A. Appendix

Quality of augmentation policies on Reduced SVHN. For Reduced SVHN dataset, we show data augmentations learned by MADAug are superior to AdaAug on the improvement of per-class accuracy, especially in the "4", "7", "8", and "9" classes in Figure 6.

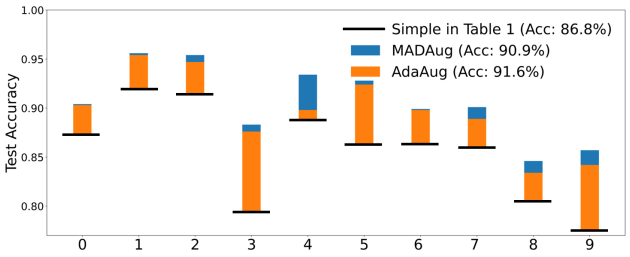

Figure 6: **MADAug and AdaAug's improvements to different classes on Reduced SVHN dataset.** Compared with AdaAug, MADAug enhances the test accuracy across various classes, particularly demonstrating a more notable positive impact on same classes such as "4", "7", "8", and "9".

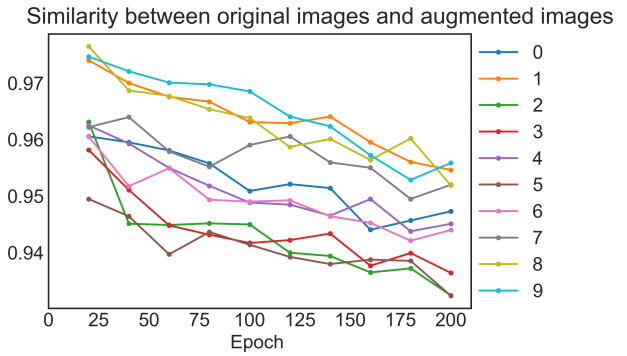

Figure 7: **Similarity between the original images and MADAug-augmented images at various training phases.** As the training process, MADAug gradually generates more "challenging" augmentation policies for images.

AdaAug [4] provides detailed augmentation policies for models throughout the entire training stage. These augmentation policies make samples distinguish from the original images, which can potentially hinder model convergence during the early stages of training. However, MADAug gradually applies the data augmentations for samples. Figure 7 demonstrates that as the training epoch progresses, more "adversarial" images generated by MADAug can be provided for the task model. Figure 8 visually illustrates that during the early training phase, the model receives the original images, while as the training progresses, MADAug learns and applies more "challenging" augmentation policies to augment the images. However, these policies are designed to avoid collapsing the intrinsic meanings of the images and instead, emphasize the crucial information within them.

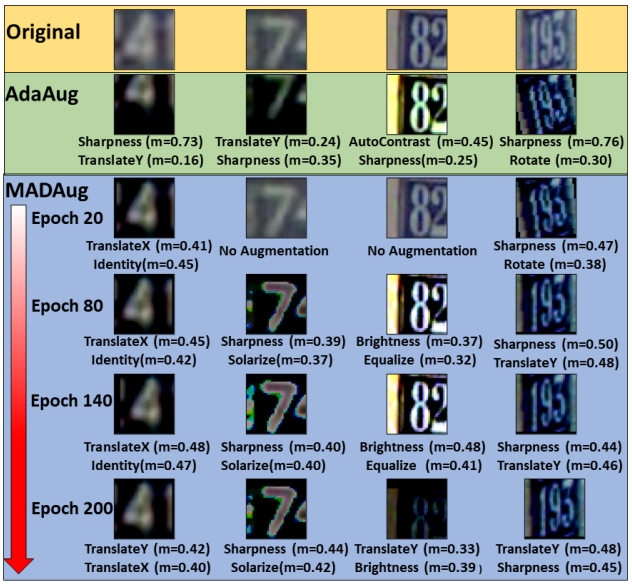

Figure 8: **Augmentations learned by AdaAug and MADAug applied to the "4", "7", "8", and "9" class images at various training epochs on Reduced SVHN dataset.** Each augmentation operation is formatted with its name and magnitude, respectively.

Advantages of MADAug over AdaAug. AdaAug employs a two-stage training process, where the first stage involves learning data augmentation strategies for individual samples through alternating "exploration" and "exploitation" steps. In the second stage, the learned strategies are fixed while training the task model. However, AdaAug exhibits limitations, including the inability to dynamically update the learned data augmentation strategies based on the performance of the current task model and the suboptimal choice of a two-stage training process.

To address these drawbacks, we propose the MADAug method that overcomes these limitations. In MADAug, we dynamically optimize the data augmentation strategies for each sample by leveraging the current model's performance on the validation set. This facilitates training the model with the most effective data augmentation for the given training stage. We adopt an end-to-end training approach for the task model, differing from AdaAug and AutoAugment, which utilize a two-stage training methodology.

Additionally, we discover that data augmentations do not improve model performance obviously in the early stages of training. Therefore, we use the monotonic curriculum strategy, gradually applying data augmentations to each sample

as the training progresses, thereby enhancing the robustness of the task model.

Through the use of MADAug, we demonstrate significant advancements over AdaAug, achieved by its ability to dynamically optimize data augmentation strategies and employ the monotonic curriculum strategy.

Model hyperparameters. We show the important hyperparameters on different benchmark datasets in Table 8. For other details on the hyperparameters and implementation, we display them in the open source code.

Table 8: **Hyperparameters on benchmark datasets.** We do not specifically tune these hyperparameters, and all of these are consistent with PBA and AutoAugment.

Dataset	Model	Learning Rate	Weight Decay	Batch Size	epoch
CIFAR-10	Wide-ResNet-40-2	0.1	5e-4	128	200
CIFAR-10	Wide-ResNet-28-10	0.1	5e-4	128	200
CIFAR-10	Shake-Shake (26 2x96d)	0.01	1e-3	128	1,800
CIFAR-10	PyramidNet+ShakeDrop	0.05	5e-5	64	1,800
CIFAR-100	Wide-ResNet-40-2	0.1	5e-4	128	200
CIFAR-100	Wide-ResNet-28-10	0.1	5e-4	128	200
CIFAR-100	Shake-Shake (26 2x96d)	0.01	2.5e-3	128	1,800
CIFAR-100	PyramidNet+ShakeDrop	0.025	5e-4	64	1,800
Reduced CIFAR-10	Wide-ResNet-28-10	0.05	5e-3	128	200
Reduced CIFAR-10	Shake-Shake (26 2x96d)	0.025	2.5e-3	128	1,800
SVHN	Wide-ResNet-28-10	0.005	1e-3	128	200
SVHN	Shake-Shake (26 2x96d)	0.01	1.5e-4	128	1,800
Reduced SVHN	Wide-ResNet-28-10	0.05	1e-2	128	200
Reduced SVHN	Shake-Shake (26 2x96d)	0.025	5e-3	128	1,800
ImageNet	ResNet-50	1.6	1e-4	4096	270