Understanding the Role of Mixup in Knowledge Distillation: An Empirical Study (Supplementary Material)

Hongjun Choi, Eun Som Jeon, Ankita Shukla, Pavan Turaga

Geometric Media Lab
School of Arts, Media and Engineering, Arizona State University
School of Electrical, Computer and Energy Engineering, Arizona State University
hchoi71@asu.edu, ejeon6@asu.edu, ashukl20@asu.edu, pturaga@asu.edu

1. Additional Experimental Results

Here, we further evaluate the effectiveness of KD for various combinations of teacher-student networks. In Table 1, we can see several intriguing points; Firstly, as we claimed in observation 2), **distilling decent quality of knowledge** is crucial to success in KD. In other words, transferring knowledge from low-accuracy teacher models would hurt student performance. If we look at the last row in this table, the student network trained with the help of the lowest accuracy teacher (T:RN20) always shows degraded performance, compared to the baseline. Also, we can see the best accuracy (in **bold**) when the capacity of the teacher is larger than the student, but the best teacher models always do not lead to the best students. This result indicates that a better teacher sometimes may not be the best option for distillation. Secondly, we still anticipate **the benefit of smaller-capacity teachers' supervision** in training the larger-capacity student as seen in some cases such as T:RN32&S:RN44 and T:RN44&S:RN56, etc. This observation supports a recent finding [7] that the poorly-trained teacher with worse performance can also enhance the student. Consequently, we conjecture that the large capacity gap between a teacher and student would not result in favorable distillation performance, yet it is difficult to conclude the optimal capacity gap between them.

Without hard labels: In a conventional KD [3], the student network is trained by minimizing the similarity with two types of labels: the hard one-hot labels and the soft labels generated by the teacher network. Here, we minimize the cross-entropy loss with hard labels and minimize the KL divergence between output logits in the teacher and the student. In this way, KD gives more importance to the KL divergence loss to improve student performance. To figure out the effects of soft labels, in this experiment, we only train the student network where our students solely rely on the KL divergence loss without the hard labels. Table 2 presents test accuracy when using only soft labels with balancing parameter $\alpha_{kd} = 1$ and we obtain comparable performance as compared to Table 1. This result indicates that if knowledge is distilled from a well-trained teacher network, the student preserves its generalization using soft labels only. These results are in accord with the previous studies [1, 4, 6] that addressed the efficacy of soft labels.

Table 1. Test accuracy (%) on CIFAR100 for all teacher-student pairs when $\mathcal{T} = 4$ in T&S configuration.

	S:RN20	S:RN32	S:RN44	S:RN56	S:RN110
Baseline	68.90	71.43	72.29	72.41	74.31
KD, T:RN110	70.40	73.23	74.25	**74.98**	**76.08**
KD, T:RN56	**70.98**	**73.34**	**74.26**	74.82	75.28
KD, T:RN44	70.67	72.67	73.40	74.38	75.21
KD, T:RN32	70.57	72.73	73.54	74.00	74.56
KD, T:RN20	68.88	70.16	71.01	71.78	72.58

Table 2. Test accuracy (%) on CIFAR100 under all combinations of teacher/student when $\mathcal{T} = 4$ in T&S configuration. Note, we train the student model only with the soft labels.

	S:RN20	S:RN32	S:RN44	S:RN56	S:RN110
Baseline	68.90	71.43	72.29	72.41	74.31
KD, T:RN110	70.23	**73.11**	**74.40**	**74.89**	**76.01**
KD, T:RN56	70.27	72.65	73.68	74.75	75.53
KD, T:RN44	70.35	72.58	73.22	73.98	75.05
KD, T:RN32	**70.43**	71.95	73.08	73.68	74.70
KD, T:RN20	69.09	69.96	70.65	70.77	71.74

Vanilla models with mixup: In section 5, we compared our models with various distillation methods. Through our extensive results, we found that in some cases, partial mixup (PMU) results in better distillation performance and full mixup (FMU) sometimes outperforms partial mixup in

Table 3. Test accuracy (%) on CIFAR100 with vanilla models. Mix-S results are for the vanilla student models trained with mixup ($\alpha = 0.2$ and $\alpha = 1.0$). The higher accuracy among those two values is highlighted in red. The reported results (Mix-S) are averaged over 3 runs.

