2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

Long-Tailed 3D Semantic Segmentation with Adaptive Weight Constraint and Sampling

Jean Lahoud[1], Fahad Shahbaz Khan[1,2], Hisham Cholakkal[1], Rao Muhammad Anwer[1,3], Salman Khan[1,4]

***Abstract*— Existing 3D understanding datasets typically provide annotations for a limited number of object classes, with sufficient examples per class. However, real-world object classes are not equally represented in practical settings, leading to poor performance on rarely-occurring categories if the class imbalance is neglected. In this work, we address the challenge of 3D semantic segmentation with a long-tail distribution of classes. Common methods to reduce class imbalance during training include data re-sampling, loss re-weighting, and transfer learning. In contrast, our work proposes to effectively utilize network classifier weights in 3D models to balance the training on long-tail class distributions. While previous work in the 2D domain has studied imposing constraints on the classifier weights to regularize the training, it is sensitive to hyper-parameter choices and has not been yet explored for the 3D domain. To address these challenges, our work proposes adaptive regularization for frequent classes and sampling-based regularization for rare classes that alleviate the need to manually select thresholds and can dynamically focus training on the hard classes. Our experiments on the large-scale ScanNet200 benchmark show that our method achieves improved performance, surpassing methods that rely on re-sampling, re-weighting, and pre-training.**

I. Introduction

With the emergence of various depth sensing devices, such as LiDAR and RGB-D sensors, 3D perception has received increased attention. One common 3D task is semantic segmentation, in which each 3D element, *e.g*let@tokeneonedot, a point in a point cloud, is classified with an associated object class label. While 2D image understanding relies on color for understanding, 3D scene understanding makes use of shape information. The shape information represents the real 3D geometry of objects, which is not affected by lighting conditions or viewpoint as opposed to 2D images.

Numerous deep learning approaches have been proposed for 3D semantic segmentation, supported by the availability of various 3D datasets. Nevertheless, most of these datasets provide annotations for a limited number of classes. Since real-world scenarios commonly occur with high variability in the class instance occurrences, recent datasets have been introduced with a considerably higher number of classes [1]. The highly imbalanced data imposes a training challenge for deep learning methods. Fig. 1 (left) shows the high variability in the number of points among the classes of the ScanNet200 dataset. While naive training tends to appropriately learn to recognize frequent classes, recognizing rare classes remains a challenge.

[1] Mohamed bin Zayed University of AI, [2] Linköping University
[3] Aalto University, [4] Australian National University

Common approaches to address the long-tailed recognition task include balanced data sampling [2], [3], [4], loss re-weighting [5], [6], [7], and pre-training [8], [9]. In the balanced data sampling case, training data is re-sampled such that training occurs on uniformly distributed classes. For loss re-weighting, rare or hard examples have a higher contribution to the weight updates, to remedy the fewer times they invoke the weight update. For pre-training approaches, methods aim to extract general feature information that is beneficial in distinguishing all the classes.

Recent research in long-tailed recognition on 2D data has focused on achieving better training balance by examining the network classifier weights [10], [11], [12]. As the structure of the classifier network differs between balanced and imbalanced training scenarios, regularization methods have been explored to improve the classifiers when training data is imbalanced. These solutions include projecting weight norms onto an L2-norm sphere, choosing the classifiers as vertices of an equiangular tight frame, adding a weight decay into the loss, or imposing an upper bound on the L2-norm of the weights. They have been motivated by the high variability in the classifier norms after regular training in a long-tail setting, where frequent classes tend to have larger norms than rare ones. Fig. 1 (left) shows the norms of the weights of the last layer when trained using a standard 3D semantic segmentation technique [1] with loss balancing and instance re-sampling. The high variation motivates us to consider similar network weight regularization methods for 3D. Nonetheless, introducing such regularization requires careful hyperparameter selection, which significantly affects the training.

Motivated by the previous work that examines the network weight norms to balance the training in 2D, we propose a classifier weight-based regularization scheme for 3D. We point out two shortcomings in the existing 2D balanced training schemes: (1) thresholding of weight norms uses a hard threshold that requires a manual setting, and does not adapt to different training stages, and (2) weight norms do not contribute to choices that can be made during training. Therefore, we propose in this work two novel training strategies to circumvent that. In the first strategy, we propose an adaptive constraint on the weight norm threshold depending on the training evolution. In the second strategy, we introduce a 3D-specific data sampling based on the classifier norms to focus on the difficult cases during training.

