

FPSMix: data augmentation strategy for point cloud classification

Taiyan CHEN, Xianghua YING (✉)

Key Lab of Machine Perception (MoE), School of Intelligence Science and Technology, Peking University,
Beijing 100871, China

© Higher Education Press 2025

Abstract Data augmentation is a widely used regularization strategy in deep neural networks to mitigate overfitting and enhance generalization. In the context of point cloud data, mixing two samples to generate new training examples has proven to be effective. In this paper, we propose a novel and effective approach called Farthest Point Sampling Mix (FPSMix) for augmenting point cloud data. Our method leverages farthest point sampling, a technique used in point cloud processing, to generate new samples by mixing points from two original point clouds. Another key innovation of our approach is the introduction of a significance-based loss function, which assigns weights to the soft labels of the mixed samples based on the classification loss of each part of the new sample that is separated from the two original point clouds. This way, our method takes into account the importance of different parts of the mixed sample during the training process, allowing the model to learn better global features. Experimental results demonstrate that our FPSMix, combined with the significance-based loss function, improves the classification accuracy of point cloud models and achieves comparable performance with state-of-the-art data augmentation methods. Moreover, our approach is complementary to techniques that focus on local features, and their combined use further enhances the classification accuracy of the baseline model.

Keywords point cloud classification, data augmentation, loss function, point cloud understanding, point cloud analysis

1 Introduction

In recent years, deep neural networks (DNNs) have shown remarkable performance in processing various types of data, including speech, image, video, and point cloud. PointNet [1] has pioneered the direct application of DNNs to process 3D point cloud data without preprocessing, which has sparked growing interest in the field [2–5]. However, to enhance the robustness of DNN models, it is crucial to increase the diversity of training data. Unfortunately, 3D point cloud datasets often have limited samples compared to image datasets. For instance, the widely used image benchmark dataset ImageNet [6] consists of over one million samples

from 1000 categories, whereas the point cloud benchmark dataset ModelNet40 [7] contains only 12311 samples across 40 categories. This scarcity of quantity and diversity in point cloud datasets can result in overfitting issues and hinder the robustness and generalization performance of DNN models.

Data augmentation (DA) is a commonly used strategy to mitigate overfitting and improve the generalization capability of models. In the context of point cloud data, previous studies on DA can be broadly categorized into two streams.

The first stream involves generating new point clouds by applying various perturbations, such as scaling, translation, rotation, and jittering, to the original point clouds in 3D space. Conventional data augmentation (ConvDA) [1,2] typically applies these operations within a small and fixed predefined range to preserve the class label. Another approach [8] formulates a point augmentation function with shape-wise transformations and point-wise displacements, whose parameters are sample-aware and learnable.

The second stream draws inspiration from mixed sample DA methods in the image domain [9,10]. These methods involve mixing two samples, such as replacing parts of a sample with subsets from another one [11,12], or interpolating between point clouds to generate new samples [13].

In this paper, we propose Farthest Point Sampling Mix (FPSMix), a mix-based strategy for augmenting point cloud data. Farthest Point Sampling (FPS) is widely used [2,14,15] due to its ability to uniformly downsample point clouds while preserving as many features as possible, summarized as iteratively selecting the farthest point from the existing sample points. Our proposed method directly mixes two point clouds and performs the FPS operation on the mixed point cloud. This approach is straightforward to implement and computationally efficient, guiding deep neural networks to learn better global features. Combined with other data augmentation methods that focus on local features, such as PointCutMix [12], DNNs can achieve higher classification accuracy.

In this study, we address a limitation of existing mix-based point cloud augmentation methods, where the proportion of points is used as the weight for soft labels. While this approach may seem intuitive, it is not necessarily reasonable as the number of points does not always accurately represent the significance of features. To overcome this limitation, we

propose a novel significance-based loss function that replaces the proportion of the number of sample points with the classification loss of the partial point cloud as the weight. This new approach guides the model to converge towards higher classification accuracy by taking into account the importance of features rather than solely relying on the number of points.

This paper presents several significant contributions, including

- A novel augmentation method. We propose FPSMix, a novel data augmentation method for point clouds that is easy to implement and highly effective. Our approach achieves comparable classification accuracy to state-of-the-art methods and is also validated on the task of normal estimation.
- A novel significance-based loss. We introduce a significance-based loss that replaces the proportional-based loss commonly used in mix-based data augmentation approaches. Our proposed loss takes into account the importance of features, leading to improved performance of the augmentation method.
- Complementary method. FPSMix and the significance-based loss can be used in conjunction with other data augmentation methods, such as ConvDA and PointCutMix. Extensive ablation studies demonstrate that our proposed approach can be seamlessly combined with existing methods to enhance the robustness of the baseline model.

2 Related work

2.1 Deep learning on point clouds

PointNet [1] is a pioneering method that directly applies deep neural networks (DNNs) to point clouds. It uses shared point-wise multi-layer perceptions (MLPs) and max pooling to ensure permutation invariance and learn global features from point clouds. Subsequent methods have focused on capturing local features by hierarchically grouping points in various ways [2, 4, 16–19]. Another approach involves graph-based networks [3, 20–22], where each point cloud is treated as a graph and each sample point is considered as a vertex in the graph. Additionally, some methods [5, 23–27] project non-gridded point clouds into a regular space or interpolate weights in gridded convolutional kernels to apply conventional convolutions for point cloud processing. Recently, Xiang et al. [28] propose to group and aggregate sequences of points (curves) for better depiction of point cloud object geometries. And [29, 30] try to use Transformer for point cloud data. And there are also studies of various point cloud problems [31–33].

