Gradient-Aware for Class-Imbalanced Semi-supervised Medical Image Segmentation

Wenbo Qi[1], Jiafei Wu[2](✉), and S. C. Chan[1](✉)

[1] Department of Electrical and Electronic Engineering, The University of Hong Kong, Pok Fu Lam, Hong Kong
qiwb@connect.hku.hk, scchan@eee.hku.hk

[2] Zhejiang Lab, Hangzhou, China
wujiafei@zhejianglab.com

Abstract. Class imbalance poses a significant challenge in semi-supervised medical image segmentation (SSLMIS). Existing techniques face problems such as poor performance on tail classes, instability, and slow convergence speed. We propose a novel Gradient-Aware (GA) method, structured on a clear paradigm: identify extrinsic data-bias → analyze intrinsic gradient-bias → propose solutions, to address this issue. Through theoretical analysis, we identify the intrinsic gradient bias instigated by extrinsic data bias in class-imbalanced SSLMIS. To combat this, we propose a GA loss, featuring GADice loss, which leverages a probability-aware gradient for absent classes, and GACE, designed to alleviate gradient bias through class equilibrium and dynamic weight equilibrium. Our proposed method is plug-and-play, simple yet very effective and robust, exhibiting a fast convergence speed. Comprehensive experiments on three public datasets (CT&MRI, 2D&3D) demonstrate our method's superior performance, significantly outperforming other SOTA of SSLMIS and class-imbalanced designs (*e.g.* + 17.90% with CPS on 20% labeled Synapse). Code is available at https://github.com/cicailalala/GALoss.

Keywords: Medical image segmentation · class imbalance · semi-supervised

1 Introduction

Accurate and efficient segmentation of medical images is essential in clinical applications such as computer-aided diagnosis and treatment planning [31,37,48]. However, labeling substantial data is time-consuming and labor-intensive, necessitating specialized expertise and often involving subjective interpretation. To address this challenge, semi-supervised learning (SSL) techniques have emerged as a promising approach in medical image segmentation

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-73001-6_27.

A. Leonardis et al. (Eds.): ECCV 2024, LNCS 15113, pp. 473–490, 2025.
https://doi.org/10.1007/978-3-031-73001-6_27

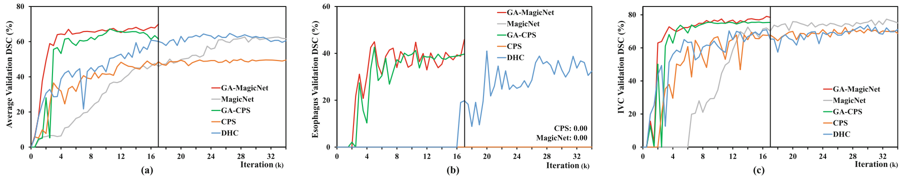

Fig. 1. Validation DSC (%) of MagicNet [7], CPS [9] and DHC [39] trained for 34k iterations, and applying proposed GA loss trained for 17k, on 20% labeled Synapse dataset [22]. Significant improvements are demonstrated. The convergence speed is much faster after applying GA. **(a):** average DSC of 13 organs. **(b):** especially tail class (esophagus). **(c):** normal class (inferior vena cava). (Best viewed zoom in)

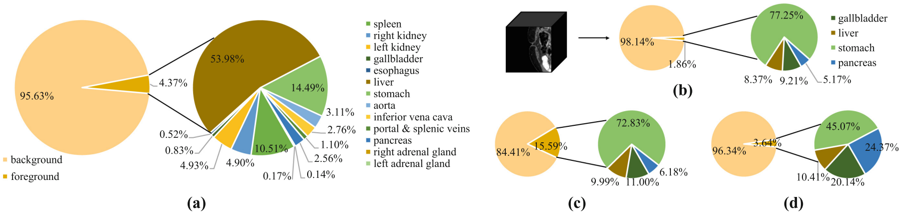

Fig. 2. (a) Classes distribution in Synapse dataset [22]. (b) Classes distribution of a randomly cropped patch. (c)–(d) Class distribution of the patch in (b) at 1k, 5k iterations are partially balanced after class equilibrium via hard instance mining.

[7,32,45,47,48]. SSL approaches have shown great potential to improve segmentation performance and generalization capabilities by leveraging limited labeled data and a larger pool of unlabeled data. Existing SSL medical image segmentation methods leverage self-training methods through the generation of pseudo-labels [10,23] or consistency regularization [38,48]. Nonetheless, the applicability of these strategies to class-imbalanced tasks encounters problems such as instability, decelerated convergence, and suboptimal performance on tail classes, as shown in Fig. 1. Class imbalance constitutes a notable challenge in semi-supervised medical image segmentation (SSLMIS). As shown in Fig. 2(a), classes are imbalanced between the background and foreground, as well as within the foreground. For instance, the foreground voxels account for a mere 4.37% of the total dataset, with the right adrenal gland representing just 0.14% of these foreground elements, highlighting the acute nature of this imbalance.