Our proposed strategies aim at better balancing the training stage and do not affect the testing stage. These strategies complement each other, where one imposes a direct con-

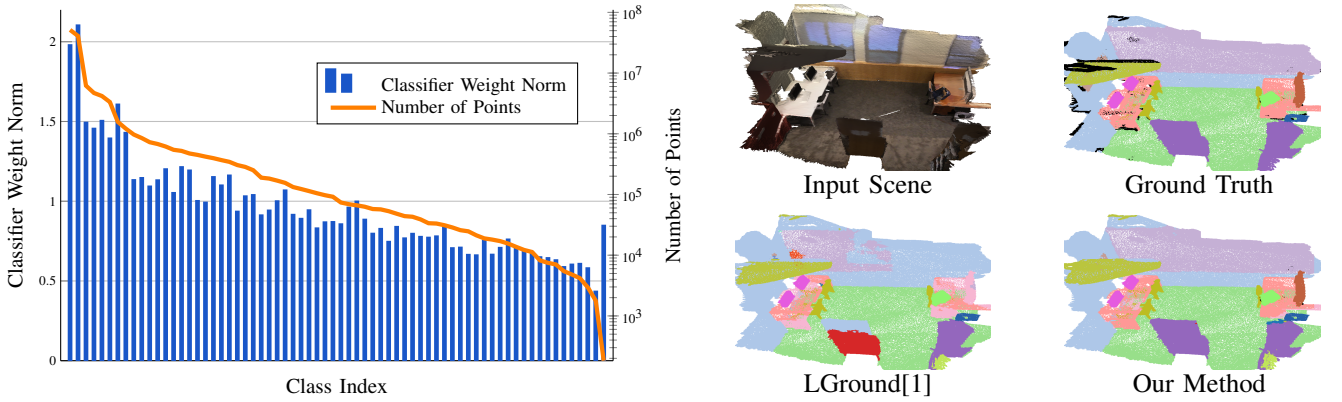

Fig. 1: (Left) The network weight norms of the last layer after using standard 3D semantic segmentation training [1] compared to the number of points for the corresponding classes of ScanNet200 dataset. The highly imbalanced training data poses a challenge to regular 3D semantic segmentation techniques. Even with loss balancing and instance re-sampling, the magnitude of the weight norms is highly variable. (Right) Our proposed method balances classifier weights through constraints and sampling, which leads to better semantic segmentation of rarely occurring object classes (room divider in dark purple) as well as more common objects (window in light purple).

straint on the weight norm of common and rare classes classifiers, while the other balances the classifiers through better sampling. This sampling makes use of the classifier weight norms to introduce diverse examples of rare classes, strengthening the classifier weights of rare and hard classes and dropping examples of common classes. We believe that classifier weight norms give more insights into the hardness of certain classes compared to balancing based on class instance occurrences.

Contributions: In summary, our main contributions are:

- We propose an adaptive range for limiting the classifier weight norms to balance training for long-tail classes.
- We propose to use the weight norms to improve the selection of data through 3D-specific sampling, which allows focusing on difficult classes that can be rare or common.
- Experiments on ScanNet200 benchmark test set show that our method improves baseline [1] performance by an absolute gain of 3.3 %, without relying on extra data.

II. Related Work

3D Semantic Segmentation. With the emergence of multiple large-scale point clouds datasets that provide per-point class annotation [13], [14], [15], [16], several approaches have been proposed to segment point clouds. One approach uses voxelization to transform a given point cloud into a 3D regular grid. While the straight-forward approach would be to directly utilize 3D convolutions on the 3D volumetric grid, this is computationally inefficient since it does not benefit from the sparse nature of point clouds. Therefore, several methods have proposed efficient implementation, in which convolution operations are applied to the locations in the volumetric grid that hold information [17], [18], [19]. Other methods are proposed to directly learn point features without transforming into an intermediate representation. In that direction, PointNet [20], [21] proposes to learn point set features through shared Multi-Layer Perceptrons and pooling to retain permutation invariance. Other 3D semantic segmentation methods have relied on graph convolutions [22], octree structures [23], or deformable convolutions [24].

Given the various 3D point cloud processing methods, numerous strategies have been proposed to improve 3D semantic segmentation performance. Some methods have explored the benefit of unsupervised pre-training for improving 3D understanding tasks [25], [26], especially with scenarios having limited reconstruction or annotations. Other methods relied on data augmentation to create new data samples [27]. Recently, CLIP text embeddings [28] have also been utilized for better 3D semantic pre-training [1].

Long-tailed Recognition. Given the imbalanced real-world distribution of object classes, training without considering this imbalance would lead to inferior results on rare classes compared to commonly occurring classes. Therefore, several techniques have been suggested to improve the performance of the less frequent classes (long-tail classes).

One proposed solution is to re-sample the classes to flatten the class distribution. While a naive approach would under-sample frequent classes and over-sample the rare classes, this solution is not effective: the under-sampling might lead to discarding important information, whereas the over-sampling would lead to over-fitting. Therefore, many methods have proposed re-sampling with better strategies [2], [29], [30], [3], [4]. Moreover, some methods re-sample in feature space instead of the input space [31], [32], [33]. Other methods propose to re-weigh the loss based on per-class measures [5], [34], [35], [36], [37], [6], [7], or assign different losses to different training examples [38], [39], [40]. Loss re-weighting allows all classes to have an adequate contribution to the parameters update during training.

