DA^2: Degree-Accumulated Data Augmentation on Point Clouds with Curriculum Dynamic Threshold Selection
Supplementary Material

Ta Chun Tai[1], Nhat-Tuong Do-Tran[1], Ngoc-Hoang-Lam Le[1], Yung-Hui Li[2], and Ching-Chun Huang[1]⋆

[1] National Yang Ming Chiao Tung University, Taiwan
{maxtai.10, tuongdotn.cs11, lengochoanglam.ee12, chingchun}@nycu.edu.tw
[2] Hon Hai Research Institute, Taiwan
yunghui.li@foxconn.com

In this supplementary material, we provide notations of symbols, training details, additional experiments and analyses, as well as sample visualizations. Due to space constraints, these elements are not included in the main paper.

A Notations of Symbols

Table 1: Table of Notations

Notation	Description
D_0	Initial dataset
D_t	Dataset at time t
T_{warmup}	Number of warmup epochs
T	Total number of epochs for the DA^2 stage
f_s	Student classifier
f_t	Teacher classifier
θ_s	Weights of the student classifier
θ_t	Weights of the teacher classifier
$A_\phi()$	Augmenter
ϕ	Weights of the augmenter
α	EMA decay rate
λ	Weight that determines the influence of the teacher
η_s	Learning rate for the student classifier
η_ϕ	Learning rate for the augmenter
$\hat{x}$	Original sample from D_0
$\mathcal{L}_{T\_S}$	Teacher-student loss

Continued on next page

⋆ Corresponding author.

Table 1: Table of Notations (continued)

Notation	Description
$\mathcal{L}_{SWD}$	Loss based on Sliced Wasserstein Distance
$\mathcal{L}_g$	Regularization loss on the squared outputs of f_s
$\mathrm{CE}(\cdot)$	Cross-entropy loss function
$\mathrm{SWD}(,)$	Sliced Wasserstein Distance between two point clouds
$\mathrm{EMD}(,)$	Earth Mover's Distance between two point clouds
$\mathcal{R}_{\theta_i}$	A linear projection onto direction θ_i
$\mathcal{L}_{aug}$	Total loss for training augmenter
$\mathrm{Threshold}_c$	Threshold of class c
r_h	Upper ratio of threshold
r_l	Lower ratio of threshold
γ	Decay rate controlling threshold decrease
μ_c	Average of true positive prediction scores for class c

B Training Details

B.1 Warm-up

To prevent the teacher model from erroneously guiding the augmenter to produce inappropriate samples during the early training stages, we initialize the teacher model with a warm-up phase. During this phase, we do not use the proposed augmentation; instead, the student classifier is trained solely on the original dataset D_0, while the teacher classifier copies the weights of the student classifier. After this warm-up stage, we then employ our augmenter and begin updating the teacher with Exponential Moving Average (EMA).

B.2 Accumulated Augmentation Degree Flow

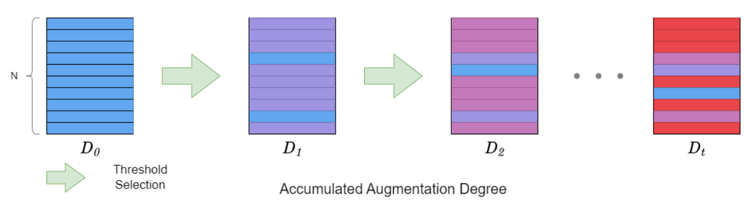

Fig. 1. Accumulated Degree Flow. The color spectrum represents the Accumulated Augmentation Degree. Samples that are filtered out by CDTS will be replaced with their original form from D_0, denoted as a reset operation. N is the size of the dataset.

Figure 1 illustrates the changes in the dataset during the interaction between our degree-accumulated augmentation and Curriculum Dynamic Threshold Selection (CDTS). The initial input is the original dataset D_0. After augmentation, all samples gain an augmentation degree. These samples are then filtered through the CDTS process, where qualified samples are retained, and unqualified samples are replaced with their original form from D_0. In D_1, the blue color represents the samples that were filtered out and reset.

This approach ensures that we do not continuously increase the number of samples and, consequently, the training cost. However, we can still learn from samples of varying difficulties. This design guarantees that the model learns the original data well despite continuous augmentation. The filtering mechanism ensures the model masters simple samples before moving on to more challenging ones.

B.3 Algorithm

To better understand the proposed method, we provide the following algorithm 1, detailing the implementation step-by-step.