Teacher Student	W40-2 W16-2	W40-2 W40-1	RN56 RN20	RN110 RN20	RN110 RN32	RN32×4 RN8×4	VG13 VG8	VG13 MN2	RN50 MN2	RN50 VG8	RN32×4 SN1	RN32×4 SN2	W40-2 SN1
Teacher	75.61	75.61	72.34	74.31	74.31	79.42	74.64	74.64	79.34	79.34	79.42	79.42	75.61
Student	73.26	71.98	69.06	69.06	71.14	72.50	70.36	64.6	64.6	70.36	70.50	71.82	70.50
Mix-S ($\alpha = 0.2$)	**73.49**	**72.03**	**68.95**	**68.95**	**71.47**	**72.85**	70.99	65.82	65.82	70.99	73.90	74.79	73.90
Mix-S ($\alpha = 1.0$)	72.64	70.89	67.01	67.01	70.38	71.56	**71.25**	**66.14**	**66.14**	**71.25**	**74.48**	**75.64**	**74.48**
KD[3]	74.92	73.54	70.66	70.67	73.08	73.33	72.98	67.37	67.35	73.81	74.07	74.45	74.83
Ours (No mixup)	75.38	73.70	71.85	71.61	73.60	75.46	72.92	67.37	67.72	73.10	73.38	75.06	75.09
Ours (PMU=10%)	**76.06**	74.42	**72.09**	**71.94**	**74.07**	76.87	73.60	68.52	69.55	74.29	75.89	77.06	76.78
Ours (PMU=50%)	75.87	**74.69**	71.80	71.78	73.97	77.13	**74.00**	**69.14**	69.69	**74.61**	76.83	77.60	**77.18**
Ours (FMU)	75.69	73.34	70.98	70.99	73.48	**77.25**	73.84	68.81	**69.80**	74.50	**77.17**	**77.92**	77.00

some combinations. To further analyze these results, we train the vanilla student model from scratch with mixup according to the change in α and evaluate the performance. For comparison, we also present our results including the conventional KD method [3] as shown in Table 3. In this table, we explore two different α values, 0.2 and 1.0, and the higher accuracy among different α is highlighted in red. This result shows that there exists a network that is more favorable to mixup-augmentation, which also could result in better performance with strongly interpolated pairs ($\alpha = 1.0$). In addition, when the network that showed better performance in $\alpha = 0.2$ is used as the student in KD, a small amount of mixup pairs (PMU=10%) generally works well for our distillation method. Also, we observe that when the network that showed better performance in $\alpha = 1.0$ is used as the student in KD, either a large amount of mixup pairs or full mixup (PMU=50% or FMU) works well for ours, but they are not exactly linear related.

2. Additional Visualization Results

In this section, we provide additional feature representations for various networks from ResNet20 to ResNet110 [2] with the V-score [5].

Feature representation of the network trained with/without mixup: Figure 1 depicts the feature representations for various ResNet networks on the train and test sets. Similar to previous analyses, we selected (1) semantically similar classes (Baby, Boy, Girl, Man, and Woman) and (2) semantically different classes (Beaver, Apple, Aquarium Fish, Rocket, and Turtle). As we can see in this figure, the small-capacity network promotes more scattered projections than the large-capacity network. Specifically, if we look at the embeddings of ResNet20 in similar classes, they are notably dispersed on both train and test sets while it still well-preserves the feature separability in different classes.

Similar observations can be made in Figure 2. This figure illustrates the effect of the network trained with mixup (Mix-ResNet) in feature representation. As compared to the Figure 1, the projections of mixup-trained networks in similar classes are more scattered on both sets while the projections of the instances in different classes relatively form a tight and concentrated cluster. The supported measurement based on V-score is given in Figure 3.