Another approach for long-tailed recognition uses parameter regularization to improve generalizability and reduce overfitting [10]. In this approach, additional constraints are introduced on the network weights to regularize the training across different classes. This approach is relatively simple

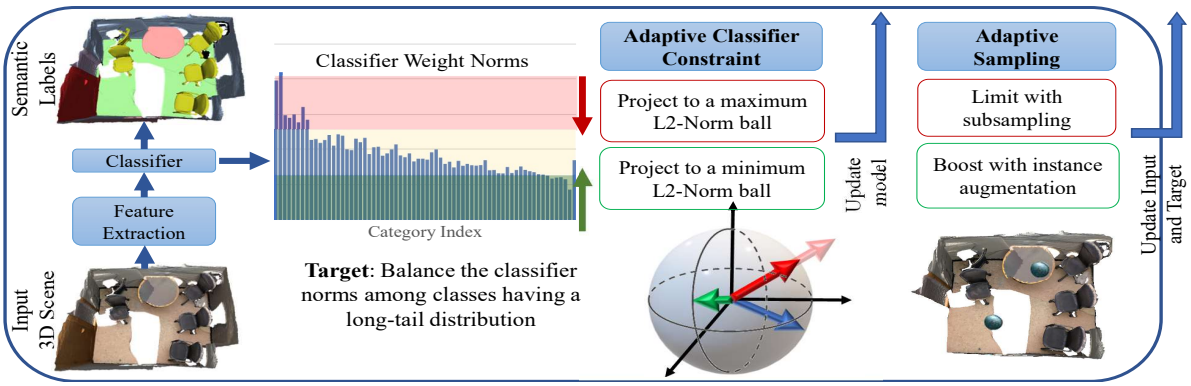

Fig. 2: Our proposed balanced training approach for long-tail 3D semantic segmentation. At each step, we perform two updates. The first update involves the network weights, where the weights of the last layer (classifier) are constrained to an L2-norm ball with an adaptive radius. The second update considers classes corresponding to low weight norm in the long-tail distribution, and augments scenes with instances of that class.

but requires proper parameter tuning to achieve competitive performance. Since the challenge in long-tail recognition originates from the lack of training data for rare classes, numerous methods propose to use transfer learning [8], [9], [41]. Transfer learning methods leverage information trained on frequent classes to improve the learning of the rare classes. Additionally, self-supervised approaches have been shown to benefit long-tail recognition problems by learning better representations [42], [43]. Other methods relied on contrastive learning [44], [45], [46] to improve the separation between attributes of different classes.

Supported by the availability of large-scale annotated datasets with long-tail class distribution [47], [48], [36], long-tailed recognition has been well studied in the image domain. In the 3D domain, ScanNet200 [16], [1] has been introduced for the task of long-tailed 3D semantic segmentation. While re-weighting, re-sampling, and transfer learning approaches have been explored in the 3D context [1], dedicated parameter regularization has received less attention. Recently, CeCo [49] proposes an auxiliary loss during training to encourage class feature centers to lie on an equiangular tight frame. Nevertheless, the proposed loss does not influence the data sampling and augmentation during training.

III. Method

Our proposed method includes two main balancing components (Fig. 2). In the first component, we target regularizing the weight norms of the classifier to dampen the dominance of the frequent classes classifiers on the weight updates and boost the norms of the rare classes classifiers. For the second component, we propose training data sampling through selective instance augmentation and subsampling.

Problem Formulation: Given an input scene represented by a point cloud set, our task is to label each point in the point cloud with one of the N semantic classes. Deep learning approaches focus on learning a semantic segmentation network parameterized by a set of weights at multiple layers. At the last layer, the labels of the points are generated using a classifier that takes as input a learned feature representation and outputs classification scores for each of the N classes. In the long-tailed recognition setting, the distribution of classes shows an imbalance among the frequency of occurrences of different object classes, which makes learning harder for rare examples. The level of imbalance is usually measured using the imbalance factor, which is the ratio of the number of training examples for the most frequent class to that of the least occurring class. An imbalance factor of 1 means that the dataset is perfectly balanced. The larger the imbalance factor gets, the higher the difference between the number of training examples per class.For semantic segmentation, the imbalance in the training examples has two origins: (1) some objects naturally occur more frequently than others, and (2) some objects are much larger than others. Common and large objects are therefore represented in a higher number of labeled points. For ScanNet200 indoor scene dataset, the imbalance factor is in the order of 10^5 which poses a challenge to improve learning on all the classes.

Motivation: Parameter regularization has been shown to be effective in improving performance in a long-tail setting. More specifically, we investigate the importance of the classifier weights at the last layer of the semantic segmentation network, motivated by previous work in the 2D domain [10], [11], [12]. Some of these works target placing the classifier vectors on vertices of a simplex equiangular tight frame [11], [12] to imitate training with balanced data. Other works target classifier norms only by fixing, thresholding, or introducing them into the loss [10]. While imposing a strict constraint on the classifier can be considered for a balanced prediction, our experiments performed by these methods have shown that introducing a tight/fixed constraint on the classifier vectors does not yield the best improvement. Additionally, parameter regularization is highly affected by the set of hyper-parameters used, where the improper setting of parameters might lead to a drastic decrease in performance. Moreover, classifier weights might differ between various strategies, and introducing additional loss or sample balancing would require re-setting the regularization parameters. Therefore, we propose to regularize the classifier vectors using a set of data-driven parameters, which would adapt to different training strategies and stages.

