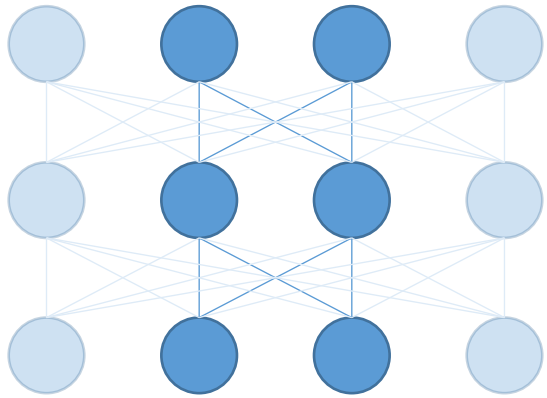
Network Augmentation





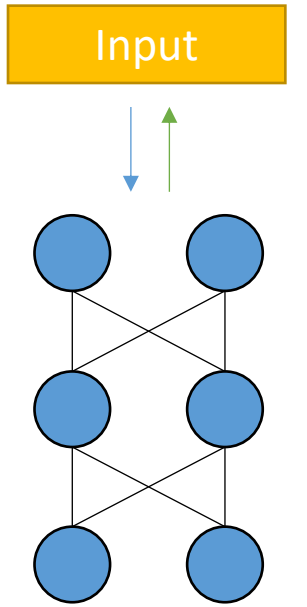
Augment the model to get extra supervision during training for tiny models.