Recently, several techniques have been proposed to mitigate class imbalance on SSLMIS. Basak *et al.* [3] introduced a class-wise sampling strategy by keeping track of category-wise confidence during training. CLD [25] adjusted the loss by weighting classes according to the count of instances (pixels/voxels) within the dataset. DHC [39] further combined the class distribution and learning difficulty to address this issue. However, these methods relying on the dataset class distribution [25,39] ignored a crucial problem, that there exists a **distribution mismatch** between the dataset and input patches after sampling. As depicted in Fig. 2(a)–(b), the class distribution of the patches differs from that of the dataset. For instance, the liver constitutes 53.98% of the dataset. However, it accounts for only 8.37% of the patch in Fig. 2(b). Additionally, the distribution within patches also varies, and some classes even disappear after sampling. Moreover, a 3D patch contains a large number of voxels, approximately one million in a commonly used $96 \times 96 \times 96$ patch. During the training process, it's necessary to calculate the loss for each voxel and back-propagate the gradients. We refer to these phenomena: class absence in sampled patches, enormous instances, and imbalanced distribution, as **extrinsic data-bias** in class-imbalanced SSLMIS. These data-bias result in imbalanced gradients during back-propagation, which we named **intrinsic gradient-bias** (will be theoretically analyzed in Sect. 3.1).

Loss considerably influences the training efficacy of neural networks. Dice loss and Cross-Entropy (CE) [36] are widely used in SSLMIS [9,25,38,48]. Efforts to mitigate class imbalance through loss modifications have embraced strategies such as prioritizing difficult examples [24,46] and penalizing the majority classes in CE [33] or Dice loss [35]. As mentioned earlier, both CLD and DHC incorporated the weighting class strategies. However, counting instances either before or during the training to weight the majority class is inconvenient, and the counted weights are inconsistent with the patch distribution. Tversky loss emphasizes false negatives in Dice loss to achieve a better trade-off between precision and recall [1]. Nevertheless, these general methods did not theoretically address the limitations of loss in class-imbalanced SSLMIS caused by extrinsic data-bias.

To this end, we propose a novel **G**radient-**A**ware (GA) method that follows the paradigm: *identify extrinsic data-bias* → *analyze intrinsic gradient-bias* → *propose solutions*, to address the class imbalance problem in SSLMIS. Specifically, (1) our investigation reveals the presence of the extrinsic data-bias, as mentioned above. (2) We uncover the intrinsic gradient-bias through theoretical analysis, which consists of the inconsistent gradients for non-target classes of Dice loss, diminutive gradient, and the mismatch between the batch class distribution and weight of Cross Entropy, all of which stem from the extrinsic data-bias. (3) We propose a GA loss to address these issues. Our Gradient-Aware Dice (GADice) loss incorporates a probability-aware gradient for absent classes to Dice loss. Our Gradient-Aware Cross Entropy (GACE) alleviates the gradient-bias by achieving class equilibrium through hard instance mining, and dynamic weight equilibrium between the batch's volume weight and class weight in CE. The key contributions can be summarized as follows:

- Our research identifies the intrinsic problems of class-imbalanced SSLMIS as the inconsistent gradients for non-target classes, gradient-bias of diminutive gradient, and the mismatch between batch's class distribution and gradient weight. To the best of our knowledge, we are the first to identify these intrinsic gradient-bias through theoretical analysis.
- We propose a novel GA loss to alleviate the intrinsic gradient-bias by incorporating a probability-aware gradient for the absent classes, achieving class equilibrium via hard instance mining, and dynamic weight equilibrium between the batch's volume weight and class weight.
- Our proposed method is **plug-and-play, simple yet very effective and robust**, exhibiting a **fast convergence speed**.

- We verify our GA loss in three public multi-class datasets with different modalities and dimensions: Synapse [22], AMOS [16] and ACDC [4]. Extensive experiments underscore our GA loss's superiority, outperforming existing eight state-of-the-art semi-supervised methods by a large margin, with 19.56% on UA-MT [48] and 17.90% on CPS [9] (20% labeled Synapse). Our GA loss also surpasses seven state-of-the-art class-imbalanced designs, particularly noticeable in the segmentation of tail classes.

2 Related Work

2.1 Semi-supervised Medical Image Segmentation

Various approaches for SSLMIS have been proposed to address the limited availability of labeled data in the medical domain. Consistency regularization has emerged as a popular method in semi-supervised learning, with MT [38] representing a typical approach consisting of a teacher network that updates parameters using EMA and a student network that updates parameters using gradient propagation. UA-MT [48] incorporates the uncertainty information and transformation consistency to improve segmentation performance. Starting from prior anatomy, MagicNet [7] utilizes a data augmentation strategy via a magic-cube partition and recovery to regularize the consistent training. Pseudo-labeling is another widely used method that involves model training on labeled data followed by the generation of pseudo-labels on an unlabeled dataset. Several strategies have been proposed to generate reliable pseudo-labels, including uncertainty estimation [32], selecting high-confidence unlabeled samples [23], and utilizing two subnets to generate pseudo-labels [10]. Additionally, some methods based on contrastive learning [45,47] aim to minimize the similarity between views of negative pairs and maximize the similarity between augmented views of the positive pairs. However, most of these SSL methods did not consider the issue of class imbalance with barely labeled data.