C Additional Experiments and Analysis

C.1 Additional Experiments

We provide the additional comparison of different data augmentation methods on the PB_T50_RS of ScanObjectNN (SON) [4] and ModelNet40 (MN40) [6] datasets as shown in Table 2. Remarkably, our DA^2 has achieved an impressive overall accuracy (OA) of 86.6% with PointNet++ [2] and 86.3% with DGCNN [5] on the SON dataset. This shows a margin of 2.9% and 2.7% over the nearest competitor SageMix [1]. However, our DA^2 achieves comparable OA on the simpler MN40 dataset.

Table 2. Overall Accuracy (%) of data augmentation methods.

Classifier	Augmentation	SON [4]	MN40 [6]
PointNet++ [2]	+SageMix [1]	83.7	93.3
	+PointCutMix-K [7]	82.8	93.4
	+DA^2 (ours)	86.6	92.4
DGCNN [5]	+PointMixSwap [3]	-	93.5
	+SageMix [1]	83.6	93.6
	+PointCutMix-K [7]	82.9	93.1
	+DA^2 (ours)	86.3	92.6

Algorithm 1 Training procedure of (DA^2)

Require: Dataset D_0, warmup epochs T_{warmup}, total epochs T, student classifier f_s with weights θ_s, teacher classifier f_t with weights θ_t, augmenter with weights ϕ, EMA decay rate α, loss weight λ, learning rate η_s for student, learning rate η_ϕ for augmenter

Ensure: Trained student classifier f_s, trained teacher classifier f_t

Warmup Stage:
for $t = 0$ to T_{warmup} **do**
  Train f_s with D_0
  $\mathcal{L}_{cls} = CE(f_s(\hat{x}), y)$, $\hat{x} \in D_0$
  Update student classifier: $\theta_s \leftarrow \theta_s - \eta_s \nabla_{\theta_s} \mathcal{L}_{cls}$
  Update teacher classifier: $\theta_t \leftarrow \theta_s$
end for
DA^2 Stage:
for $t = 0$ to T **do**
  Input current dataset D_t (same size as D_0)
  1. Training Augmenter:
  $x' = A_\phi(x)$, $x \in D_t$
  $\mathcal{L}_{T\_S} = |1 - exp(CE(f_s(x'), y) - \lambda * CE(f_t(x'), y))|$
  $\mathcal{L}_{SWD} = SWD(\hat{x}, x')$
  $\mathcal{L}_g = ||f_s(x')^2||$
  $\mathcal{L}_{aug} = L_{T\_S} + L_{SWD} + L_g$
  Update augmenter: $\phi \leftarrow \phi - \eta_\phi \nabla_\phi \mathcal{L}_{aug}$
  2. Curriculum Dynamic Threshold Selection:
  $x' = A_\phi(x)$, $x \in D_t$
  if $(f_s(x')) <$ Threshold **then**
    $x' \leftarrow \hat{x}$
  end if
  Form new dataset D_{t+1}
  3. Student and Teacher Updating
  $\mathcal{L}_{cls} = CE(f_s(x), y)$, $x \in D_{t+1}$
  Update student classifier: $\theta_s \leftarrow \theta_s - \eta_s \nabla_{\theta_s} \mathcal{L}_{cls}$
  Update teacher classifier: $\theta_t \leftarrow \alpha\theta_t + (1 - \alpha)\theta_s$
end for

C.2 Visualization in Feature Space

To take a closer look at the behavior in feature space, we test our method using two classes as an example. In this experiment, we visualize the point cloud of the augmentation process to generate degree-accumulated multiple versions of an original "nightstand" sample, as shown in Fig. 2. From the feature space distribution, it is evident that one-step augmentation is insufficient to move away from the original samples. In contrast, by accumulating augmentations, our method explores a more diverse range and generates variant samples, leaving the original distribution while approaching the feature distribution of the other class. When the augmented samples move too far from their class distribution, CDTS pulls them back to the original point, preventing the generation of out-of-distribution samples.