Tiny Neural Network

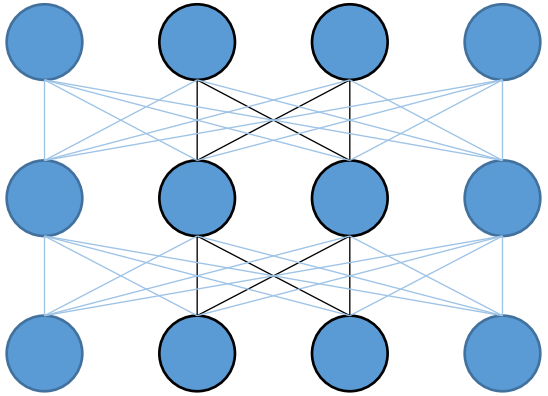


H. Cai, C. Gan, J. Lin, et al. Network Augmentation for Tiny Deep Learning. In ICLR, 2022

NetAug - Training Process



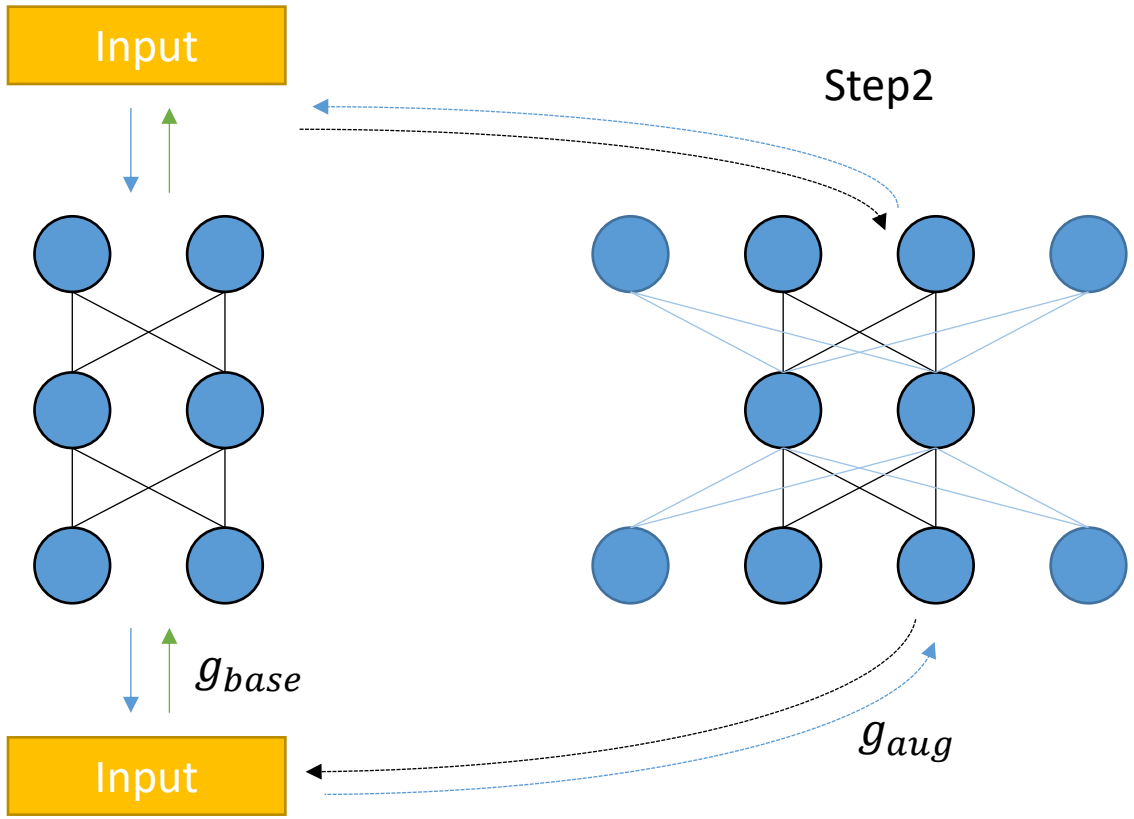
Build an Augmented Model



$$\mathcal{L}_{aug} = \mathcal{L}(W_{base})$$

Base supervision

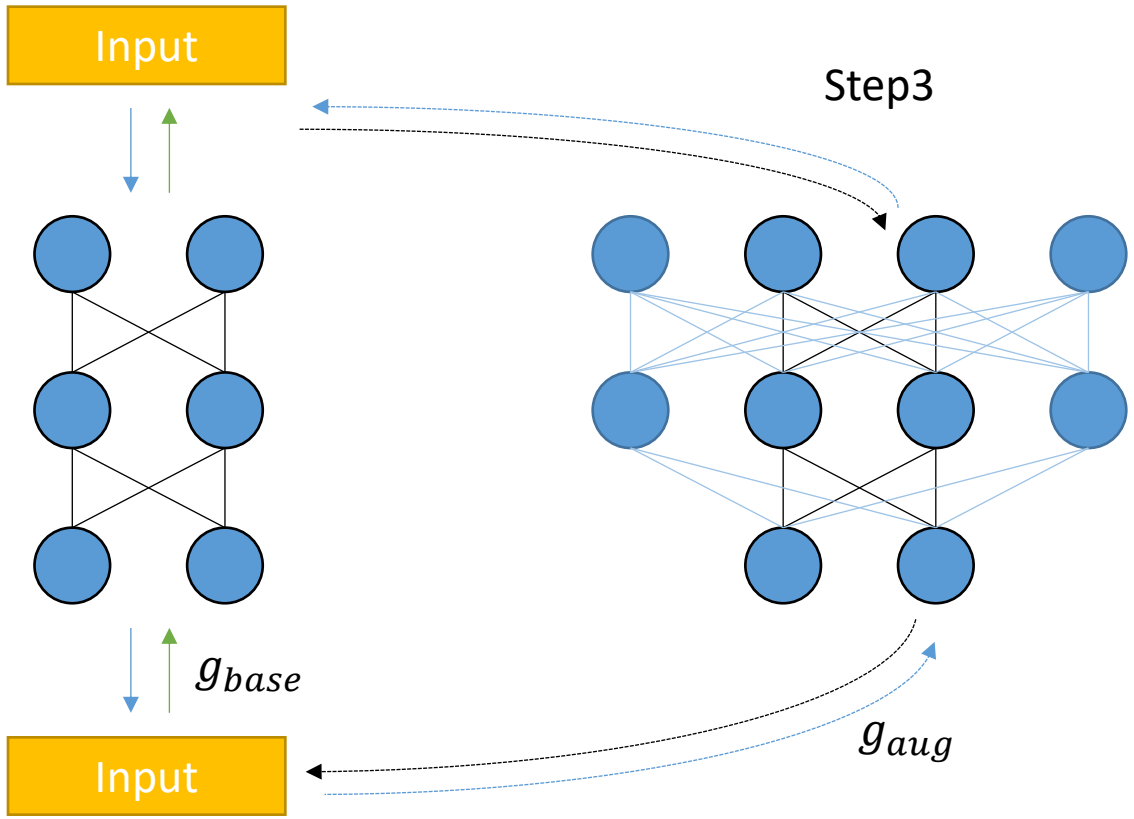
NetAug - Training Process



$$\mathcal{L}_{aug} = \mathcal{L}(W_{base}) + \alpha\mathcal{L}([W_{base}, W_{aug}])$$

