ONLINE DISTILLATION WITH MIXED SAMPLE AUGMENTATION

Yiqing Shen\textsuperscript{1}, Liwu Xu\textsuperscript{2}, Yuzhe Yang\textsuperscript{2}, Yaqian Li\textsuperscript{*2}, Yandong Guo\textsuperscript{2}

\textsuperscript{1} Shanghai Jiao Tong University \textsuperscript{2} OPPO Research Institute
shenyq@sjtu.edu.cn, \{xuliwu,yangyuzhe,liyaqian\}@oppo.com, yandong.guo@live.com

ABSTRACT
Mixed Sample Regularization (MSR), such as MixUp or CutMix, is a powerful data augmentation strategy to generalize convolutional neural networks. Previous empirical analysis has illustrated an orthogonal performance gain between MSR and the conventional offline Knowledge Distillation (KD). To be more specific, student networks can be enhanced with the involvement of MSR in the training stage of the sequential distillation. Yet, the interplay between MSR and online knowledge distillation, a stronger distillation paradigm, where an ensemble of peer students learn mutually from each other, remains unexplored. To bridge the gap, we make the first attempt at incorporating CutMix into online distillation, where we empirically observe a significant improvement. Encouraged by this fact, we propose an even stronger MSR specifically for online distillation, named as Cut\textsuperscript{a}Mix. Furthermore, a novel online distillation framework is designed upon Cut\textsuperscript{a}Mix, to enhance the distillation with feature level mutual learning and a self-ensemble teacher. Comprehensive evaluations on CIFAR10 and CIFAR100 with six network architectures show that our approach can consistently outperform state-of-the-art distillation methods.

Index Terms— Online knowledge distillation, data augmentation, CutMix, knowledge ensemble distillation.

1. INTRODUCTION
Knowledge Distillation (KD) \cite{hinton2015distilling} has demonstrated a bright promise to generalize deep neural networks in various tasks \cite{wang2021empirical} \cite{jang2021ensemble}. In vanilla offline distillation, the student model gains extra knowledge by minimizing a discrepancy between its predictions and the soft labels generated by a stronger teacher model, i.e.

\[
\mathcal{L}_{KD} = \frac{1}{\tau} \cdot \mathcal{D}_{KL}(\sigma\left(\frac{z}{\tau}\right), \sigma\left(z^t / \tau\right)).
\]  

The temperature $\tau$ is a hyper-parameter, which determines the strength to smooth the predicted probabilities. $\sigma\left(\cdot\right)$ is the softmax operation, and $\mathcal{D}_{KL}(\cdot, \cdot)$ writes for the Kullback-Leibler divergence. $z^s$, $z^t$ are the logits from student and teacher respectively. In addition to the two-stage distillation scheme, online distillation is another KD paradigm, for example, deep mutual learning (DML) \cite{lee2017training}. To improve the training efficiency, it trains an ensemble of students by learning mutually from each other, without the involvement of a complex pre-trained teacher. Afterward, a special case of online distillation, namely self distillation, is proposed to use identical structures for both the teacher and the student models to further enhance the simplicity of distillation approaches \cite{park2020self} \cite{park2021self}.

Strong data augmentation schemes, particularly the Mixed Sample Augmentation (MSA) e.g. CutOut \cite{devries2017improved}, MixUp \cite{zhang2017mixup} and CutMix \cite{yun2019cutmix}, show their effectiveness to generalize CNNs in various tasks including classification, segmentation, detection. Thus, straightforwardly, previous works have studied the interplay between offline KD and data augmentation strategies. Yet, the performance of the student network is impaired if its teacher is trained with MSA. In contrast, the training of the student itself is complementary to MSA \cite{zhao2019self} \cite{zhao2020fusing}. This finding also echoes with the empirical success of the compatibility of MSA and many self distillation approaches \cite{park2021self} \cite{park2020self}. Conclusively, the performance improvement achieved by KD does not conflict with MSA.

Yet, to the best of our knowledge, the compatibility between MSA and online distillation \cite{zhang2019online} has not been fully explored. Compared with vanilla two-stage distillation, online distillation can reduce the training expense and computational cost to derive a complex pre-trained model is required \cite{wu2020online} \cite{zhang2021online} \cite{wu2021online}, and also demonstrate a competitive performance to offline distillation. Consequently, exploring the interplay between online distillation and MSR is on the demand. In this work, we primarily focus on CutMix as a case study, which has demonstrated a superiority among other MSRs when combined with the conventional two-stage KDs \cite{park2020self}.