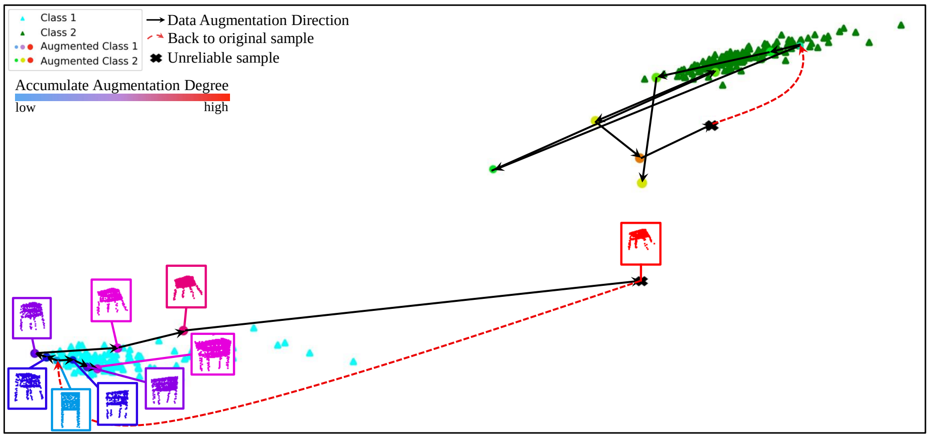

Fig. 2. We use PCA to visualize features on image. Augmentation process of two classes in the feature space, where each point represents a feature of a point cloud. To aid understanding of the augmentation level, we visualize their original point clouds. The visualization shows that the augmented feature points continue to explore the space until the two classes are nearly adjacent, suggesting the formation of a compact decision boundary.

Moreover, we can observe that after resetting to the original form, subsequent augmentations explore different directions, showcasing our method's ability to explore multiple directions and expand the exploration space. This behavior is also evident in the other class, where our method significantly broadens the distribution of the class through accumulated augmentations. In summary, our approach significantly enhances the exploration space. With the assistance of the teacher-guided auto augmenter and CDTS, augmented samples are controlled to avoid over-augmentation that may lead to misclassification.

C.3 Additional Ablation Study on Losses

To demonstrate that the effectiveness of our method is not solely based on the loss functions, we extended the ablation study from the main paper. We evaluated $\mathcal{L}g$ and $\mathcal{L}_{SWD}$ with and without using $\mathrm{DA}^2 + \mathrm{CDTS}$, analyzing the interaction between these two losses and our method.

From Table 3, it is evident that both $\mathcal{L}_g$ and $\mathcal{L}_{SWD}$ effectively improve model performance. When our framework ($\mathrm{DA}^2 + \mathrm{CDTS}$) is not used, augmentations are not accumulated, meaning the augmentation takes only one step. In this case, $\mathcal{L}_g$ is very effective, providing a repulsive force that effectively expands the distribution in this single step. However, adding $\mathcal{L}_{SWD}$ limits this single-step augmentation, making the training results closer to those without these losses.

When DA^2 is introduced, the model can perform more steps of augmentation. However, taking more steps significantly increases the risk of generating unsuitable

Table 3. Interaction between Losses and DA^2 with CDTS

TA	$\mathcal{L}_{T\_S}$	DA^2 + CDTS	$\mathcal{L}_{SWD}$	$\mathcal{L}_g$	OA	mAcc
✓	✓				87.37	85.81
✓	✓		✓		87.47	85.23
✓	✓			✓	87.96	86.41
✓	✓		✓	✓	87.54	85.66
✓	✓	✓	✓		86.81	84.80
✓	✓	✓		✓	87.68	85.46
✓	✓	✓	✓	✓	**88.55**	**87.37**

training samples, leading to learning in the wrong direction. This causes a performance drop when using either $\mathcal{L}_{SWD}$ or $\mathcal{L}_g$ alone. However, when both losses are present, along with DA^2 and CDTS, $\mathcal{L}_g$ helps the augmentation expand in a more generalized direction, while $\mathcal{L}_{SWD}$ provides sufficient constraints, resulting in the best performance.

This table also shows that each element in our design supports each other and is meaningful. Also, all these components must coexist to create a robust framework that ensures effective and balanced augmentation, achieving the best performance.

D Additional Visualization of Point Clouds

We visualize the augmented samples generated with our method, shown in Fig. 3. These accumulated samples are all extracted from the same 10 consecutive epochs. It can be observed that the accumulated degree varies for different samples within each epoch. When an augmented sample reverts back to the original sample, it indicates that the sample is too challenging and has been filtered out by the CDTS stage.

For example, the second one (chair) and the fifth one (car) continue to accumulate augmentations over ten epochs, indicating that the classifier is capable of learning from these augmented samples. Although the changes between successive epochs are not significant, the shape changes considerably after several epochs, demonstrating the effectiveness of our method. Moreover, the smooth variation in shape creates a smooth learning curve, enabling the classifier to learn difficult features. The sixth one (person) shows that in the initial iterations, all samples remain in their original form, indicating that the model has not yet learned this sample well. As a result, it is filtered out by our CDTS. Once the model has learned the sample, it starts accumulating augmentations to tackle more difficult variations. If the sample becomes too difficult to learn, it resets to the original data.