Base supervision auxiliary supervision

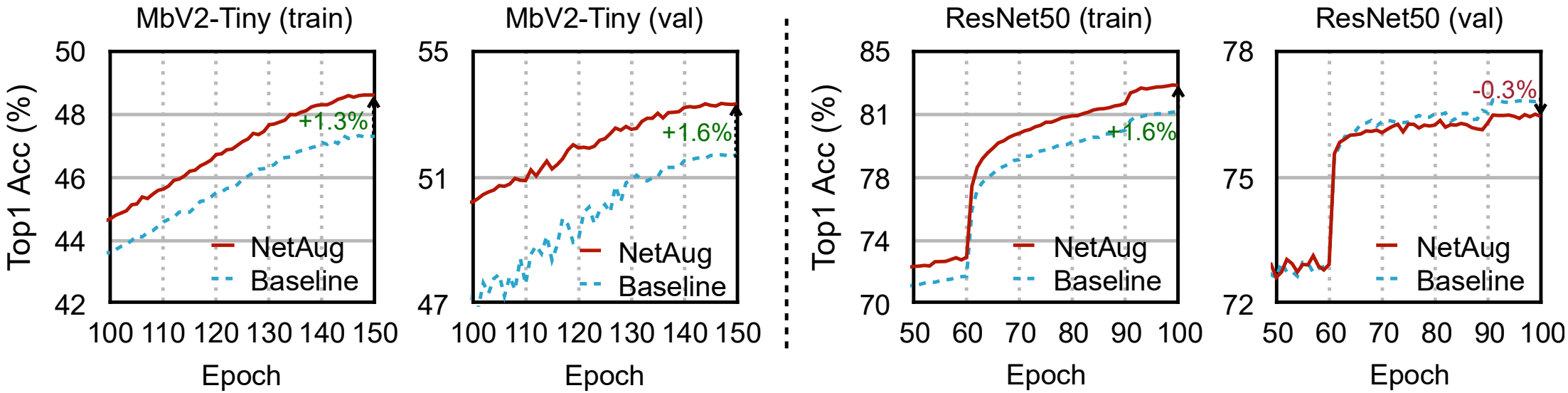
NetAug - Training Process



$$\mathcal{L}_{aug} = \mathcal{L}(W_{base}) + \alpha\mathcal{L}([W_{base}, W_{aug}])$$