2.2 Loss in Medical Image Segmentation

In the realm of medical image segmentation, loss functions can be roughly grouped into four categories [27]. (1) Distribution-based loss functions [24,33,46], which aim to minimize dissimilarity between two distributions. These functions, derived from the Cross Entropy, include variants that penalize majority classes based on class frequency [33] or prioritize hard examples [24,46]. (2) Region-based loss functions [11,12,28,34,35], which focus on maximizing the overlap between the predicted segmentation and the ground truth. The Dice loss [11,28] is a fundamental and representative function belonging to this category. (3) Boundary-based loss [17,18] that attends to minimize the distance between predicted segmentation and ground truth. (4) Compound loss functions [36,43,49] which are the weighted combination of the aforementioned loss. In particular, the combination of CE and Dice loss [36] is widely used in segmentation, especially for SSLMIS [7,9,25,38,39,48]. Although these loss functions are plug-and-play and some have been improved to address class imbalance, they are inadequate for SSL as they lack a systematic and comprehensive analysis from a gradient perspective. Hoel *et al.* [19] analyzed dice loss from gradient, but this analysis is not comprehensive for class-imbalanced SSLMIS.

2.3 Class Imbanlance

Class imbalance is a common and challenging problem in many applications [6,15,21,29]. While many SSL methods have been proposed for medical image segmentation, only a few have been specifically designed to address class imbalance. Basak *et al.* [3] proposed a class-wise sampling strategy and a fuzzy fusion based confidence array to record class-wise performance during training, effectively addressing learning bias. CLD [25] introduced a class-aware weighted loss and a probability-aware cropping approach to handle data bias. DHC [39] further improved performance by simultaneously weighting the distribution and difficulty. A&D [40] developed a generic framework that utilizes a Diffusion [14] encoder for aggregating and three decoders to decouple labeled and unlabeled data. These methods have advanced the effectiveness of class-imbalanced semi-supervised segmentation. However, they ignored the distribution mismatch of the dataset and sampled patches. Furthermore, sampling strategy for specific task and calculating distribution either before or during training to penalize the majority class are inconvenient, not plug-and-play for other tasks.

3 Method

The loss is usually calculated for a mini-batch, we define the input volume batch with B patches of 3D medical images as $\mathbf{X} \in \mathbb{R}^{B\times K\times W\times H\times D}$ and the ground truth annotation as $\hat{\mathbf{Y}} \in \{0, 1, ..., C\}^{B\times W\times H\times D}$, where K is the input channel and $W \times H \times D$ is the patch size. The model discerns a total of $C + 1$ distinct classes, encompassing one background class (0) and C foreground classes. The objective of SSL medical image segmentation is to predict the semantic label for each instance, which is obtained by the Argmax of the segmentation probability map $\mathbf{P}^{B\times(C+1)\times W\times H\times D}$. We define N as the total number of instances of the segmentation output for the mini-batch, where $N = B\times W\times H\times D$. Then, the one-hot ground truth can be formulated as $\mathbf{Y} = \{y_{i,c}|y_{i,c} \in \{0, 1\}\}^{B\times(C+1)\times W\times H\times D}$, where $y_{i,c}$ is the ground truth binary indicator of class c for the i–th instance, and $p_{i,c} \in \mathbf{P}$ is the corresponding segmentation probability. $p_{i,c} \in (0, 1)$, can infinitely approach but cannot be equal to 0 or 1, as it's the output of Softmax.

3.1 Gradient-Bias in Class-Imbalanced SSLMIS

As mentioned in Introduction Sect. 1, the extrinsic data-bias in class-imbalanced SSLMIS are class absence in sampled patches, enormous instances, and imbalanced distribution. In this section, we will theoretically analyze the limitations of Dice loss and CE from the gradient caused by extrinsic data-bias.

Dice Loss. Before analyzing the Dice loss, it is important to comprehend the characteristics of class absence in the segmentation probability map. During the training process, the neural network generates a $C+1$ dimensional probability vector for each instance. Even if no voxels of class c are randomly cropped into the batch, the neural network still generates a sub probability map $\mathbf{P}_c$ for the batch, where all corresponding one-hot labels are 0.

The Dice loss enables direct optimization of the Dice Similarity Coefficient (DSC), the most widely used segmentation evaluation metric. Previous studies have demonstrated that treating the batch as a whole to calculate Dice loss during training is helpful in improving the performance [7,20]. A multi-class Dice loss of a batch is defined by

$$\mathcal{L}_{\text{Dice}} = \frac{1}{C+1}\sum_{c=0}^{C}\left(1-\frac{2I_c}{U_c}\right) \tag{1}$$

where $I_c = \sum_{i=1}^{N} p_{i,c}y_{i,c}$, $U_c = \sum_{i=1}^{N} p_{i,c}^2 + \sum_{i=1}^{N} y_{i,c}^2$.