V-scores: Now, we provide the V-score to support our findings as seen in Figure 3. The left figure illustrates the V-score on the train set for the network trained with/without mixup (i.e., Normal vs Mixup) and the right one depicts the V-score on the test set. In similar classes, interestingly, we observe that mixup-trained networks enforce V-scores drastically dropped in the train set. We note that in KD, the student and teacher models are trained on the same dataset. Therefore, this result supports our claim that a student trained with supervision by a mixup-trained teacher cannot take advantage of learning the superior knowledge because of the feature scattering in similar classes. We also can see a slight increase in V-score on the test set for different classes (the yellow bar is slightly above the gray bar in the right figure), and this might be because mixup improves generalization on the unseen data.

References

[1] Tommaso Furlanello, Zachary Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. Born again neural networks. In *Proceedings of the International Conference on Machine Learning*, pages 1607–1616, 2018.

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[3] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

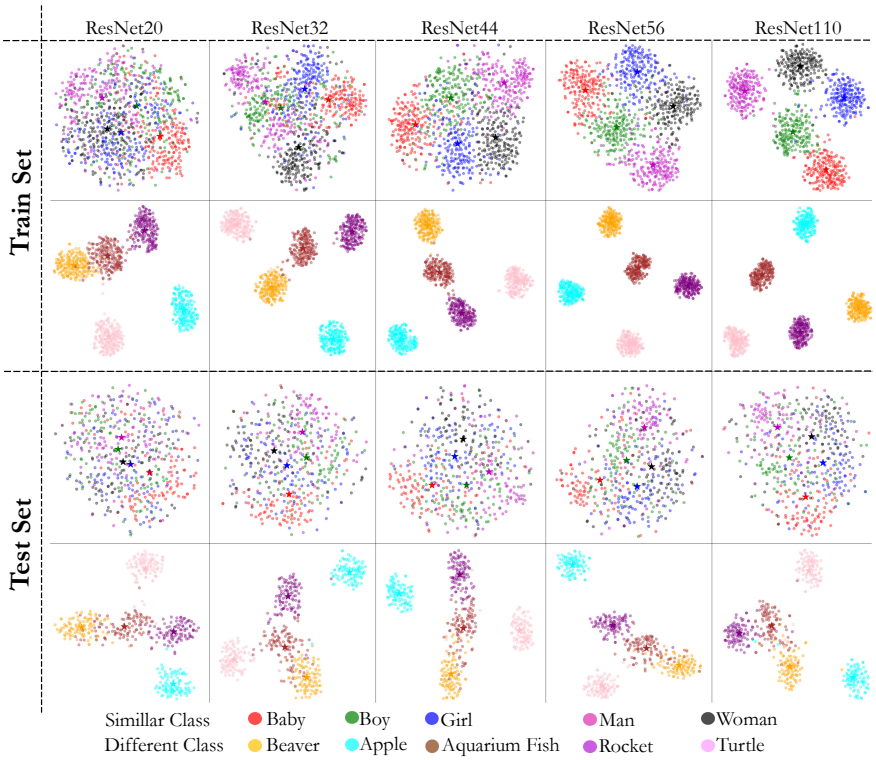

Figure 1. Feature representations of the penultimate layer with various ResNet networks trained **without** mixup.