In a nutshell, the major contributions are four-fold. 1) To explore the interplay between MSR and online distillation, we make the first attempt to empirically uncover an interesting phenomenon that MSR can consistently improve deep mutual learning \cite{zhang2019online}. 2) Inspired by this fact, we further propose a stronger MSR strategy specific for online distillation, named as Cut\textsuperscript{a}Mix. 3) Moreover, we build a novel online distillation framework upon Cut\textsuperscript{a}Mix to achieve more significant performance gain. 4) Finally, we conduct comprehensive experi-
ments on CIFAR-10 and CIFAR-100 to show that our method can outperform state-of-the-art KD methods.

2. METHODS

2.1. Preliminaries

We focus on the classification task as a case study, given \( N \) annotated samples \( \{(x_i, y_i)\}_{i=1}^{N} \) from \( K \) classes. In online distillation [4], multiple peer students learn from each other. Specifically, when given \( J \) networks, then the objective function of \( j \)-th network is

\[
\mathcal{L}_j = \mathcal{L}_{CE}^j + \alpha \cdot \tau^2 \cdot \mathcal{L}_{DML}^j
\]

with \( \mathcal{L}_{DML}^j = \frac{1}{J-1} \sum_{k \neq j} \mathcal{D}_{DML}(\sigma(z_i^j / \tau), \sigma(z_k^j / \tau)) \),

where \( \mathcal{L}_{CE}^j \) is the cross-entropy loss, \( z_i^j \) is the logits from \( j \)-th network, and the balancing coefficient \( \alpha \) is set to 1 in [4]. Subsequently, we plug the CutMix into DML by assigning identically the same batch of images after mixed operation to each networks. To be more specific, data samples are augmented progressively by

\[
\tilde{x} = M \odot x_A + (1 - M) \odot x_B,
\]

where \( \odot \) denotes the pixel-wise multiplication and mask \( M \in \{0, 1\}^W \times H \). Meanwhile, the associated one-hot labels \( y_A, y_B \) are mixed in the same fashion i.e.

\[
\tilde{y} = \lambda y_A + (1 - \lambda) y_B
\]

with a random balancing coefficient \( \lambda \) sampled from beta distribution Beta(1, 1) [9]. We derive a uniform distorted samples from the peer students with the mixed sample and labels. The combination between CutMix and DML results in a marginal performance improvement, as reported in Table [4] which suggest the compatibility between CutMix and online distillation. Inspired by this fact, we aim to design a more powerful online distillation with mixed sample augmentation, as depicted in Fig. [1]

2.2. Cut\textsuperscript{a}Mix

We adopt data distortion with different random seeds for raw images separately to each network, which targets to enhance the invariance against perturbations in the data domain [17]. We denote the batch of augmented images associated with \( j \)-th network as \( B_j = \{x_i^j, y_i\}_{i=1}^{n} \), where \( n \) writes for the batch size, and the superscript \( j \) is omitted for \( y_i \) as all versions of distorted batches share the identical ground truth labels. Subsequently, in our Cut\textsuperscript{a}Mix, the mixed training sample and the label are generated as follows:

\[
\tilde{x}_i^j = M \odot x_i^j + (1 - M) \odot x_i^j,
\]

\[
\tilde{y}_i^j = \lambda i \cdot y_i + (1 - \lambda i) \cdot y_i.
\]

Here, we employ multiple independent binary masks for each network, which is generated from the same beta distribution, to increase the degree of data distortion. Thus, the peer networks can benefit from a strong regularization effect.

Each binary masks \( M \) indicating where to drop out and fill in from two images follows the previous sampling strategy in [9]. Notably, we impose a addition constraint on the combination ratio as follows

\[
\lambda_i = \lambda \text{ (for } j = 1, 2, \cdots, J) \text{,}
\]

where \( \lambda \) is sampled from beta distribution Beta(1, 1). Eq. (7) guarantees a valid online distillation process, since each network is trained with identical series of groundtruth.