A. Adaptive Classifier Norm Constraint

A training with balanced class samples leads to classifier weights with similar L2-norms [10], [11], [12]. When training with highly imbalanced class distribution, the L2-norm of classifier weights of frequent classes tend to increase to higher values than rare classes. Our initial experiments show that even with approaches that apply loss re-weighting, data re-sampling, and transfer learning [1], the variance in classifier weights remains high (Fig. 1 left). Moreover, although the norm of the classifier weights is affected by the additional balancing, they are not proportional to the data statistics used for balancing.

Therefore, we investigate setting a constraint on the norm of the classifier weights at each training step for better training regularization. Let w_i be the vector containing the network weights of the last layer corresponding to the classifier of the ith category, where $i \in \{1,..,N\}$. At each training step, and following the loss back-propagation from a given batch of samples, the network weights at the last layer are updated as follows:

$$w_i = \begin{cases} \frac{\delta_l}{||w_i||_2} w_i & \text{if } ||w_i||_2 > \delta_l \\ w_i & \text{if } \delta_s \leq ||w_i||_2 \leq \delta_l \\ \frac{\delta_s}{||w_i||_2} w_i & \text{if } ||w_i||_2 < \delta_s \end{cases}$$

where δ_l is the radius of the L2-norm ball that caps all the weight norms and δ_s amplifies the small weight norms. Here, the limit imposes projecting network weights with high norm values onto an L2-norm ball with radius δ_l (large ball) and projects low norm values onto one with radius δ_s (small ball).

While the previous constraint on the weight norms improves training through regularization, it requires to manually set the threshold. This threshold requires prior knowledge of the expected weight norm value range.Moreover, using a fixed threshold throughout the training might yield excessive constraining, where all classes weights are limited, or it might not affect any class. A strict constraint is not desirable as it does not achieve good balancing among classes [10]. Additionally, using other long-tail balancing solutions, such as loss re-weighting or data balancing, would lead to variation in the classifier weight distribution. This variation would require parameter resetting to achieve similar behavior. A straightforward way to bypass setting a threshold value that can strike a balance in the training is to enforce a fixed number of weights to be normalized. For example, the k weight with the largest norms can be projected back into an L2-norm ball with radius set to the smallest norm among them. Nevertheless, this solution does not take into account the distribution of the weight norms across the classes. Therefore, we propose to use an adaptive threshold based on the mean and standard deviation of the weight norms:

$$\delta_l = \mu + \eta\rho \qquad \delta_s = \mu - \eta\rho$$

where $\mu = \frac{1}{N}\sum_i^N ||w||_2$ is the mean L2-norm of the weights over the classes and $\rho = \left(\sum_i^N (||w_i||_2 - \mu)^2/N\right)^{0.5}$ is the standard deviation of the norms.

Our proposed adaptive threshold limits the L2-norm weight of the classes that are high compared to weight norms of other classes. This allows the threshold to vary depending on the training evolution i.e once recurring training examples of rare examples lead to an increase in the model weight norms, the threshold is relaxed to allow more dynamic balancing of weight norms. Moreover, setting η value requires less time-consuming tuning as it follows normal distribution rules. Our proposed threshold is also easier to set with other sampling and loss criteria.

B. Adaptive Sampling to Balance Classifier Norms

Common sampling strategies make use of the class frequency for data balancing. This allows less frequent classes to have a sufficient contribution to the network weight updates. For 3D semantic segmentation, we believe that for well-balanced training, the sampling should not only consider the frequency of points for each class but also the frequency of object instances and their shape complexity. Nonetheless, a measure of complexity is not simple and depends not only on the data, but also on the models used. In the 3D domain, various backbone architectures are used for feature extraction, such as point-based, transformer-based, and voxel-based networks. These point cloud processing methods have different abilities in learning 3D shape information. Alternatively, we here propose to use the network weight norms as a precursor to the ability of a given network to classify a certain object class.

With the adaptive maximum weight norm we previously introduced, we impose an additional regularization on the network training. We here propose to use the weight norms for better data sampling strategies. Unlike the 2D domain, where images have a set resolution, point clouds occur in variable sizes. While some methods sample a given scene into a fixed input size, this approach does not take into consideration the spatial extents of a scene nor the required number of points to properly represent a scene. On the other hand, using the full scenes as input varies the memory allocation during training and requires reserving memory that can fit the largest scenes. Instead, we crop the scenes to a fixed maximum input size in terms of number of voxels. This allows processing more scenes simultaneously and reducing large scene. Instead of randomly sub-sampling the large point clouds, we propose to sample according to the classifier weight norms, i.e classes with corresponding high classifier norms are dropped from the training in favor of training with classes with low classifier norms. Therefore, the probability p_i of choosing a point belonging to class i is

$$p_i \propto (1 - ||w_i||_2).$$

Moreover, we consider classes with the lowest weight norms, and sample additional object instances. We use the augmentation technique from [1] to place the sampled instances into the training scenes. The sampled instances are added to plausible locations in the scenes of the next training iterations (see Fig. 2). Moreover, the added object

instances help in learning object-specific shape representation and reduce the reliance on the context information to recognize the rare classes. Given the difference between the class frequency and the weight norm obtained from regular training, our method does not solely add tail instances, but considers the classes that require improvement based on their corresponding classifier weight norm.