2.2 Mix-based data augmentation in the image domain

Data augmentation (DA) is a popular regularization strategy used in training deep neural networks (DNNs) to improve their generalization performance. Mix-based DA [9, 10, 34–37] is a widely used approach in the image domain for generating new training samples by blending existing samples. Mixup [9] and CutMix [10] are two representative methods in this category. Mixup interpolates the whole images and their labels from the training data to create new training samples,

while CutMix cuts out a rectangular area from one image and covers it with another image, using the proportion of the covered area to the uncovered area as the weight for the soft labels after mixing.

2.3 Data augmentation on point clouds

Existing data augmentation methods for 3D point clouds can be broadly categorized into three types: conventional methods [2], PointAugment [8], and mix-based methods [11–13]. Conventional methods [2] apply combinations of scaling, rotation, translation, and jittering within a small range to the original point clouds to generate new samples while preserving the original labels. PointAugment [8] is an auto-augmentation framework that consists of an augmenter and a classifier. The augmenter generates shape-wise transformation parameters and point-wise displacement parameters, both of which are learnable, and the classifier classifies samples before and after augmentation, with joint optimization. Mix-based methods [11–13] extend the idea of mix-based data augmentation from the image domain to point clouds. For example, Chen et al. [13] propose an extension of Mixup [9] to point clouds, using linear interpolation between point clouds, while Lee et al. [11] and Zhang et al. [12] extend CutMix [10] to point clouds by replacing subsets of points from one sample with subsets from another sample. In this work, we propose FPSMix, a novel mix-based approach for augmenting point cloud samples while preserving their original contours, which is crucial for point cloud classification tasks.

2.4 Soft label in mix-based augmentation methods

A soft label is a set of scores, each one of which implies the probability or likelihood of a mixed sample belongs to one class. In [9–12], the new sample's soft label can be written as

$$c = \lambda c_1 + (1 - \lambda)c_2. \quad (1)$$

c_1 and c_2 are one-hot labels of samples before mixing, for [9], $\lambda \in [0, 1]$ is the summed weights of the first image, and $1 - \lambda$ is the summed weights of the second image. In [10], λ is the ratio of the number of pixels from the first image to the total number of pixels in the generated image. Similarly, in [11, 12], λ is the ratio of the number of points from the first point cloud to the total number of points in the generated point cloud. In this paper, we propose a significance-based loss that no longer uses a proportional-based loss, but uses the respective classification loss of the two parts as the score of the soft label.

3 Method

3.1 Problem setting

Given a training data set $\{(P_i, c_i)\}_{i=1}^N$ consisting of N point clouds, the goal of point cloud classification task is to learn a function $f: P_i \rightarrow [0, 1]^C$ that maps each point cloud to a C classes one-hot semantic label distribution vector. Here $P_i \in \mathbf{R}^{M \times d}$ is a point cloud consists of M sample points $P_i = \{p_m^i\}_{m=1}^M$, and each sample point is a d -channel vector. In this paper, we only use the 3D coordinate information, so $d = 3$. And c_i is the one-hot label with respect to P_i . The goal of

training is to learn optimal parameters θ^* of function f by minimizing the loss

$$\theta^* = \arg \min_{\theta} \sum_{x \in \mathcal{T}} \mathcal{L}_{\mathcal{T}}(f(P), c), \quad (2)$$

where \mathcal{T} is the training data set and \mathcal{L} is the loss function.

3.2 FPSMix strategy

The algorithm for FPS is that after selecting an initial point, iteratively select the point farthest from the current point set and add it to the point set until the number of points in the set reaches the sampling requirement. Since FPS can maintain the contours of the point cloud very well, an intuitive mix-based sample generation method can thus be proposed, which we call FPSMix.

The key idea of FPSMix is to generate a new training sample (\tilde{P}', \tilde{c}') from two given point clouds (P_1, c_1) and (P_2, c_2) . Here, as in previous, P is the training point cloud while c is the corresponding label. The combining operation is

$$\tilde{P}' = FPS(P_1 \parallel P_2, 0.5). \quad (3)$$

here $P_1, P_2 \in \mathbf{R}^{M \times d}$, $FPS(P_i, \mu)$ means sampling the points with the ratio μ from point cloud P_i with the FPS method. And \parallel is concatenation operation, means to directly mix up two point clouds. For example, in Eq. (3), $P_1 \parallel P_2$ makes the result $\tilde{P} \in \mathbf{R}^{2M \times d}$ is a point cloud that contains all sampling points from P_1 and P_2 . The farthest point sampling operation is then performed on \tilde{P} , such that the sampling result $\tilde{P}' \in \mathbf{R}^{M \times d}$ is the final generated new training sample.

$\lambda \in [0, 1]$ is the mixing rate, representing the proportion of points from the original sample P_1 to the total number of points in the final training sample \tilde{P}' . For example, \tilde{P}' has M sample points, of which there are m_1 points from P_1 , then $\lambda = \frac{m_1}{M}$. The execution process of FPSMix is shown in Fig. 1.