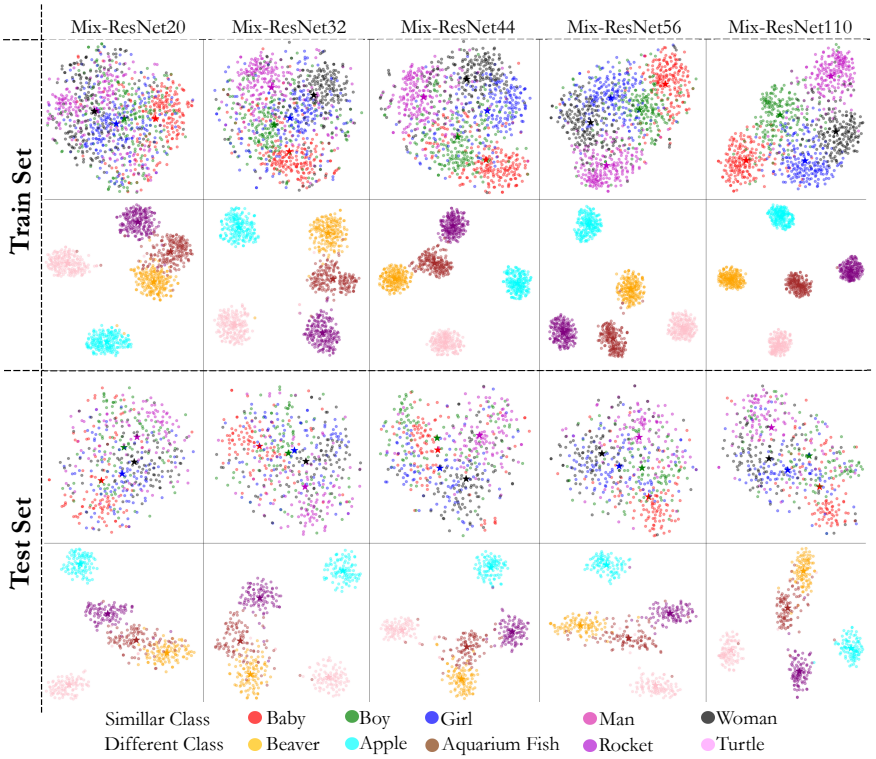

Figure 2. Feature representations of the penultimate layer with various ResNet networks trained **with** mixup.

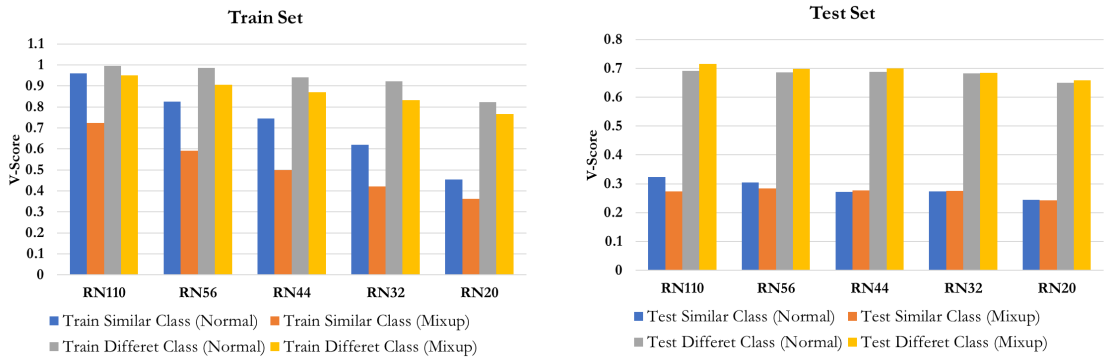

Figure 3. V-scores on train and test set for various ResNet models. The higher value is the better clustering.

[4] Taehyeon Kim, Jaehoon Oh, NakYil Kim, Sangwook Cho, and Se-Young Yun. Comparing kullback-leibler divergence and mean squared error loss in knowledge distillation. *arXiv preprint arXiv:2105.08919*, 2021.

[5] Andrew Rosenberg and Julia Hirschberg. V-measure: A conditional entropy-based external cluster evaluation measure. In *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 410–420, 2007.

[6] Jiaxi Tang, Rakesh Shivanna, Zhe Zhao, Dong Lin, Anima Singh, Ed H Chi, and Sagar Jain. Understanding and improving knowledge distillation. *arXiv preprint arXiv:2002.03532*, 2020.

[7] Li Yuan, Francis EH Tay, Guilin Li, Tao Wang, and Jiashi Feng. Revisiting knowledge distillation via label smoothing regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3903–3911, 2020.