While previous methods re-sample the least occurring classes, it does not necessarily boost the weights of all tail classes. For example, Fig. 1 shows that for the class with the least number of sample points, the previously used class balancing technique puts high weight on it which yielded a high weight norm. Instead of re-sampling and augmenting instances for that class, our method targets other classes. Our target classes not only include objects from the rare set of classes but also objects from the common set.

Our adaptive sampling works side by side with the adaptive classifier norm constraint. The adaptive sampling encourages the hard classes to contribute higher to the weight updates and skips training on easy classes, whereas the adaptive classifier constraint regularizes the network parameters. Here, we distinguish between hard classes and non-recurring classes to prioritize training on the hard classes.

Implementation Details: We use LGround method [1] with the 3D sparse U-Net MinkUNet34 backbone [19] as our baseline framework. We also initialize our network parameters with LGround pretrained model. For the loss, we use a weighted focal loss in addition to Tversky loss [50]. We set η to 1, and sample 100K points from each scene using the proposed adaptive sampling.

IV. Experiments

In this section, we perform experiments on the ScanNet200 dataset [1]. We compare our method against three competitive approaches: (1) SupCon [44], (2) CSC [26], and (3) LGround [1]. We also perform an ablation study to analyze the effectiveness of each of our added modules.

A. Dataset

The ScanNet200 dataset [1] provides labels for 200 semantic classes on the ScanNet dataset [16]. It contains scans of indoor scenes that are reconstructed from multiple RGB-D viewpoints. We use the common training and testing split, with 1201 scenes for training and 312 for validation. Additionally, 100 unlabeled test scenes can be evaluated using the online benchmark. The 200 classes are split into 66 head, 68 common, and 66 tail categories based on the frequency of the number of points labels.

To evaluate the various models, we use the common intersection-over-union (IoU) metric. Per-class IoUs are obtained by averaging the IoU score of all the class examples. The mean IoU (mIoU) is the average of all class IoUs. Therefore, all classes have the same weight on the final mIoU, irrespective of the corresponding object sizes or frequencies. We also provide precision and recall values to assess whether there is variation between false positives or false negatives on the various head, common, and tail classes.

B. Comparison to Other Approaches

To validate our proposed method, we perform experiments on the ScanNet200 validation set. We compare it to methods using the same model architecture MinkUNet34 [19] as a backbone. The first method is MinkUNet34 trained from scratch using the LGround finetuning framework. We also compare to SupConv, which uses a contrastive loss to pull together the embeddings of points belonging to the same class. CSC and LGround provide different pretraining strategies, where CSC uses unsupervised pretraining with constrastive loss, and LGround uses CLIP text embeddings.

We show the quantitative comparison in Table I. The experiments show that our method outperforms all other methods in terms of mIoU for all classes.Compared to the baseline method, LGround, our method improves on the mIoU score of all classes by more than 3%. Looking into the classes of different frequencies, our proposed method yields good improvement in common and tail classes, which demonstrates the effectiveness of our proposed method when training with a dataset with a long-tail distribution. We note here that we use LGround pretrained weights for initialization, and our proposed approach can adapt to any pretraining strategy. The results validate that our proposed network weight regularization and sampling improve the 3D semantic segmentation accuracy. This improvement is also observed when considering classes with various frequencies.

In addition to the experiments on the ScanNet validation set, we also evaluate on the ScanNet benchmark test set. The results are provided in Table II. We compare to training the baseline Minkowski34D [19] model from scratch. We also compare to two other methods that rely on pretraining, CSC-Pretrain [26] and LGround [13]. With our long-tail tailored solution, our method was able to achieve the best performance in terms of mIoU on all the classes, surpassing the previous best-performing method by more than 3%. Our proposed classifier constraint and sampling, which are tailored to the long-tail dataset problem, have prompted good improvement in common and tail classes.