3.3 Significance-based loss function

Existing mix-based point cloud data augmentation methods [11,12] use the proportion of points from the original point clouds as the weight of soft labels, which we call proportional-based loss. Specifically, the total loss is

$$Loss_{total} = \lambda Loss(f(\tilde{P}'), c_1) + (1 - \lambda) Loss(f(\tilde{P}'), c_2), \quad (4)$$

where $f(\tilde{P}')$ is the predicted logits obtained after inputting the mixed point cloud \tilde{P}' to the model. The proportional-based loss is intuitive but not entirely reasonable, because the proportion of points from a particular point cloud does not represent its significance, i.e., the contribution to classifying it correctly. Based on this consideration, we separate the sample points from the two original point clouds to form two new point clouds, and feed these two point clouds into the model for classification. The classification loss represents the significance of the corresponding part of the point cloud. The smaller the loss value is, the easier this point cloud is to classify, and the higher the significance of its features is, the lower the weight of its category in the soft label should be when classifying the mixed samples. Conversely, the larger the loss value is, the higher the weight of its category in the soft label should be. Therefore, we can directly use the classification loss of these two point clouds as the weights of the soft labels of the mixed samples. This is more reasonable than using the proportional-based loss.

Expressed in symbolic language, for mixed sample \tilde{P}' , note points sampled from point cloud P_1 be \tilde{P}'_1 , points from P_2 be \tilde{P}'_2 , then

$$\tilde{P}' = \tilde{P}'_1 \parallel \tilde{P}'_2. \quad (5)$$

We rewrite Eq. (4) as

$$Loss_{total} = Loss(f(\tilde{P}'_1), c_1).detach() \times Loss(f(\tilde{P}'), c_1) + Loss(f(\tilde{P}'_2), c_2).detach() \times Loss(f(\tilde{P}'), c_2). \quad (6)$$

We name Eq. (6) as significance-based loss. We compare the two loss functions in our experiments and find that Eq. (6) can guide the model to converge to higher accuracy.

In addition, the significance-based loss can also be understood as a regularization mechanism similar to dropout [2], where the baseline model should still correctly classify the point cloud after dropping a portion of the points. However, using significance-based loss can achieve better results than using dropout [2] for data augmentation, which we will verify experimentally later.

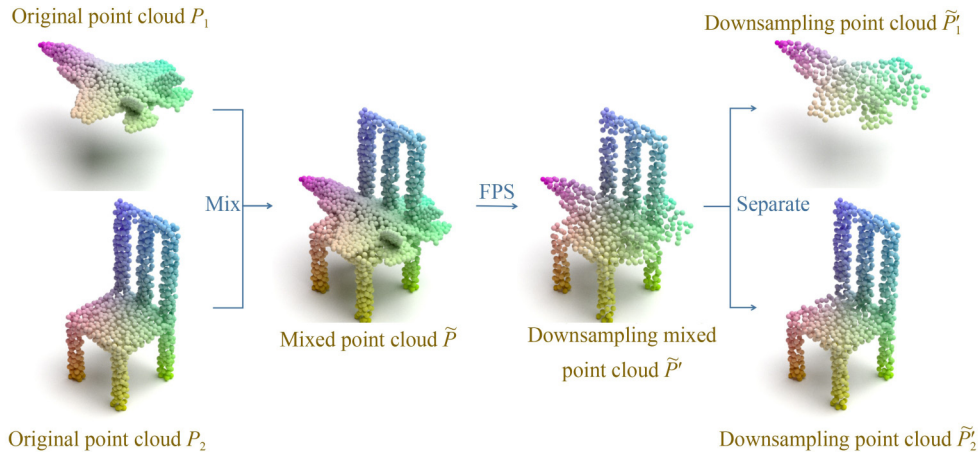


Fig. 1 Illustration of FPSMix. “Separate” step is for significance-based loss in Eq. (6)

4 Experiments

4.1 Experiment setups

4.1.1 Datasets

We perform FPSMix experiments on ModelNet40 [7] and ModelNet10 [7], which are widely used benchmark datasets for point cloud classification. ModelNet40 contains 12311 CAD models in 40 categories, 9843 samples of which are used for training and 2468 for testing. ModelNet10 is a subset of ModelNet40, it contains 4899 samples in 10 categories, 3991 samples of which are used for training and the rest are used for testing. For each sample, we use 1024 of all points and for all experiments, we only use xyz coordinates.

4.1.2 Backbone networks

We select three representative neural networks for 3D point cloud as the backbone networks for our experiments: PointNet [1], PointNet++ [2], DGCNN [3]. PointNet only utilizes global features, and the other two networks take local information into account. We employ FPSMix to these networks to show it is a general data augmentation method and it is agnostic to networks' architecture.

4.1.3 Implementation details

Our experimental code is all implemented using the Pytorch [38] framework. All models are trained for 250 epochs with a batch size of 24. In PointNet++, The multi-scale grouping radius set as [0.1, 0.2, 0.4] and [0.2, 0.4, 0.8]. For DGCNN, the number of neighbors when obtaining graph features is set to 20. The network parameters use their default values. For all baseline models, we use Adam [39] optimizer, we use cosine annealing strategy [40] to adjust the learning rate with the initial learning rate being 0.001 and the end learning rate being 0. Point clouds are augmented with translation and scaling if not specified. We gradually increase the proportion of mixed samples to the training samples during the training process, initially to zero and up to 50%. And *Loss* in Eqs. (4) and (6) uses label-smoothing loss [3], and the ϵ parameter for label-smoothing loss is set to 0.4.

4.2 Point cloud classification

We perform FPSMix for point cloud classification using three backbone networks on ModelNet40 and ModelNet10, and all experimental settings are described in previous. The results are shown in Tables 1 and 2.