According to the definition, multi-class Dice loss is the average dice loss of each class. It should be noted that the loss is calculated on all classes, regardless of whether these classes exist in the input batch or not. However, as we have previously analyzed, class absence may exist in the batch. For a class c that is not sampled in the batch, $I_c = \sum_{i=1}^{N} p_{i,c}y_{i,c} = 0$ as all $y_{i,c}$ are 0. Look at Eq. (1), the Dice loss for this class is always 1, even if $p_{i,c}$ in sub probability map $\mathbf{P}_c$ are all correctly predicted (which means **no** voxel is predicted as this class). This is obviously unreasonable. This inconsistency arises from the original of the Dice Similarity Coefficient, which does not evaluate the true negative [1].

Then, take a look at the gradient of dice loss. As $\frac{\partial \mathcal{L}_{I_c}}{\partial p_{i,c}} = y_{i,c}$, and $\frac{\partial \mathcal{L}_{U_c}}{\partial p_{i,c}} = 2p_{i,c}$, the two-valued gradient of Dice loss can be formulated as,

$$\frac{\partial \mathcal{L}_{\text{Dice}}}{\partial p_{i,c}} = \begin{cases} -2\frac{U_c - 2p_{i,c}I_c}{U_c^2} & \text{if } y_{i,c} = 1, \\ 4\frac{p_{i,c}I_c}{U_c^2} & \text{otherwise.} \end{cases} \tag{2}$$

The gradient vector of the *i–th* voxel can be written as,

$$\frac{\partial \mathcal{L}_{\text{Dice}}}{\partial p_i} = \left[4\frac{p_{i,0}I_0}{U_0^2}, \cdots, -2\frac{U_c - 2p_{i,c}I_c}{U_c^2}, \cdots, 4\frac{p_{i,C}I_C}{U_C^2}\right]^T \tag{3}$$

We have $\frac{U_c - 2p_{i,c}I_c}{U_c^2} \geq 0$, as $U_c - 2p_{i,c}I_c \geq U_c - 2I_c = \sum_{i=1}^{N} p_{i,c}^2 + \sum_{i=1}^{N} y_{i,c}^2 - 2\sum_{i=1}^{N} p_{i,c}y_{i,c} = \sum_{i=1}^{N} (p_{i,c} - y_{i,c})^2 \geq 0$.

We observe that for the *i–th* instance, the Dice loss generates a non-positive gradient for the target class and a non-negative gradient for other classes. The gradient of the non-target class c for the *i–th* instance is determined by both probability $p_{i,c}$ and the whole segmentation results of this class $4\frac{I_c}{U_c^2}$, which highlights that Dice loss is a region-based loss. Furthermore, even if only one instance of class c is predicted correctly, $I_c > 0$, thus $p_{i,c}I_c > 0$. The gradient of this non-target class for the *i–th* voxel will be positive. However, for the situation of class absence, the gradient is always 0 as I_c is always 0. This is unreasonable. In contrast, this inconsistency does not exist in CE as it assigns a 0 gradient to all non-target classes, which will be analyzed in Eq. (6).

Cross Entropy. When utilized in a multi-class segmentation task, Cross Entropy quantifies the divergence of the predicted segmentation probability from the ground truth for each instance individually, subsequently computing the average value across all instances within the mini-batch,

$$\mathcal{L}_{\mathrm{CE}} = -\frac{1}{N}\sum_{c=0}^{C}\sum_{i=1}^{N} y_{i,c}\log p_{i,c} \tag{4}$$

The two-valued gradient of CE for each instance is defined as,

$$\frac{\partial \mathcal{L}_{\mathrm{CE}}}{\partial p_{i,c}} = \begin{cases} -\frac{1}{N}\frac{1}{p_{i,c}} & \text{if } y_{i,c}=1, \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

The gradient vector of the *i–th* instance can be written as,

$$\frac{\partial \mathcal{L}_{\mathrm{CE}}}{\partial p_{i}} = \left[0, 0, \cdots, -\frac{1}{N}\frac{1}{p_{i,c}}, \cdots, 0\right]^{T} \tag{6}$$

It can be observed that CE only calculates the gradient on the target class for each instance, multiplied by the same weight $-\frac{1}{N}$. Recalling the identified issues, enormous instances, and imbalanced class distribution in the sampled patches. These issues give rise to two problems: 1) all gradients are extremely diminutive to zero, as N is a number around several million; 2) different classes should not be weighted equally as the distribution is imbalanced. Although some studies have addressed the issue of mismatch weight of CE [25,33,39], they ignored the problem of the very diminutive gradient.

In summary, the intrinsic problems on the gradient of class-imbalanced SSLMIS are 1) the inconsistent gradients for non-target classes of Dice loss resulting from class absence; 2) the diminutive gradient of CE resulting from enormous instances; 3) the mismatch between the batch class distribution and weight of CE. We name these as gradient-bias in SSLMIS. The gradient biases slow down convergence speed and reduce segmentation performance, even failing to segment tail classes. The validation of gradient-bias is presented in the supplementary material. In the following sections, we will address these issues one by one.