C. Ablation Study

We perform a set of ablation experiments to better understand the effect of each part of our proposed method. For the baseline, we use the LGround method with two modifications: we add Tversky loss to the balanced focal loss, and perform random down-sampling to a fixed number of occupied voxels. Down-sampling the input data allows us to experiment with a similar batch size among all the experiments. Since our proposed method relies on network weight regularization, we explore previous similar solutions [10]. We first compare to the MaxNorm constraint, in which the weight norms constraint is set using a pre-defined threshold. While this method improves upon the baseline LGround method, it requires testing a range of thresholds to achieve the desired performance. On the other hand, our proposed method's threshold parameter is adjusted using the model information. The effectiveness of our adaptive weight norm threshold can be observed in the mIoU performance.

	mIoU				Precision				Recall			
	All	Head	Common	Tail	All	Head	Common	Tail	All	Head	Common	Tail
MinkUNet34 [19]	25.02	48.29	19.08	7.86	58.32	68.81	66.29	39.88	33.67	60.45	25.50	15.06
SupCon [44]	26.02	48.55	19.17	10.34	58.52	69.52	65.42	40.62	35.23	60.27	26.28	19.14
CSC [26]	26.41	49.43	19.52	10.28	59.51	70.00	67.75	40.78	34.79	61.01	25.75	17.62
LGround [1]	28.87	51.51	22.68	12.41	65.90	**72.72**	66.69	58.30	39.40	62.50	29.09	26.61
Ours	**32.04**	**51.53**	**26.35**	**18.43**	**67.33**	71.74	**69.58**	**60.67**	**41.67**	**63.25**	**34.31**	**29.26**

TABLE I: Quantitative comparison on ScanNet200 dataset. Our method achieves improved performance, surpassing all previous methods methods in terms of mIoU. Our proposed regularization and re-sampling improve the performance not only on the tail classes, but also on the common and head classes.

	All	Head	Common	Tail
Minkowski34D [19]	25.3	46.3	15.4	10.2
CSC-Pretrain [26]	24.9	45.5	17.1	7.9
LGround [1]	27.2	48.5	18.4	10.6
Ours	**30.5**	**50.8**	**22.5**	**14.2**

TABLE II: 3D semantic segmentation performance on the ScanNet200 benchmark test set. We report the mIoU metric averaged over all 200 classes, as well as mIoU for the head, common, and tail classes.

	All	Head	Common	Tail
Baseline	28.97	52.10	23.20	11.79
MaxNorm [10]	29.87	51.93	23.79	14.07
L2-Norm [10]	30.05	52.33	23.53	14.48
Classifier Constraint	31.34	52.32	24.74	17.18
Adaptive Sampling	30.18	51.77	24.69	14.25
Ours	32.04	51.53	26.35	18.43

TABLE III: Ablation results on ScanNet200 validation set.

Another regularization approach is the L2-normalization, which constrains all class weights to an L2-norm sphere. While this strategy ensures similar network weights for all the classes, it imposes a strict artificial balance among all classes. Nevertheless, this approach improves the baseline LGround method, and unlike previous experiments in the 2D domain [10], it achieves better performance compared to MaxNorm. This result suggests that the network weight norm regularization is recommended not only for high norm values but also for regularizing weights with lower norm values.

We also study the effect of adding the classifier norm constraint alone without the adaptive sampling. Results show that our proposed constraint boosted the 3D semantic segmentation mIoU by 2.37 %. We also test our method with the adaptive sampling approach, without constraining the norms of the classifier. Our proposed sampling approach yields an improved mIoU of 1.21 %. Overall, our proposed method, which combines the classifier constraint and the adaptive sampling, achieves the best performance. This shows the complementarity between our two proposed modules.

D. Generalizability to the Image Domain

Although our framework is mainly designed for 3D long-tail distributions, we also test our adaptive classifier weight constraint method on the CIFAR100 image classification dataset [51] with long-tail distribution and imbalance factor of 100 [34]. Note that we only test the classifier constraint here since our proposed combination with sampling is specific to 3D point clouds only. We show the results in Table IV and report the top-1 accuracy for classes the occur frequently (many), less frequently (medium), and rare times (few). We perform our experiments using a ResNet32 architecture

	All	Many	Medium	Few
Baseline (w/ CB Loss)	47.5	77.5	47.0	13.1
Baseline + max	50.9	76.6	51.7	19.9
Baseline + ours	51.3	76.6	51.3	21.9

TABLE IV: Image classification results on the CIFAR100 dataset with imbalance factor of 100 [34] (Top-1 accuracy). We use a baseline with class-balanced (CB) loss and compare to the maximum norm constraint (max) from [10]. Our method yields an absolute 2% increase in the accuracy of 'few' classes compared to the max constraint.

[52] baseline with a class balancing loss. We compare our weight constraint method to the maximum norm constraint (MaxNorm) from [10]. As can be seen, our proposed weight constraint has a better improvement in the accuracy of the rarely occurring classes (few), with an absolute gain of 2% over the maximum norm constraint.

We also test using the weight decay loss in the training stage of CIFAR100 image classification with ResNet32 backbone. The weight decay reduces the classifier norms, which improves the accuracy of the rarely occurring classes and yields an accuracy of 53.0%. Adding a maximum constraint to the weight decay loss has no effect, mainly due to having all classifier norms below the maximum norm. On the other hand, our proposed classifier weight constraint still improves on the top-1 accuracy, and achieves the best overall accuracy of 53.5%. We also notice that the improvement achieved by the weight decay loss on the rare classes was at the expense of the common classes. Our proposed method presents a better balancing between classes with different frequencies and validates its generalizability.