Table 1 ModelNet40 classification results

Method	PointNet	PointNet++	DGCNN
baseline	89.2	90.7	92.3
PointMixup-U [13]	89.9	91.7	–
PointMixup-A [13]	–	92.7	92.9
PointAugment [8]	90.9	92.9	93.4
PointCutMix-R [12]	89.2	92.8	92.8
PointCutMix-K [12]	89.1	93.4	93.1
PointCutMix-S [12]	–	93.4	93.2
RSMix [11]	88.7	92.7	93.5
Ours (FPSMix)	90.9	93.4	93.2
Ours (FPSMix+s-loss)	91.2	93.4	93.3

Table 2 ModelNet10 classification results

Method	PointNet	PointNet++	DGCNN
baseline	91.9	93.3	94.8
PointAugment [8]	94.1	95.8	96.7
PointCutMix-R [12]	94.5	96.3	95.2
PointCutMix-K [12]	94.2	95.7	95.7
RSMix [11]	93.1	94.3	95.9
Ours (FPSMix)	93.8	95.3	95.3
Ours (FPSMix+s-loss)	95.3	95.5	95.6

In Tables 1 and 2, baseline refers to the classification accuracy reported in original papers, s-loss refers to our proposed significance-based loss. PointMixup-U and PointMixup-A represent the results on the unaligned and pre-aligned ModelNet40 dataset. PointCutMix-R and PointCutMix-K represent the replaced points that are randomly sampled and optimal alignment sampled, PointCutMix-S introducing the saliency map to guide the selection of the center point. And for RSMix, for a fair comparison, we select the results of the single-view evaluation.

From Table 1, we can see that our approach obtains improvements and outperforms PointMixup on all baseline models. For some of the state-of-the-art methods, i.e., PointAugment, PointCutMix, and RSMix, our method also achieves comparable results. And as shown in Fig. 2, on the three baseline models i.e., PointNet, PointNet++, and DGCNN, the FPSMix we proposed can stably improve the test accuracy. Furthermore, if s-loss is added during training, the test accuracy of the model will be stabilized at a higher level.

It is worth mentioning that our FPSMix makes PointNet obtain a great improvement, even surpassing PointNet++

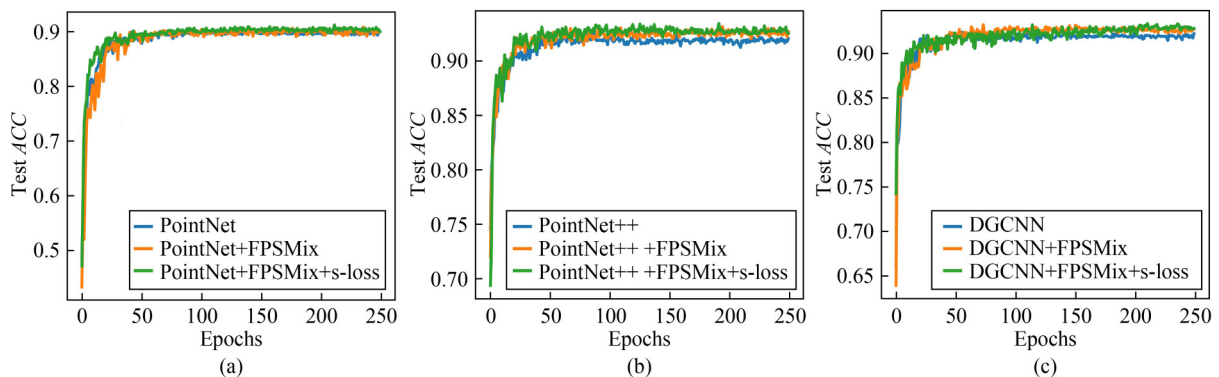


Fig. 2 The test accuracy of baseline models, models using FPSMix and s-loss change as the training progresses. From left to right: (a) PointNet, (b) PointNet++, (c) DGCNN

using RSMix and DGCNN using PointCutMix-R on ModelNet10. This is because our FPSMix makes the mixed samples retain the contours of the two original point clouds so that PointNet learns global features better.

These results are exciting because FPSMix is simpler than the existing methods. PointMixup needs to pre-process point clouds to align them in the horizontal facing direction. PointAugment takes an adversarial learning strategy and requires an additional network as an augments, which needs more computational resources. Our approach does not require pre-alignment and adversarial learning. For PointCutMix and RSMix, our results are comparable to theirs, and analysis by cProfile tool shows that our method runs at a faster speed than theirs. We load all training set data with batch size as 24, for all batches, we augment them with PointCutMix, RSMix, and our FPSMix, we run them for 250 epochs and use cProfile to get the cumulative time, the results are shown in Table 3. As can be seen from Table 3, our method has an advantage over RSMix and PointCutMix in terms of the speed of generating new samples.

The prediction results of some mixed samples in the training process are shown in Fig. 3. Some classes with similar shapes such as vases and bottles, chairs and sofas are easily misclassified in mixed samples. By allowing the model to discriminate difficult samples during training, the model enhances its ability to discriminate between classes of similar appearance and can achieve higher classification accuracy when tested on clean samples.

Table 3 Run time of mix-based methods to generate new samples

Method	Cumulative time/s
PointCutMix-R	4.275
PointCutMix-K	4.643
RSMix-knn	3.976
RSMix-ball	4.446
Ours	3.710

4.3 Ablation studies

4.3.1 Evaluations with various combinations of conventional augmentations and FPSMix

In this part, we focus on the effectiveness of FPSMix in combination with conventional data augmentation (ConvDA) methods and the advantages of FPSMix over mixing two point clouds and sampling them randomly.