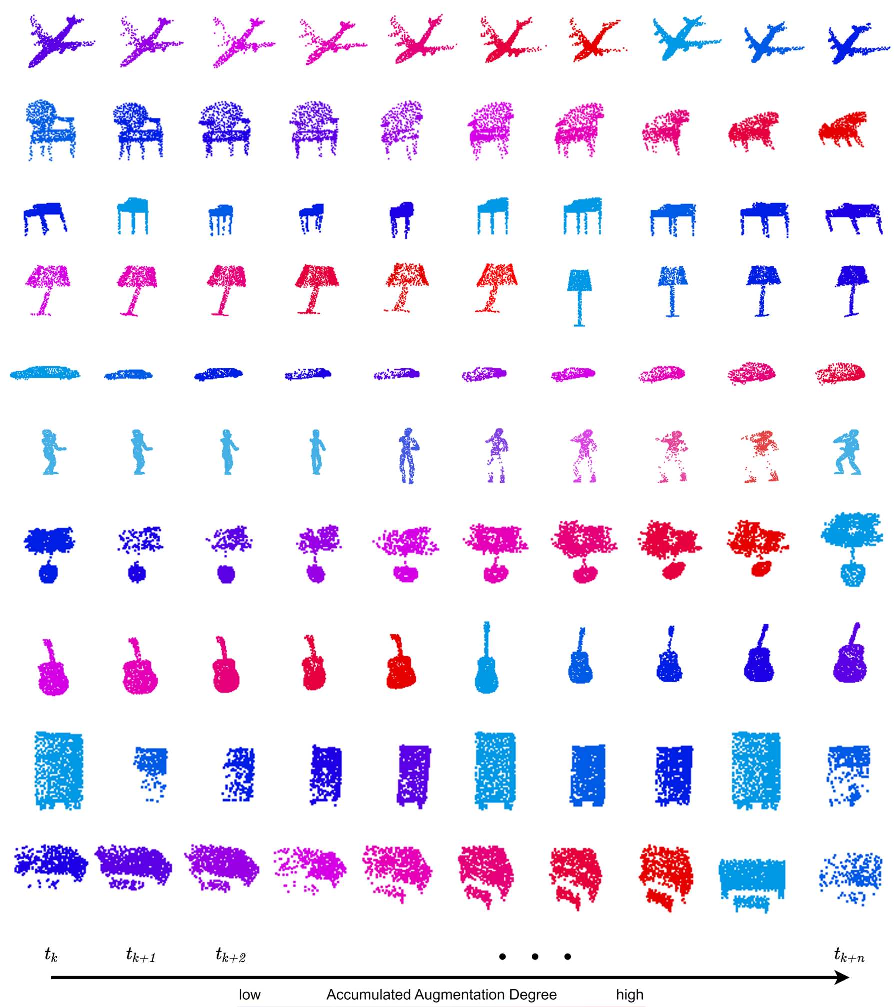

Fig. 3. Visualizations of augmented samples generated with our method. The correlation between the accumulated degree and the color scale is illustrated in the figure.

References

1. Lee, S., Jeon, M., Kim, I., Xiong, Y., Kim, H.J.: Sagemix: Saliency-guided mixup for point clouds. Advances in neural information processing systems **35**, 23580–23592 (2022)
2. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: Deep hierarchical feature learning on point sets in a metric space. Advances in neural information processing systems **30** (2017)

3. Umam, A., Yang, C.K., Chuang, Y.Y., Chuang, J.H., Lin, Y.Y.: Point mixswap: Attentional point cloud mixing via swapping matched structural divisions. In: European Conference on Computer Vision. pp. 596–611. Springer (2022)
4. Uy, M.A., Pham, Q.H., Hua, B.S., Nguyen, T., Yeung, S.K.: Revisiting point cloud classification: A new benchmark dataset and classification model on real-world data. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 1588–1597 (2019)
5. Wang, Y., Sun, Y., Liu, Z., Sarma, S.E., Bronstein, M.M., Solomon, J.M.: Dynamic graph cnn for learning on point clouds. ACM Transactions on Graphics (tog) **38**(5), 1–12 (2019)
6. Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., Xiao, J.: 3d shapenets: A deep representation for volumetric shapes. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1912–1920 (2015)
7. Zhang, J., Chen, L., Ouyang, B., Liu, B., Zhu, J., Chen, Y., Meng, Y., Wu, D.: Pointcutmix: Regularization strategy for point cloud classification. Neurocomputing **505**, 58–67 (2022)