Base supervision auxiliary supervision

NetAug - Learning Curve

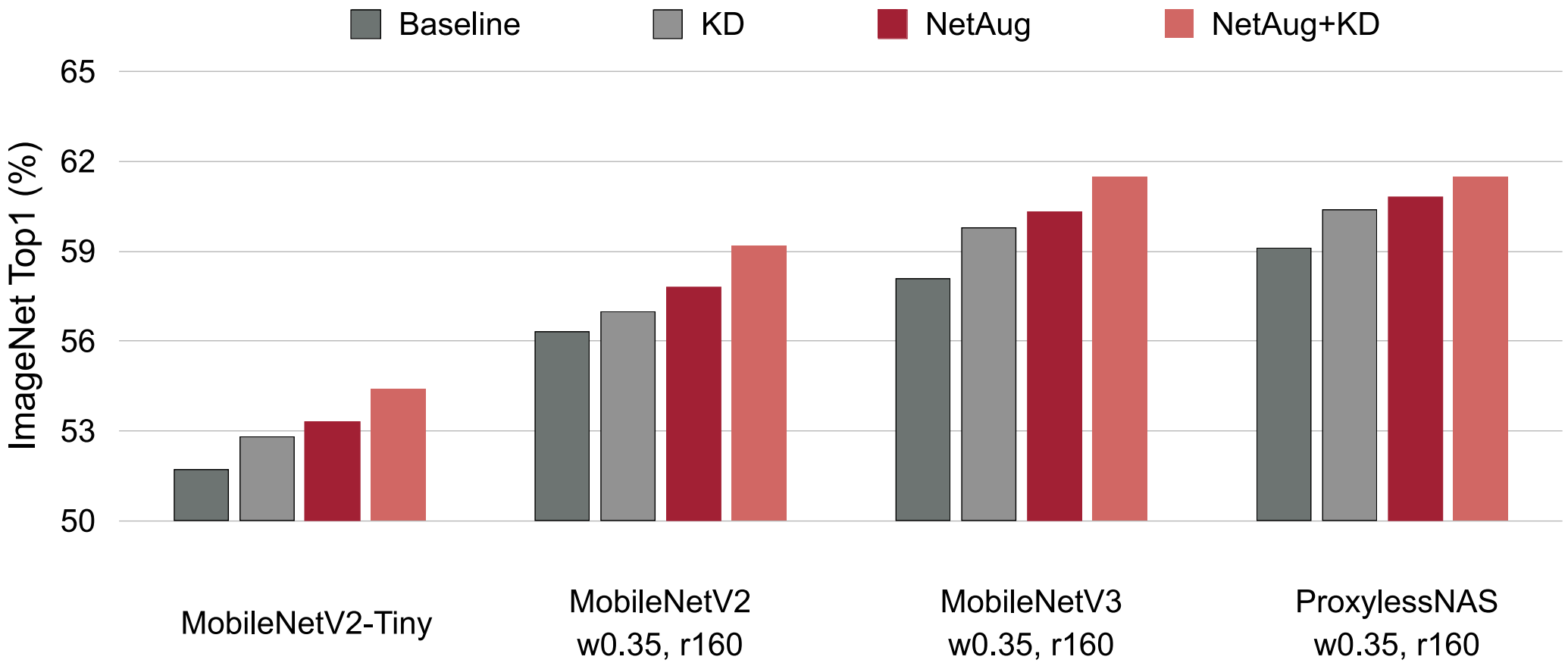


For a tiny neural network, NetAug improves both the training accuracy and val accuracy

For a large neural network, NetAug improves the training accuracy but hurts the val accuracy.

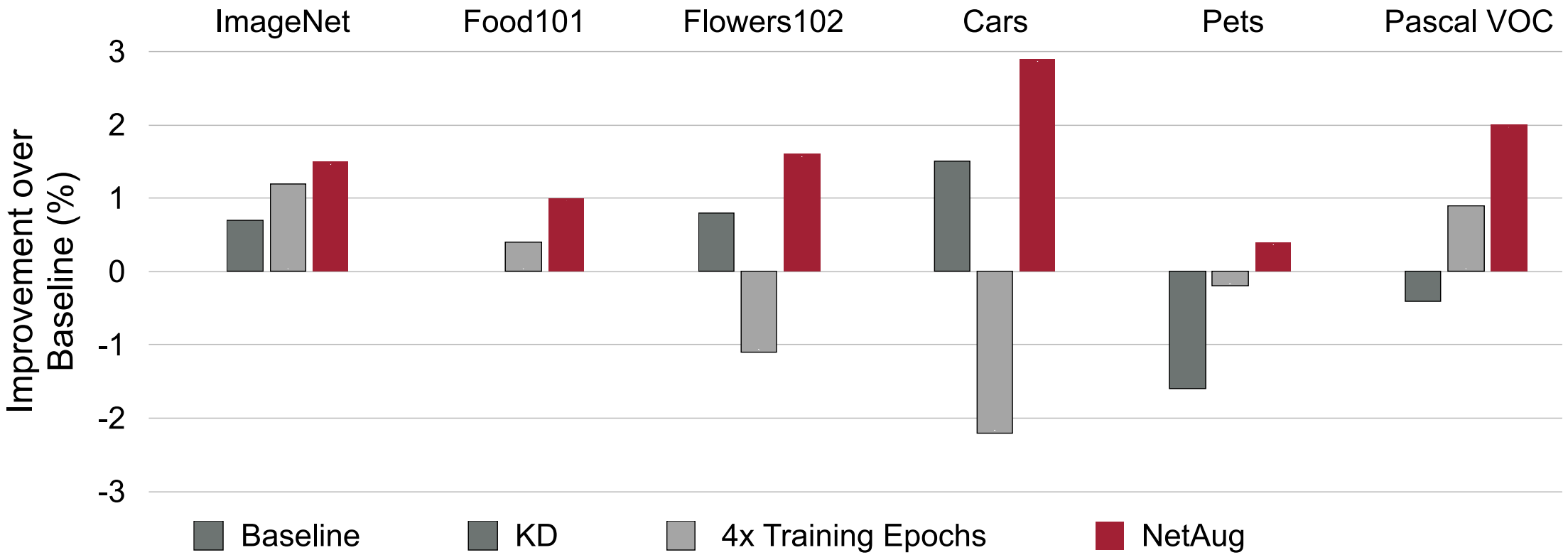
NetAug: Results

- NetAug is orthogonal to KD



NetAug - Transfer Learning

- NetAug provides better transfer learning performances than KD and 4x training schedule
  - Thought their ImageNet performances are similar.



NetAug - Transfer to Object Detection