2.3. Online Distillation with Cut\textsuperscript{a}Mix

To facilitate the online distillation on more variant data samples augmented by Cut\textsuperscript{a}Mix, we adopt a Maximum Mean Discrepancy (MMD) loss [18] to assist the feature level distillation and employ a classifier \( g \) as a peer teacher by assembling the feature representations from penultimate layers [15]. Subsequently, the overall training optimisation objective for \( j \)-th network is made up of four components: 1) standard cross-entropy loss \( \mathcal{L}_{CE}^j \), 2) logit-level mutual distillation loss \( \mathcal{L}_{DML}^j \) in Eq. (2) to learn from soft labels, 3) MMD loss \( \mathcal{L}_{MMD}^j \) for peer feature-level distillation, and 4) peer teacher distillation loss \( \mathcal{L}_{PT}^j \) to transfer knowledge from the capable assembled teacher. Thus, the overall loss function is

\[
\mathcal{L}_j = \mathcal{L}_{CE}^j + \alpha \cdot \mathcal{L}_{DML}^j + \beta \cdot \mathcal{L}_{MMD}^j + \gamma \cdot \mathcal{L}_{PT}^j,
\]

with three balancing coefficient \( \alpha, \beta, \gamma \), for \( j = 1, 2, \cdots, J \).

We write the feature representations generated from \( x_i^j \) by the \( j \)-th network as \( f_i^j \). Then the empirical MMD loss, targeting at feature level mutual distillation, is formulated as:

\[
\mathcal{L}_{MMD}^j = \frac{1}{J-1} \sum_{k \neq j} \frac{1}{n} \sum_{i=1}^{n} \left\| f_i^j - f_i^k \right\|_2.
\]

where \( \| \cdot \|_2 \) is the L-2 norm. Peer teacher is a classifier, using the feature representations from peer networks i.e. \( z_i^{PT} = g(f_i^j, \cdots, f_i^j) \), which is trained by a standard cross-entropy loss. Then, we transfer the knowledge from the ensemble teacher to peer student by:

\[
\mathcal{L}_{PT}^j = \tau^2 \frac{1}{n} \sum_{i=1}^{n} \mathcal{D}_{KL}(\sigma(z_i^j / \tau), \sigma(z_i^{PT} / \tau)).
\]

We summarize the overall training process in Algorithm [1]

3. EXPERIMENTS

3.1. Datasets and Implementations

We evaluate the proposed method on two commonly used image classification datasets, namely CIFAR-10 and CIFAR-
we use SGD with Nesterov momentum for optimization and set the momentum, weight decay rate, and the initial learning rate to 0.9, 5 × 10^{-4} and 0.05 respectively. Each network is trained with 240 epochs, where the learning rate is scheduled to decay by 10% at 150th, 180th and 210th epoch. In the proposed framework, we set the number of peer students J = 2, temperature τ = 3 for distillation, α = 0.6, β = 0.2 and γ = 0.1 in Eq.(8).

### 3.2. Compared Methods

We compare our proposed method with the existing state-of-the-art self knowledge distillation including Class-wise Knowledge Distillation (CS-KD) [5], Data-Distortion Guided Self-Distillation (DDGSD) [13] and online knowledge distillation methods including Deep Mutual Learning (DML) [4], Knowledge Distillation via Collaborative Learning (KDCL) [17], On-the-fly Native Ensemble (ONE) [14] and Online

---

**Table 1.** Top-1 classification accuracy(%) comparison on CIFAR-10 and CIFAR-100 with the state-of-the-arts. The best and second best performance are highlighted in Red and Green respectively.

| Dataset | Methods | Vgg-16 | Vgg-19 | ResNet-32 | ResNet-110 | WideResNet | DenseNet |
|---------|---------|--------|--------|-----------|------------|------------|----------|
| CIFAR-10 | Baseline | 93.89±0.06 | 93.95±0.06 | 93.51±0.05 | 94.83±0.02 | 94.47±0.10 | 92.89±0.28 |
|         | DML [4]  | 94.13±0.07 | 93.93±0.18 | 93.91±0.11 | 94.46±0.24 | 95.34±0.06 | 93.26±0.13 |
|         | CutMix [9] | 94.71±0.03 | 94.59±0.02 | 94.04±0.20 | 95.18±0.07 | 95.11±0.12 | 93.24±0.12 |
|         | CS-KD [5] | 93.72±0.20 | 93.54±0.04 | 93.21±0.08 | 93.90±0.05 | 95.03±0.26 | 92.13±0.27 |
|         | DDGSD [13] | 94.35±0.14 | 94.27±0.14 | 94.04±0.06 | 95.24±0.16 | 95.25±0.12 | 93.18±0.21 |
|         | KDCL [17] | 93.67±0.08 | 93.49±0.02 | 93.71±0.02 | 94.91±0.17 | 95.10±0.10 | 92.32±0.16 |
|         | ONE [14] | 93.70±0.29 | 93.73±0.13 | 94.03±0.08 | 95.24±0.16 | 95.44±0.00 | 92.90±0.15 |
|         | OKDDip [16] | 93.57±0.16 | 93.33±0.25 | 93.55±0.00 | 95.21±0.25 | 94.47±0.10 | 92.28±0.06 |
| Ours    | 94.84±0.25 | 94.81±0.08 | 94.24±0.06 | 95.76±0.14 | 95.90±0.20 | 93.64±0.12 |