FPSMix can be combined with existing data augmentation methods to further increase the diversity of training data because these methods are independent. We follow [8,12] to take translation and scaling as the ConvDA.

As shown in Tables 4 and 5, whether used alone or in conjunction with ConvDA, FPSMix can improve the classification accuracy of the backbone network. In particular, we randomly replace a portion of sample points of one point cloud with the same number of sample points of another point cloud, calling this method RandMix. We found that FPSMix achieved a greater improvement than RandMix when ConvDA are also used. This shows that FPSMix does retain global

Table 4 Comparison of random mixing and FPSMix. ACC_{PN} , ACC_{PN2} , ACC_D are the accuracy of PointNet, PointNet++, and DGCNN

DA method	ACC_{PN}	ACC_{PN2}	ACC_D
ConvDA	90.0	92.5	92.5
ConvDA+RandMix	90.6	92.9	92.9
ConvDA+FPSMix	91.0	93.4	93.2

Table 5 Ablation studies on evaluation accuracy of three baselines on ModelNet40. ACC_{PN} , ACC_{PN2} , ACC_D are the accuracy of PointNet, PointNet++, and DGCNN

ConvDA	FPSMix	s-loss	ACC_{PN}	ACC_{PN2}	ACC_D
			88.5	92.8	92.0
✓			90.0	92.5	92.5
	✓		89.8	93.0	92.2
✓	✓		91.0	93.4	93.2
✓	✓	✓	91.2	93.4	93.3

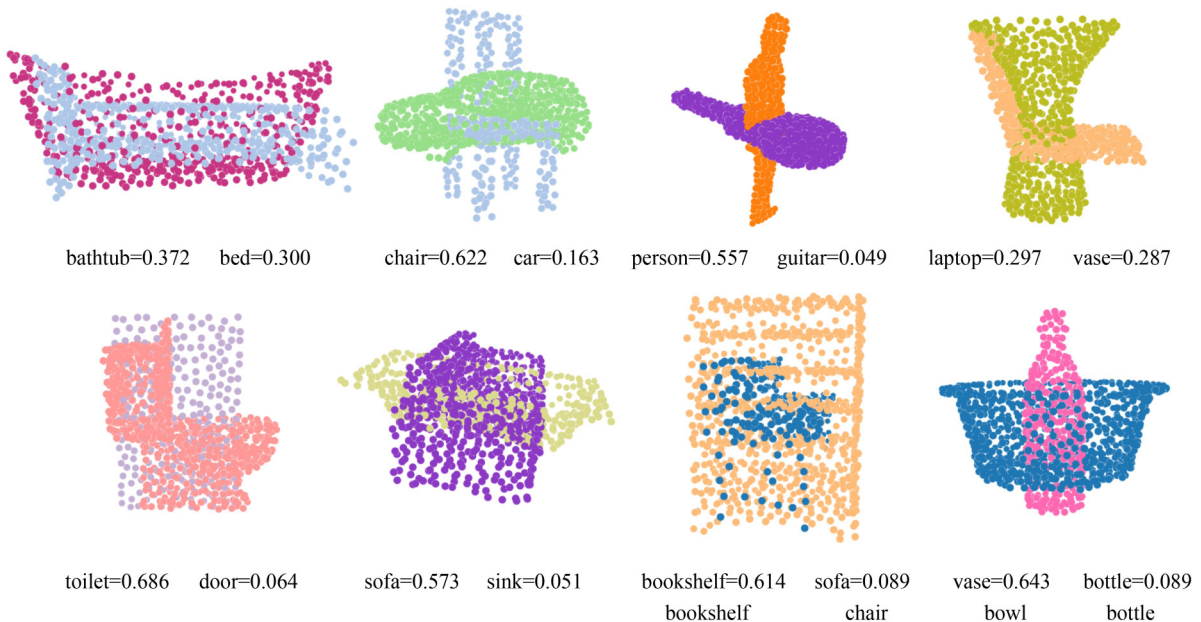


Fig. 3 Some examples of mixed-sample prediction results, with the wrong prediction results highlighted in red, and the correct label is marked on the second line

features better than RandMix. And our significance-based loss further improves classification accuracy based on applying FPSMix.

4.3.2 Evaluations with various combinations of augmentations and significance-based loss

We experimentally confirm that significance-based loss is effective not only for our proposed FPSMix but also for other mix-based data augmentation methods, such as PointCutMix.

In Table 6, when significance-based loss is not marked, the loss function is proportional-based loss if FPSMix or PointCutMix is used, otherwise it is a label-smoothing classification loss, and the baseline model is DGCNN. Regardless of whether the ConvDA is used or not, using significance-based loss when using FPSMix will result in higher classification accuracy than using proportional-based loss, and the same conclusion can be obtained when using PointCutMix.

4.3.3 Global and local features

In the previous subsection, we saw that the PointNet, which uses only global features of the point cloud, gets a great improvement when using our FPSMix, which shows that our proposed FPSMix enables baseline models to learn global features of the point cloud better. However, the current state-of-the-art point cloud classification models such as [26,28] emphasize the importance of local features, and [2,3] also take local features into consideration.

An intuitive idea is whether combining our methods with another method that focuses on local features, i.e., PointCutMix, can achieve better results. We use DGCNN with ConvDA as the baseline model, gradually increase the probability of performing mix-based DA from 0% to 100% as training progresses, and if DA needs to be performed, randomly select FPSMix or PointCutMix as the DA method with 50% probability, the experimental results are shown in Table 6. Table 6 shows that the combination of using FPSMix and PointCutMix with significance-based loss enables the model to better learn local and global features and improve the model accuracy.