3.2 Gradient-Aware Dice Loss

The gradient-bias of Dice loss stems from the class absence of patches. We can easily devise a specific treatment for this situation, which we defined as Gradient-Aware Dice (GADice) loss as follows,

$$\mathcal{L}_{\mathrm{GADice}} = \frac{1}{C+1}\sum_{c=0}^{C}\mathcal{L}_{\mathrm{GADice,c}} \tag{7}$$

where $\mathcal{L}_{\text{GADice,c}}$ is the loss for class c which is formulated as,

$$\mathcal{L}_{\text{GADice,c}} = \begin{cases} \frac{\langle p_{i,c} \rangle}{N} p_{i,c} & \text{if } \sum_{i=1}^{N} y_{i,c} = 0, \\ 1 - \frac{2I_c}{U_c} & \text{otherwise.} \end{cases} \tag{8}$$

where $\langle\rangle$ refers to stop-gradient.

Then, the two-valued gradient can be formulated as,

$$\frac{\partial \mathcal{L}_{\text{GADice}}}{\partial p_{i,c}} = \begin{cases} \frac{\langle p_{i,c} \rangle}{N} & \text{if } \sum_{i=1}^{N} y_{i,c} = 0, \\ -2\frac{U_c - 2p_{i,c} I_c}{U_c^2} & \text{else if } y_{i,c} = 1, \\ 4\frac{p_{i,c} I_c}{U_c^2} & \text{otherwise.} \end{cases} \tag{9}$$

Thus, if the class is not sampled into the batch, GADice loss backpropagates a gradient of $\frac{\langle p_{i,c} \rangle}{N}$. For non-target classes, when $p_{i,c}$ indicating better predictions is small, our added $\frac{\langle p_{i,c} \rangle}{N}$ is also small. It's probability-aware for the instance.

3.3 Grandient-Aware Cross Entropy

We propose a Gradient-Aware Cross Entropy (GACE) to address the issues of diminutive gradient and distribution mismatch of CE through class equilibrium via hard instance mining and dynamic weight equilibrium.

Class Equilibrium via Hard Instance Mining. Hard example mining typically encourages the networks to focus on hard samples during training [46], which is defined as,

$$\mathcal{L}_{\text{TopK}} = -\frac{1}{N_K} \sum_{c=0}^{C} \sum_{i \in \mathbf{K}} y_{i,c} \log p_{i,c} \tag{10}$$

where $\mathbf{K}$ is the set of $k\%$ worst voxels, and $N_K = N \times k\%$.

However, we are pleased to discover that we can partially balance the classes within a batch using this method. Though it's challenging for neural networks to differentiate every foreground class during segmentation, distinguishing background instances is relatively easy. By discarding a majority of the easy background instances, the classes of remaining instances are partially balanced, as shown in Fig. 2(c)–(d).

Dynamic Weight Equilibrium. Although most simple instances within the batch are discarded, the class imbalance issue still persists. The ideal scenario would be to dynamically match the gradient weights with class distribution in the batch at each training iteration. This is achievable, thanks to our proposed GACE, which is formulated as,

$$\mathcal{L}_{\text{GACE}} = -\left(\frac{1}{N_K}\right)^{\gamma} \sum_{c=0}^{C} \left(\frac{1}{N_c}\right)^{1-\gamma} \sum_{i \in \mathbf{K}} y_{i,c} \log p_{i,c} \tag{11}$$

where $N_c = \sum_{i \in \mathbf{K}} y_{i,c}$.

As shown in Eq. (11), GACE is computed: 1) calculate CE for each instance; 2) mine the $k\%$ worst instances; 3) assign class weight $\left(\frac{1}{N_c}\right)^{1-\gamma}$ for each class; 4) assign the volume weight $-\left(\frac{1}{N_K}\right)^{\gamma}$ for each mined instance. $\gamma \in [0,1]$ is the Weight Equilibrium Factor to adjust the batch's volume weight and class weight.

It should be emphasized that, unlike previous methods [25,33,39] that utilize the class distribution of the entire dataset, our class weight is calculated based on the batch itself. Therefore, our method does not require prior statistics but also can be dynamically matched with the class distribution of the batch at each iteration. Furthermore, the Weight Equilibrium Factor we introduced is extremely important. The experiments in Sect. 4.4 have demonstrated that only applying Dynamic Weight Equilibrium strategy can achieve significant improvements.

The two-valued gradient of our GACE is formulated by

$$\frac{\partial \mathcal{L}_{\text{GACE}}}{\partial p_{i,c}} = \begin{cases} -\left(\frac{1}{N_K}\right)^{\gamma}\left(\frac{1}{N_c}\right)^{1-\gamma}\frac{1}{p_{i,c}} & \text{if } y_{i,c}=1, \\ 0 & \text{otherwise.} \end{cases} \tag{12}$$

Compared to Eq. (5), $\left(\frac{1}{N_K}\right)^{\gamma}\left(\frac{1}{N_c}\right)^{1-\gamma} \geq \frac{1}{N}$ as $N_c \leq N_K \leq N$, the issue of diminutive gradient has also been alleviated (= achieved only when $k = 100$ & $\gamma = 1$ & all instances in the batch belong to background).

3.4 Gradient-Aware Loss and Application

Finally, our GA loss is the compound of GADice loss and GACE,

$$\mathcal{L}_{\text{GA}} = \mathcal{L}_{\text{GADice}} + \mathcal{L}_{\text{GACE}} \tag{13}$$

The values of $k = 10$ and $\gamma = 0.5$ are used in the experiments unless stated otherwise. It's convenient to utilize our GA loss to existing SSL methods as it's plug-and-play. Just replace the Dice loss and CE with our GADice loss and GACE. The only thing to note is that when using GACE with pseudo label related consistent loss, k should be set to 100. This is because hard instance mining strictly relies on accurate labels. We demonstrated that our GACE is still effective for consistent loss with pseudo labels in Sect. 4.4 and Table 6.