| CIFAR-100 | Baseline | 73.72±0.15 | 72.83±0.19 | 71.71±0.24 | 76.28±0.26 | 77.71±0.08 | 71.62±0.47 |
|           | DML [4]  | 75.39±0.41 | 74.14±0.24 | 72.65±0.40 | 77.90±0.01 | 79.57±0.00 | 72.36±0.07 |
|           | CutMix [9] | 75.43±0.32 | 74.34±0.33 | 72.69±0.30 | 78.58±0.08 | 79.43±0.16 | 71.74±0.36 |
|           | CS-KD [5] | 74.53±0.30 | 73.59±0.20 | 70.85±0.31 | 76.79±0.24 | 78.39±0.00 | 70.50±0.00 |
|           | DDGSD [13] | 75.65±0.64 | 75.32±0.61 | 73.68±0.23 | 77.45±0.17 | 79.17±0.08 | 72.32±0.32 |
|           | KDCL [17] | 73.04±0.44 | 72.19±0.43 | 71.67±0.04 | 78.37±0.22 | 79.49±0.01 | 72.16±0.31 |
|           | ONE [14] | 73.33±0.43 | 72.09±0.82 | 73.71±0.43 | 78.91±0.18 | 78.62±0.06 | 72.14±0.02 |
|           | OKDDip [16] | 73.97±0.67 | 71.71±0.14 | 72.34±0.08 | 78.17±0.06 | 79.34±0.40 | 70.62±0.30 |
| Ours     | 76.56±0.24 | 75.68±0.30 | 73.84±0.25 | 79.32±0.15 | 80.09±0.07 | 72.64±0.12 |

---
Algorithm 1 Proposed Online Distillation with Cut^Mix

Input: Training set \( \{(x_i, y_i)\}_{i=1}^{n} \)

1: for \( i = 1, \ldots, \text{Max\_Epoch} \) do
2: Random sample a batch of data \( \{(x_i, y_i)\}_{i=1}^{n_i} \)
3: Derive the mixed data samples \( \{(x^{j}_{ij}, y_i)\}_{i=1}^{n_i} \)
4: Mix the data and label by Eq. (5), (6)
5: Compute features and logits
6: Assemble features by the peer teacher
7: Compute cross entropy loss \( L_{CE} \)
8: Compute distillation loss \( L_{DML} \) in Eq. (2)
9: Compute MMD loss \( L_{MMD} \) in Eq. (9)
10: Compute peer teacher distillation loss \( L'_{PT} \) in Eq. (10), and peer teacher supervision loss
11: Update peer students by Eq. (8)
12: Update peer teacher
13: end for

Knowledge Distillation with Diverse Peers (OKDDip) \[16\]. For a fair comparison, we adopt two peer students in DML and set the number of branches to 2 in KDCL, ONE, and OKDDip. We follow the original settings of the remaining extra hyper-parameters. In the ‘baseline’, each CNN is trained directly from the hard labels. We compute the average and standard deviation of top-1 classification accuracy (%) over five runs.

### 3.3. Experimental Results

First, we provide an illustrative example of the training dynamics in Fig. 2 In Table 1 our method consistently improves the performance of various backbone networks (baseline), achieving a 0.73-1.48% performance gain on CIFAR-10 and 1.02-3.04% on CIFAR-100 respectively. The fact proves the compatibility between Cut^Mix and online distillation. And it shows its effectiveness to enhance the generalization ability of various networks. Notably, our method can also outperform the state-of-the-art distillation schemes, including self-distillation and online distillation approaches, without extra computational cost. Our method also achieves the highest performance in terms of model ensembling, as demonstrated in Table 2. We also make the following observations: more complex backbones, e.g. WideResNet, obtain a higher performance boost from our methods than the lightweight ones.

### 3.4. Ablation Study

Our proposed method contains three components: Cut^Mix, MMD loss \( L_{MMD} \) for feature level mutual learning, and peer teacher distillation loss \( L'_{PT} \). To analyze the effectiveness of each component, we conduct an ablation study with ResNet-110 on CIFAR-100. As illustrated in Table 3 each components are also jointly effective.