It is worth noting that the result obtained by combining the two methods is not better than that obtained by using one method alone, and must be used in combination with significance-based loss to obtain a higher accuracy, which demonstrates the effectiveness of this loss function.

Table 6 Ablation studies of significance-based loss

ConvDA	PointCutMix	FPSMix	s-loss	ACC
				92.0
				92.5
✓	✓			93.1
	✓		✓	93.1
✓	✓		✓	93.2
✓		✓		93.2
		✓	✓	92.9
✓		✓	✓	93.3
✓	✓	✓		92.7
✓	✓	✓	✓	93.7

4.3.4 Significance-based loss as a regularization compare with random drop

As mentioned in previous, the significance-based loss can be considered as a regularization method for DNNs. This is because the weight of significance-based loss is determined by the classification loss of the new point cloud formed by removing some points from the original point cloud, which is similar to the drop out data augmentation method [2]. However, significance-based loss is a better regularization strategy due to the fact that its sample points are obtained from the FPS strategy, which can better preserve the shape information of the point cloud, while drop out is a random sampling. In fact, as shown in Table 7, higher accuracy was achieved using significance-based loss than using the strategy of randomly dropping 50% of the points when using ConvDA and FPSMix.

4.3.5 Robustness test

To verify that our methods make the model more robust to noise, we test the robustness of FPSMix and significance-based loss with PointNet and DGCNN in four noisy environments: scaling, rotation, jittering, and drop point. Same as previously, we use random translation and scaling as ConvDA to train the model. The results in Table 8 show that except for the case of 90° rotation around the Z-axis, the robustness of the model is significantly enhanced by using FPSMix. And if the FPSMix is used in combination with significance-based loss, the ability to fight against rotation, jittering, and scaling has been reduced, but the ability to correctly classify despite the loss of sample points has improved. And as shown in Table 9, the DGCNN model trained using a combination of ConvDA, FPSMix,

Table 7 Significance-based loss compare with random drop

Model	s-loss	Random drop	ACC
PointNet	✓		91.2
		✓	90.8
PointNet++	✓		93.4
		✓	93.2
DGCNN	✓		93.3
		✓	92.8

Table 8 Robustness test of FPSMix and significance-based loss with ConvDA for PointNet on ModelNet40 using random scaling, rotation, jittering, and drop point to add noise to the data

Transform	ConvDA	ConvDA+ FPSMix	ConvDA+ FPSMix + s-loss
Jitter ($\sigma^2 = 0.01$)	0.894	0.904	0.902
Jitter ($\sigma^2 = 0.05$)	0.852	0.884	0.872
Scale (0.75)	0.895	0.903	0.897
Scale (1.25)	0.893	0.903	0.900
X-asix 90°	0.156	0.175	0.157
X-asix 180°	0.364	0.381	0.374
Y-asix 90°	0.491	0.515	0.504
Y-asix 180°	0.364	0.381	0.374
Z-asix 90°	0.142	0.127	0.148
Z-asix 180°	0.355	0.368	0.364
Drop 100	0.902	0.907	0.909
Drop 200	0.900	0.904	0.905

Table 9 Robustness test of FPSMix, PointCutmix and significance-based loss with ConvDA for DGCNN on ModelNet40 using random scaling, rotation, jittering, and drop point to add noise to the data

Transform	ConvDA	ConvDA+ FPSMix	ConvDA + FPSMix + s-loss	ConvDA + FPSMix + PointCutmix + s-loss
Jitter ($\sigma^2 = 0.01$)	0.915	0.917	0.920	0.928
Jitter ($\sigma^2 = 0.05$)	0.903	0.907	0.909	0.917
Scale (0.75)	0.918	0.920	0.921	0.928
Scale (1.25)	0.921	0.929	0.926	0.934
X-axis 90°	0.200	0.201	0.214	0.225
X-axis 180°	0.416	0.447	0.447	0.474
Y-axis 90°	0.547	0.585	0.560	0.597
Y-axis 180°	0.727	0.756	0.737	0.765
Z-axis 90°	0.180	0.203	0.195	0.215
Z-axis 180°	0.388	0.400	0.404	0.447
Drop 100	0.916	0.925	0.927	0.932
Drop 200	0.905	0.915	0.924	0.931

PointCutMix, and significance-based loss has the highest classification accuracy as well as superior performance against various types of transformations. It is worth noting that the baseline model trained with FPSMix data augmentation showed good noise immunity. One possible interpretation of this question is that in mixed samples, points from one point cloud P_1 serve as a noisy perturbation for correctly classifying the other point cloud P_2 , and they significantly alter the shape of the surface of P_2 , and vice versa. Therefore, the baseline model perceives mixed samples as noisy data and must learn to accurately classify them. This helps the model maintain its ability to make accurate classifications even when faced with noise in the test data.

4.4 Object normal estimation

Estimating object normals is a crucial task in 3D modeling and rendering, as it requires a comprehensive understanding of the object’s shape. In this study, we validate the effectiveness of our proposed FPSMix data augmentation approach for estimating object normals using the widely used ModelNet40 dataset. As baselines, we compare our results with those obtained using PointNet and DGCNN, which are trained with an initial learning rate of 0.05 and cosine annealing scheduled to 0.0005 over 200 epochs, following a similar setup as [28]. The loss function is a cosine-loss. Specifically, it is

$$Loss = 1 - \cos(n_{pred}, n_{gt}), \quad (7)$$

where n_{pred} is the predicted normal, and n_{gt} is the ground truth normal.