4 Experiments

4.1 Datasets

Synapse. The Synapse [22] consists of 30 CT scans with 13 organs. Specifically, the foreground classes include spleen (Sp), right/left kidney (RK/LK), gallbladder (Ga), esophagus (Es), liver (Li), stomach (St), aorta (Ao), inferior vena cava (IVC), portal & splenic veins (PSV), pancreas (Pa), right/left adrenal gland (RAG/LAG). To maintain consistency with [39], the dataset has been partitioned into 20, 4, and 6 for training, validation, and testing, respectively. All experiments are conducted thrice. **AMOS.** The AMOS [16] is a multi-organ dataset that consists of 360 subjects for 15 organs. In comparison to the Synapse, the AMOS introduces three new classes: prostate/uterus (P/U), duodenum (Du), and bladder (Bl), while excluding PSV. The data split [39] for training, validation, and testing is fixed at 216, 24, and 120, respectively. **ACDC.** [4] A four-class (background, right/left ventricle and myocardium) dataset with 100 scans. Following [2,26], we split training, validation, and testing as 70, 10, and 20.

4.2 Experimental Details and Evaluation Metrics

We conducted all experiments on a single NVIDIA A100 GPU (40G). For Synapse and AMOS, we use 3D V-Net [28] as the backbone which was optimized by the SGD with an initial learning rate of 0.01, following the warming-up strategy described in [7]. During training, we randomly cropped patches of size 96 × 96 × 96. Apart from the random crop, **no other** data augmentation operations or sampling strategies were employed. The batch size is set to 4, consisting of 2 labeled patches and 2 unlabeled patches. In the final testing phase, a sliding window approach is applied with the stride of $32 \times 32 \times 16$. We chose the DSC (%, Dice Score Coefficient) and ASD (Average Surface Distance in voxel) as our evaluation metrics. For ACDC, we use 2D U-Net [33] as the backbone. All parameters were set the same as [2,26,45]. DSC, Jaccard Score (%), 95% Hausdorff Distance (95HD) in voxel and ASD are chosen as evaluation metrics, following [2].

4.3 Comparison with the State-of-the-Art Methods

For Synapse and AMOS, we compare our method with eight state-of-the-art (SOTA) semi-supervised segmentation methods (MT [38], UA-MT [48], RDrop [44], CPS [9], DeSCO [5], DePL [41], Co-BioNet [30] and MagicNet [7]) and seven SOTA class-imbalanced designs (Adsh [13], CReST [42], SimiS [8], Basak *et al.* [3], CLD [25], DHC [39], and A&D [40]). MagicNet is trained for 34k iterations to achieve better convergence, while other methods, including ours, are trained for 17k iterations. For ACDC, we add SS-Net [45] and BCP [2]. More details are in the supplementary material.

Convenient and Robust on SSL Methods. The results of 20% labeled Synapse dataset are summarized in Table 1. Our GA loss is applied to eight SOTA SSL segmentation methods. Significant improvements can be seen for all eight methods: MT (↑ 12.09%, 45.58% to GA-MT with 57.67%), UA-MT (↑ 19.56%), RDrop (↑ 21.07%), CPS (↑ 17.90%), DePL (↑ 6.05%) and MagicNet (↑ 7.86%). Our method demonstrates great superiority in tail classes (small organs), avoiding situations of terrible failing prediction on all of the methods. For example, the UA-MT gets zero in the gallbladder, esophagus, and right/left adrenal gland while our GA-UAMT achieves 27.0%, 38.5%, 43.8%, and 51.6% in DSC, respectively. A similar conclusion can be obtained even if only two labeled cases are used (10% labeled, shown in supplementary material). Overall, the proposed GA loss is very convenient and robust on different SSL methods.

Outperform SOTA Class-Imbalanced Designs. As shown in Table 1, GA loss surpasses all SOTA class-imbalanced designs. Our methods achieve the best results within the same framework. Our GA-MT outperforms Basak *et al.*'s method (↑ 9.28%) which utilized a class-wise sampling strategy on MT. Our GA-CPS demonstrated superior performance compared to all CPS-based class-imbalanced designs (Adsh, CReST, SimiS, CLD, and DHC), surpassing the best CReST by 6.41% among them. Similar conclusions can be obtained with 5% labeled AMOS in Table 2 and ACDC in Table 3. Visualizations and more quantitative results of Synapse and AMOS can be found in the supplementary material.

Fast Convergence Speed. We visualized the validation DSC of some classes in Fig. 1. MagicNet converges slowly and achieves around 43% after training for 17k iterations. Most methods follow a step-by-step segmentation progress, from simple to hard, like climbing stairs. However, after applying our GA loss, we are surprised to discover that neural networks obtained prediction of all classes after only 2k iterations. They **can** segment all classes simultaneously.