V. Conclusion

We present a method for long-tailed 3D semantic segmentation. We propose to impose a constraint on the network weights using an adaptive threshold. We show that this constraint improves the network regularization and leads to a better balanced training. Additionally, we present a selection criteria for better class sample augmentation based on the classifier weight norms. Experiments on the long-tail ScanNet200 dataset show the merits of our method.

Acknowlegment: The computations were enabled by resources provided by the National Academic Infrastructure for Supercomputing in Sweden (NAISS) at Alvis partially funded by the Swedish Research Council through grant agreement no. 2022-06725, the LUMI supercomputer hosted by CSC (Finland) and the LUMI consortium, and by the Berzelius resource provided by the Knut and Alice Wallenberg Foundation at the National Supercomputer Centre.

REFERENCES

[1] D. Rozenberszki, O. Litany, and A. Dai, "Language-grounded indoor 3d semantic segmentation in the wild," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022.

[2] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

[3] A. Gupta, P. Dollar, and R. Girshick, "Lvis: A dataset for large vocabulary instance segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 5356–5364.

[4] S. Park, Y. Hong, B. Heo, S. Yun, and J. Y. Choi, "The majority can help the minority: Context-rich minority oversampling for long-tailed classification," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 6887–6896.

[5] S. H. Khan, M. Hayat, M. Bennamoun, F. A. Sohel, and R. Togneri, "Cost-sensitive learning of deep feature representations from imbalanced data," *IEEE transactions on neural networks and learning systems*, vol. 29, no. 8, pp. 3573–3587, 2017.

[6] Y. Hong, S. Han, K. Choi, S. Seo, B. Kim, and B. Chang, "Disentangling label distribution for long-tailed visual recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 6626–6636.

[7] Y.-Y. He, P. Zhang, X.-S. Wei, X. Zhang, and J. Sun, "Relieving long-tailed instance segmentation via pairwise class balance," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 7000–7009.

[8] X. Yin, X. Yu, K. Sohn, X. Liu, and M. Chandraker, "Feature transfer learning for face recognition with under-represented data," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 5704–5713.

[9] B. Zhou, Q. Cui, X.-S. Wei, and Z.-M. Chen, "Bbn: Bilateral-branch network with cumulative learning for long-tailed visual recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 9719–9728.

[10] S. Alshammari, Y.-X. Wang, D. Ramanan, and S. Kong, "Long-tailed recognition via weight balancing," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 6897–6907.

[11] Y. Yang, S. Chen, X. Li, L. Xie, Z. Lin, and D. Tao, "Inducing neural collapse in imbalanced learning: Do we really need a learnable classifier at the end of deep neural network?" in *Advances in Neural Information Processing Systems*, 2022.

[12] Z. Zhu, T. Ding, J. Zhou, X. Li, C. You, J. Sulam, and Q. Qu, "A geometric analysis of neural collapse with unconstrained features," *Advances in Neural Information Processing Systems*, vol. 34, pp. 29 820–29 834, 2021.

[13] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from rgbd images," in *European conference on computer vision*. Springer, 2012, pp. 746–760.

[14] J. Behley, M. Garbade, A. Milioto, J. Quenzel, S. Behnke, C. Stachniss, and J. Gall, "Semantickitti: A dataset for semantic scene understanding of lidar sequences," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 9297–9307.

[15] I. Armeni, O. Sener, A. R. Zamir, H. Jiang, I. Brilakis, M. Fischer, and S. Savarese, "3d semantic parsing of large-scale indoor spaces," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1534–1543.

[16] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner, "Scannet: Richly-annotated 3d reconstructions of indoor scenes," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 5828–5839.

[17] P.-S. Wang, Y. Liu, Y.-X. Guo, C.-Y. Sun, and X. Tong, "O-cnn: Octree-based convolutional neural networks for 3d shape analysis," *ACM Transactions On Graphics (TOG)*, vol. 36, no. 4, pp. 1–11, 2017.

[18] B. Graham, M. Engelcke, and L. Van Der Maaten, "3d semantic segmentation with submanifold sparse convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 9224–9232.

[19] C. Choy, J. Gwak, and S. Savarese, "4d spatio-temporal convnets: Minkowski convolutional neural networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3075–3084.

[20] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 652–660.

[21] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," *Advances in neural information processing systems*, vol. 30, 2017.

[22] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph cnn for learning on point clouds," *Acm Transactions On Graphics (tog)*, vol. 38, no. 5, pp. 1–12, 2019.

[23] G. Riegler, A. Osman Ulusoy, and A. Geiger, "Octnet: Learning deep 3d representations at high resolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 3577–3586.

[24] H. Thomas, C. R. Qi, J.-E. Deschaud, B. Marcotegui, F. Goulette, and L. J. Guibas, "Kpconv: Flexible and deformable convolution for point clouds," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 6411–6420.

[25] S. Xie, J. Gu, D. Guo, C. R. Qi, L. Guibas, and O. Litany, "Pointcontrast: Unsupervised pre-training for 3d point cloud understanding," in *European conference on computer vision*. Springer, 2020, pp. 574–591.