Figure 4 presents the average cosine-distance error comparisons between the baseline models and the models trained with FPSMix. Our results show that incorporating FPSMix as a data augmentation method during training significantly reduces the error of normal estimation. This suggests that FPSMix is effective in improving the accuracy of object normal estimation, which is essential for 3D modeling and rendering tasks.

5 Conclusion

In this paper, we propose FPSMix, a novel data augmentation method specifically designed for point cloud classification

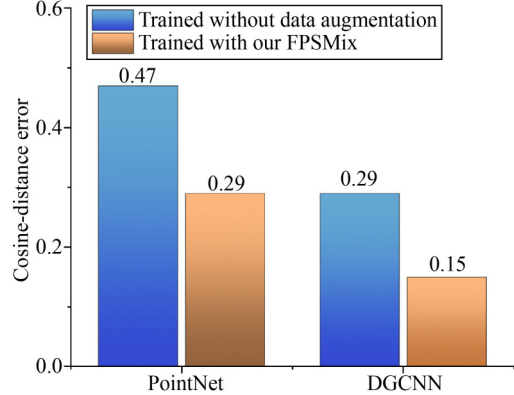


Fig. 4 Normal estimation results in average cosine-distance error

tasks. By leveraging Farthest Point Sampling (FPS) on the mixed point cloud, we generate new training samples that enhance the diversity of the training data. Compared to existing methods, FPSMix has the advantage of producing new samples more efficiently.

One key contribution of our approach is the introduction of a significance-based loss function. Unlike traditional methods that use the proportion of points as weights for soft labels, our approach assigns weights to soft labels based on the classification loss of two partial point clouds separated from the mixed sample. This novel approach improves the accuracy of the classification task, as demonstrated in various experiments.

Our results show that FPSMix, combined with the significance-based loss function, enhances the ability of deep neural networks to extract global features, leading to improved model robustness and comparable classification accuracy with state-of-the-art data augmentation methods. Furthermore, when used in conjunction with PointCutMix, our FPSMix method allows the model to learn both local and global features more effectively, resulting in higher accuracy. In addition to point cloud classification, we also demonstrate the effectiveness of FPSMix in the task of normal vector estimation.

Acknowledgements This work was supported by the National Key R&D Program of China (No. 2020YFB1708002) and the National Natural Science Foundation of China (Grant Nos. 62371009 and 61971008).

Competing interests The authors declare that they have no competing interests or financial conflicts to disclose.

References

- Qi C R, Hao S, Mo K, Guibas L J. PointNet: deep learning on point sets for 3D classification and segmentation. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. 2017, 77–85
- Qi C R, Li Y, Hao S, Guibas L J. PointNet++: deep hierarchical feature learning on point sets in a metric space. In: Proceedings of the 31st International Conference on Neural Information Processing Systems. 2017, 5105–5114
- 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, 2019, 38(5): 146
- Liu Y, Fan B, Xiang S, Pan C. Relation-shape convolutional neural network for point cloud analysis. In: Proceedings of IEEE/CVF