Table 1. Quantitative comparison (Average DSC ↑, ASD ↓) between our approach and existing SOTA methods on 20% labeled Synapse dataset. *General* or *Imbalance* indicates whether the class imbalance issue is considered or not. **GA-** indicates the SOTA method we applied our GA loss. We report the *mean*±*std* repeated three times.

	Methods	Average		Average Dice of Each Class												
		DSC↑	ASD↓	Sp	RK	LK	Ga	Es	Li	St	Ao	IVC	PSV	PA	RAG	LAG
	VNet (fully)	68.49 ± 3.5	6.08 ± 5.4	90.2	91.9	90.7	38.3	30.9	94.8	75.6	79.1	81.4	62.1	48.5	48.9	58.0
General	MT [38]	45.58 ± 1.9	26.44 ± 13.3	80.1	82.2	75.4	12.9	0.0	86.7	41.9	65.4	66.7	36.4	16.6	8.6	19.7
	UA-MT [48]	41.37 ± 4.8	41.04 ± 1.8	75.2	81.0	66.8	0.0	0.0	86.9	37.9	69.4	67.8	31.1	21.7	0.0	0.0
	RDrop [44]	39.90 ± 0.6	49.25 ± 1.8	75.1	82.2	81.7	0.0	0.0	87.7	39.8	71.3	63.6	0.0	17.3	0.0	0.0
	CPS [9]	48.49 ± 1.2	38.93 ± 0.9	83.9	87.8	85.8	0.0	0.0	92.3	50.2	75.0	74.3	55.9	25.3	0.0	0.0
	DeSCO [5]	45.74 ± 0.7	46.07 ± 3.8	82.4	89.4	87.4	0.0	0.0	89.0	49.6	75.3	76.3	1.8	26.8	0.0	0.0
	DePL [41]	59.44 ± 2.3	8.10 ± 4.2	84.4	87.4	85.7	5.5	22.1	90.9	58.7	75.4	77.4	55.8	37.4	43.5	48.6
	Co-BioNet [30]	58.83 ± 2.7	7.50 ± 5.8	82.8	90.0	86.5	11.6	19.5	92.3	47.7	77.5	77.7	51.3	30.3	47.5	50.2
	MagicNet [7]	60.57 ± 2.5	22.48 ± 6.3	82.5	91.0	89.5	11.2	0.0	89.4	62.7	77.6	79.0	66.1	47.3	36.8	54.3
Imbalance	Adsh [13]	44.06 ± 2.8	39.43 ± 1.0	77.2	81.2	77.1	0.0	0.0	86.1	43.1	70.7	71.8	43.7	21.9	0.0	0.0
	CReST [42]	59.98 ± 1.3	6.56 ± 1.0	77.3	87.6	85.6	19.4	36.5	90.0	49.5	76.3	72.6	51.0	37.6	43.3	53.2
	SimiS [8]	50.45 ± 2.7	33.11 ± 3.6	83.3	90.8	85.8	9.2	0.0	85.6	55.0	73.6	71.7	50.4	34.0	0.0	16.6
	Basak *et al.* [3]	48.39 ± 0.9	38.33 ± 0.3	84.6	86.9	79.8	0.0	0.0	90.2	54.6	72.6	73.2	55.5	31.6	0.0	0.0
	CLD [25]	49.47 ± 2.9	34.73 ± 7.6	83.3	86.7	85.7	1.3	0.0	85.9	49.1	74.5	76.3	52.4	33.8	14.1	0.0
	DHC [39]	58.97 ± 2.4	8.23 ± 0.8	81.6	87.5	85.5	12.4	27.4	88.8	51.7	74.3	73.7	55.2	33.3	46.1	49.1
	A&D [40]	60.88 ± 0.7	**2.52 ± 0.4**	85.2	66.9	67.0	**52.7**	**62.9**	89.6	52.1	**83.0**	74.9	41.8	43.4	44.8	27.2
	GA-MT	57.67 ± 3.4	5.83 ± 1.0	75.6	88.7	82.6	16.1	41.1	90.3	43.9	72.7	67.6	51.9	33.0	41.2	45.0
	GA-UAMT	60.93 ± 2.3	4.00 ± 0.5	81.0	87.1	85.5	27.0	38.5	89.8	54.7	75.2	72.9	52.9	32.2	43.8	51.6
	GA-RDrop	60.97 ± 1.4	4.83 ± 1.6	83.4	89.8	85.2	17.8	38.1	91.2	52.6	77.2	77.2	58.3	31.3	45.6	45.0
	GA-CPS	66.39 ± 0.9	5.25 ± 0.8	84.1	**92.8**	87.9	25.3	41.1	**92.8**	**66.9**	78.0	79.8	64.0	47.8	49.4	53.4
	GA-DeSCO	59.02 ± 0.6	5.19 ± 1.2	76.7	92.4	87.1	28.4	30.4	88.2	49.3	76.3	74.6	52.5	33.3	42.1	39.1
	GA-DePL	65.49 ± 1.8	6.52 ± 1.3	**87.6**	89.9	89.6	18.8	35.5	92.7	62.6	78.3	81.0	63.8	46.1	**51.1**	54.4
	GA-Co-BioNet	60.47 ± 2.8	5.79 ± 1.5	82.3	91.8	86.7	12.8	36.5	90.0	53.0	77.5	74.6	55.6	35.0	43.5	47.6
	GA-MagicNet	**68.43 ± 0.5**	3.11 ± 0.2	81.4	92.4	**90.8**	33.5	53.3	89.1	60.9	79.1	**82.1**	**66.7**	**48.7**	50.3	**61.4**

Table 2. Quantitative comparison on 5% labeled AMOS dataset.