[26] J. Hou, B. Graham, M. Nießner, and S. Xie, "Exploring data-efficient 3d scene understanding with contrastive scene contexts," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 15 587–15 597.

[27] A. Nekrasov, J. Schult, O. Litany, B. Leibe, and F. Engelmann, "Mix3d: Out-of-context data augmentation for 3d scenes," in *2021 International Conference on 3D Vision (3DV)*. IEEE, 2021, pp. 116–125.

[28] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, *et al.*, "Learning transferable visual models from natural language supervision," in *International Conference on Machine Learning*. PMLR, 2021, pp. 8748–8763.

[29] L. Shen, Z. Lin, and Q. Huang, "Relay backpropagation for effective learning of deep convolutional neural networks," in *European conference on computer vision*. Springer, 2016, pp. 467–482.

[30] D. Mahajan, R. Girshick, V. Ramanathan, K. He, M. Paluri, Y. Li, A. Bharambe, and L. Van Der Maaten, "Exploring the limits of weakly supervised pretraining," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 181–196.

[31] S. Ando and C. Y. Huang, "Deep over-sampling framework for classifying imbalanced data," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2017, pp. 770–785.

[32] P. Chu, X. Bian, S. Liu, and H. Ling, "Feature space augmentation for long-tailed data," in *European Conference on Computer Vision*. Springer, 2020, pp. 694–710.

[33] S. Li, K. Gong, C. H. Liu, Y. Wang, F. Qiao, and X. Cheng, "Metasaug: Meta semantic augmentation for long-tailed visual recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 5212–5221.

[34] K. Cao, C. Wei, A. Gaidon, N. Arechiga, and T. Ma, "Learning imbalanced datasets with label-distribution-aware margin loss," *Advances in neural information processing systems*, vol. 32, 2019.

[35] S. Khan, M. Hayat, S. W. Zamir, J. Shen, and L. Shao, "Striking the right balance with uncertainty," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 103–112.

[36] Y. Cui, M. Jia, T.-Y. Lin, Y. Song, and S. Belongie, "Class-balanced loss based on effective number of samples," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 9268–9277.

[37] T. Wang, Y. Zhu, C. Zhao, W. Zeng, J. Wang, and M. Tang, "Adaptive class suppression loss for long-tail object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 3103–3112.

[38] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

[39] M. Ren, W. Zeng, B. Yang, and R. Urtasun, "Learning to reweight examples for robust deep learning," in *International conference on machine learning*. PMLR, 2018, pp. 4334–4343.

[40] J. Shu, Q. Xie, L. Yi, Q. Zhao, S. Zhou, Z. Xu, and D. Meng, "Meta-

weight-net: Learning an explicit mapping for sample weighting," *Advances in neural information processing systems*, vol. 32, 2019.
[41] B. Kang, S. Xie, M. Rohrbach, Z. Yan, A. Gordo, J. Feng, and Y. Kalantidis, "Decoupling representation and classifier for long-tailed recognition," *arXiv preprint arXiv:1910.09217*, 2019.
[42] Y. Yang and Z. Xu, "Rethinking the value of labels for improving class-imbalanced learning," *Advances in neural information processing systems*, vol. 33, pp. 19 290–19 301, 2020.
[43] T. Li, L. Wang, and G. Wu, "Self supervision to distillation for long-tailed visual recognition," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 630–639.
[44] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, "Supervised contrastive learning," *Advances in Neural Information Processing Systems*, vol. 33, pp. 18 661–18 673, 2020.
[45] T. Li, P. Cao, Y. Yuan, L. Fan, Y. Yang, R. S. Feris, P. Indyk, and D. Katabi, "Targeted supervised contrastive learning for long-tailed recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 6918–6928.
[46] J. Zhu, Z. Wang, J. Chen, Y.-P. P. Chen, and Y.-G. Jiang, "Balanced contrastive learning for long-tailed visual recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 6908–6917.
[47] G. Van Horn, O. Mac Aodha, Y. Song, Y. Cui, C. Sun, A. Shepard, H. Adam, P. Perona, and S. Belongie, "The inaturalist species classification and detection dataset," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8769–8778.
[48] Z. Liu, Z. Miao, X. Zhan, J. Wang, B. Gong, and S. X. Yu, "Large-scale long-tailed recognition in an open world," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2537–2546.
[49] Z. Zhong, J. Cui, Y. Yang, X. Wu, X. Qi, X. Zhang, and J. Jia, "Understanding imbalanced semantic segmentation through neural collapse," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 19 550–19 560.
[50] S. S. M. Salehi, D. Erdogmus, and A. Gholipour, "Tversky loss function for image segmentation using 3d fully convolutional deep networks," in *Machine Learning in Medical Imaging: 8th International Workshop, MLMI 2017, Held in Conjunction with MICCAI 2017, Quebec City, QC, Canada, September 10, 2017, Proceedings 8*. Springer, 2017, pp. 379–387.
[51] A. Krizhevsky *et al.*, "Learning multiple layers of features from tiny images," 2009.
[52] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.