- Conference on Computer Vision and Pattern Recognition. 2019, 8887–8896
5. Thomas H, Qi C R, Deschaud J E, Marcotegui B, Goulette F, Guibas L. KPConv: flexible and deformable convolution for point clouds. In: Proceedings of IEEE/CVF International Conference on Computer Vision. 2019, 6410–6419
 6. Deng J, Dong W, Socher R, Li L J, Li K, Fei-Fei L. ImageNet: a large-scale hierarchical image database. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. 2009, 248–255
 7. 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 IEEE Conference on Computer Vision and Pattern Recognition. 2015, 1912–1920
 8. Li R, Li X, Heng P A, Fu C W. PointAugment: an auto-augmentation framework for point cloud classification. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020, 6377–6386
 9. Zhang H, Cisse M, Dauphin Y N, Lopez-Paz D. *mixup*: beyond empirical risk minimization. In: Proceedings of the 6th International Conference on Learning Representations. 2018
 10. Yun S, Han D, Chun S, Oh S J, Yoo Y, Choe J. CutMix: regularization strategy to train strong classifiers with localizable features. In: Proceedings of IEEE/CVF International Conference on Computer Vision. 2019, 6022–6031
 11. Lee D, Lee J, Lee J, Lee H, Lee M, Woo S, Lee S. Regularization strategy for point cloud via rigidly mixed sample. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021, 15895–15904
 12. Zhang J, Chen L, Ouyang B, Liu B, Zhu J, Chen Y, Meng Y, Wu D. PointCutMix: regularization strategy for point cloud classification. *Neurocomputing*, 2022, 505: 58–67
 13. Chen Y, Hu V T, Gavves E, Mensink T, Mettes P, Yang P, Snoek C G M. PointMixup: augmentation for point clouds. In: Proceedings of the 16th European Conference on Computer Vision. 2020, 330–345
 14. Ding Z, Han X, Niethammer M. VoteNet: a deep learning label fusion method for multi-atlas segmentation. In: Proceedings of the 22nd International Conference on Medical Image Computing and Computer-Assisted Intervention. 2019, 202–210
 15. He Y, Sun W, Huang H, Liu J, Fan H, Sun J. PVN3D: a deep point-wise 3D keypoints voting network for 6DoF pose estimation. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020, 11629–11638
 16. Li J, Chen B M, Lee G H. SO-Net: self-organizing network for point cloud analysis. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2018, 9397–9406
 17. Li Y, Bu R, Sun M, Wu W, Di X, Chen B. PointCNN: convolution on X -transformed points. In: Proceedings of the 32nd International Conference on Neural Information Processing Systems. 2018, 828–838
 18. Xu Y, Fan T, Xu M, Zeng L, Qiao Y. SpiderCNN: deep learning on point sets with parameterized convolutional filters. In: Proceedings of the 15th European Conference on Computer Vision. 2018, 90–105
 19. Liu Y, Fan B, Meng G, Lu J, Xiang S, Pan C. DensePoint: learning densely contextual representation for efficient point cloud processing. In: Proceedings of IEEE/CVF International Conference on Computer Vision. 2019, 5238–5247
 20. Wang C, Samari B, Siddiqi K. Local spectral graph convolution for point set feature learning. In: Proceedings of the 15th European Conference on Computer Vision. 2018, 56–71
 21. Shen Y, Feng C, Yang Y, Tian D. Mining point cloud local structures by kernel correlation and graph pooling. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2018, 4548–4557
 22. Liu J, Ni B, Li C, Yang J, Tian Q. Dynamic points agglomeration for hierarchical point sets learning. In: Proceedings of IEEE/CVF International Conference on Computer Vision. 2019, 7545–7554
 23. Su H, Jampani V, Sun D, Maji S, Kalogerakis E, Yang M H, Kautz J. SPLATNet: sparse lattice networks for point cloud processing. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2018, 2530–2539
 24. Wu W, Qi Z, Fuxin L. PointConv: deep convolutional networks on 3D point clouds. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019, 9613–9622
 25. Mao J, Wang X, Li H. Interpolated convolutional networks for 3D point cloud understanding. In: Proceedings of IEEE/CVF International Conference on Computer Vision. 2019, 1578–1587
 26. Xu M, Ding R, Zhao H, Qi X. PACConv: position adaptive convolution with dynamic kernel assembling on point clouds. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021, 3172–3181
 27. Wang H, Huang D, Wang Y. GridNet: efficiently learning deep hierarchical representation for 3D point cloud understanding. *Frontiers of Computer Science*, 2022, 16(1): 161301
 28. Xiang T, Zhang C, Song Y, Yu J, Cai W. Walk in the cloud: learning curves for point clouds shape analysis. In: Proceedings of IEEE/CVF International Conference on Computer Vision. 2021, 895–904
 29. Guo M H, Cai J X, Liu Z N, Mu T J, Martin R R, Hu S M. PCT: point cloud transformer. *Computational Visual Media*, 2021, 7(2): 187–199
 30. Zhao H, Jiang L, Jia J, Torr P, Koltun V. Point transformer. In: Proceedings of IEEE/CVF International Conference on Computer Vision. 2021, 16239–16248
 31. Liu S, Luo X, Fu K, Wang M, Song Z. A learnable self-supervised task for unsupervised domain adaptation on point cloud classification and segmentation. *Frontiers of Computer Science*, 2023, 17(6): 176708
 32. Xian Y, Xiao J, Wang Y. A fast registration algorithm of rock point cloud based on spherical projection and feature extraction. *Frontiers of Computer Science*, 2019, 13(1): 170–182
 33. Li H, Liu Y, Xiong S, Wang L. Pedestrian detection algorithm based on video sequences and laser point cloud. *Frontiers of Computer Science*, 2015, 9(3): 402–414
 34. Dabouei A, Soleymani S, Taherkhani F, Nasrabadi N M. SuperMix: supervising the mixing data augmentation. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021, 13789–13798
 35. Verma V, Lamb A, Beckham C, Najafi A, Mitliagkas I, Lopez-Paz D, Bengio Y. Manifold mixup: better representations by interpolating hidden states. In: Proceedings of the 36th International Conference on Machine Learning. 2019, 6438–6447
 36. Guo H, Mao Y, Zhang R. MixUp as locally linear out-of-manifold regularization. In: Proceedings of the 33rd AAAI Conference on Artificial Intelligence. 2019, 3714–3722
 37. Harris E, Marcu A, Painter M, Niranjana M, Prügel-Bennett A, Hare J. FMix: enhancing mixed sample data augmentation. 2020, arXiv preprint arXiv: 2002.12047
 38. Paszke A, Gross S, Chintala S, Chanan G, Yang E, DeVito Z, Lin Z, Desmaison A, Antiga L, Lerer A. Automatic differentiation in PyTorch. In: Proceedings of the 31st Conference on Neural Information Processing Systems. 2017
 39. Kingma D P, Ba J. Adam: a method for stochastic optimization. In: Proceedings of the 3rd International Conference on Learning Representations. 2015
 40. Loshchilov I, Hutter F. SGDR: stochastic gradient descent with warm restarts. In: Proceedings of the 5th International Conference on Learning Representations. 2017



Taiyan Chen received the BE degree in electronic information engineer from University of Science and Technology of China, China in 2020. He is currently pursuing the PhD degree in computer science and technology with Peking University, China. His research interests include point cloud and image processing.



Xianghua Ying received the PhD degree from the Institute of Automation, Chinese Academy of Sciences, China in 2004. He was a Visiting Professor at the University of Southern California, USA from September 2007 to August 2008. He is currently a Full Professor at the Key Laboratory of Machine Perception (Ministry of Education), School of Intelligence Science and Technology, Peking University, China. His major interests include 3D reconstruction, motion analysis, and computational photography.