### 4. CONCLUSION

In this research, we first empirically illustrate the orthogonality to the performance gain between the mixed sample regularization and online distillation. Furthermore, we design an even stronger mixed sample regularization named Cut^Mix, together with a powerful online distillation framework to combine with Cut^Mix. The proposed framework can consistently boost the classification accuracy of an ensemble of peer student networks. Due to the restriction on computational resources, evaluations on large-scale datasets such as ImageNet are left to future work. Other future directions include the applications to more scenarios e.g., object detection, image segmentation.
5. REFERENCES

[1] G. Hinton, O. Vinyals, and J. Dean, “Distilling the knowledge in a neural network,” arXiv preprint arXiv:1503.02531, 2015.

[2] F. Sarfraz, E. Arani, and B. Zonooz, “Knowledge distillation beyond model compression,” in 2020 25th International Conference on Pattern Recognition (ICPR). IEEE, 2021, pp. 6136–6143.

[3] L. Yuan, F.E. Tay, G. Li, et al., “Revisiting knowledge distillation via label smoothing regularization,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 3903–3911.

[4] Y. Zhang, T. Xiang, T.M. Hospedales, et al., “Deep mutual learning,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 4320–4328.

[5] S. Yun, J. Park, K. Lee, et al., “Regularizing class-wise predictions via self-knowledge distillation,” in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 13876–13885.

[6] K. Kim, B. Ji, D. Yoon, et al., “Self-knowledge distillation with progressive refinement of targets,” in arXiv:2006.12000. 2020.

[7] T. DeVries and G.W. Taylor, “Improved regularization of convolutional neural networks with cutout,” arXiv preprint arXiv:1708.04552, 2017.

[8] H. Zhang, M. Cisse, Y.N. Dauphin, et al., “mixup: Beyond empirical risk minimization,” arXiv preprint arXiv:1710.09412, 2017.

[9] S. Yun, D. Han, S.J. Oh, et al., “Cutmix: Regularization strategy to train strong classifiers with localizable features,” in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 6023–6032.

[10] D. Das, H. Massa, A. Kulkarni, et al., “An empirical analysis of the impact of data augmentation on knowledge distillation,” arXiv preprint arXiv:2006.03810, 2020.

[11] H. Wang, S. Lohit, M. Jones, et al., “Knowledge distillation thrives on data augmentation,” arXiv preprint arXiv:2012.02909, 2020.

[12] S. Park, K. Yoo, and N. Kwak, “On the orthogonality of knowledge distillation with other techniques: From an ensemble perspective,” arXiv preprint arXiv:2009.04120, 2020.

[13] T.-B. Xu and C.-L. Liu, “Data-distortion guided self-distillation for deep neural networks,” in Proceedings of the AAAI Conference on Artificial Intelligence, 2019, vol. 33, pp. 5565–5572.

[14] X. Lan, X. Zhu, and S. Gong, “Knowledge distillation by on-the-fly native ensemble,” arXiv preprint arXiv:1806.04606, 2018.

[15] G. Wu and S. Gong, “Peer collaborative learning for online knowledge distillation,” in AAAI, 2021.

[16] D. Chen, J.-P. Mei, C. Wang, et al., “Online knowledge distillation with diverse peers,” in Proceedings of the AAAI Conference on Artificial Intelligence, 2020, vol. 34, pp. 3430–3437.

[17] Q. Guo, X. Wang, Y. Wu, et al., “Online knowledge distillation via collaborative learning,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 11020–11029.

[18] M. Long, Y. Cao, J. Wang, et al., “Learning transferable features with deep adaptation networks,” in International conference on machine learning. PMLR, 2015, pp. 97–105.

[19] A. Krizhevsky et al., “Learning multiple layers of features from tiny images,” 2009.

[20] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” arXiv preprint arXiv:1409.1556, 2014.

[21] K. He, X. Zhang, S. Ren, et al., “Deep residual learning for image recognition,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.

[22] S. Zagoruyko and N. Komodakis, “Wide residual networks,” arXiv preprint arXiv:1605.07146, 2016.

[23] G. Huang, Z. Liu, L. Van D.M., et al., “Densely connected convolutional networks,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 4700–4708.

[24] G. Xu, Z. Liu, X. Li, et al., “Knowledge distillation meets self-supervision,” in European Conference on Computer Vision. Springer, 2020, pp. 588–604.