	Methods	Average		Average Dice of Each Class														
		DSC	ASD	Sp	RK	LK	Ga	Es	Li	St	Ao	IVC	PA	RAG	LAG	Du	Bl	P/U
	VNet (fully)	76.50	2.01	92.2	92.2	93.3	65.5	70.3	95.3	82.4	91.4	85.0	74.9	58.6	58.1	65.6	64.4	58.3
General	MT [38]	43.35	37.09	75.1	74.4	70.5	36.5	0.0	86.9	36.0	71.9	58.9	41.5	0.0	0.0	23.1	56.4	19.2
	UA-MT [48]	42.16	15.48	59.8	64.9	64.0	35.3	34.1	77.7	37.8	61.0	46.0	33.3	26.9	12.3	18.1	29.7	31.6
	RDrop [44]	39.87	45.02	70.5	73.5	69.1	6.7	0.0	84.9	32.3	64.9	52.8	43.5	0.0	0.0	19.1	61.7	19.2
	CPS [9]	41.08	20.37	56.1	60.3	59.4	33.3	25.4	73.8	32.4	65.7	52.1	31.1	25.5	6.2	18.4	40.7	35.8
	DeSCO [5]	44.39	43.51	78.9	81.4	81.8	6.7	0.0	**88.2**	44.2	78.9	61.5	37.2	0.0	0.0	21.2	66.9	19.2
	DePL [41]	41.97	20.42	55.7	62.4	57.7	36.6	31.3	68.4	33.9	65.6	51.9	30.2	23.3	10.2	20.9	43.9	37.7
	Co-BioNet [30]	48.32	26.04	76.6	82.1	75.1	41.5	38.2	87.9	40.4	75.2	53.7	40.8	4.8	0.0	25.1	64.2	19.2
	MagicNet [7]	54.08	29.03	**80.0**	84.5	86.1	47.9	0.0	85.1	50.7	81.7	69.3	57.2	46.0	0.0	**40.8**	62.9	19.2
Imbalance	Adsh [13]	40.33	24.53	56.0	63.6	57.3	34.7	25.7	73.9	30.7	65.7	51.9	27.1	20.2	0.0	18.6	43.5	35.9
	CReST [42]	46.55	14.62	66.5	64.2	65.4	36.0	32.2	77.8	43.6	68.5	52.9	40.3	24.7	19.5	26.5	43.9	36.4
	SimiS [8]	47.27	11.51	77.4	72.5	68.7	32.1	14.7	86.6	46.3	74.6	54.2	41.6	24.4	17.9	21.9	47.9	28.2
	Basak *et al.* [3]	38.73	31.76	68.8	59.0	54.2	29.0	0.0	83.7	39.3	61.7	52.1	34.6	0.0	0.0	26.8	45.7	26.2
	CLD [25],	46.10	15.86	67.2	68.5	71.4	41.0	21.0	76.1	42.4	69.8	52.1	37.9	24.7	23.4	22.7	38.1	35.2
	DHC [39]	49.53	13.89	68.1	69.6	71.1	42.3	37.0	76.8	43.8	70.8	57.4	43.2	27.0	28.7	29.1	41.4	36.7
	A&D [40]	37.82	44.31	72.8	67.5	64.4	14.6	0.0	82.3	44.6	70.7	51.9	38.1	0.0	0.0	23.7	36.7	0.2
	GA-MT	51.24	11.38	75.8	73.9	73.1	39.1	42.7	83.5	41.1	73.7	52.9	41.0	38.4	25.1	25.0	59.1	24.2
	GA-UAMT	51.70	11.18	70.7	73.9	69.6	38.4	35.7	84.9	40.7	67.4	58.2	35.5	38.8	27.1	31.1	64.2	39.3
	GA-RDrop	53.15	13.50	76.7	77.0	70.2	38.9	35.9	84.6	42.8	72.3	63.4	41.9	43.6	25.5	27.1	64.3	33.1
	GA-CPS	57.17	10.18	79.5	77.7	76.4	44.7	43.1	87.1	48.7	77.6	62.6	47.4	40.5	26.6	32.5	68.2	44.8
	GA-DeSCO	54.51	10.22	77.3	75.6	76.7	37.4	45.4	85.9	42.5	76.5	63.2	38.2	42.4	29.1	28.5	63.6	35.4
	GA-DePL	57.05	10.87	78.5	77.7	79.6	44.1	41.4	85.5	47.5	75.2	62.0	49.0	44.3	30.0	31.4	67.0	42.6
	GA-Co-BioNet	53.91	11.17	77.7	77.9	74.9	42.3	37.2	87.7	44.3	71.4	55.0	41.3	36.5	31.1	24.6	64.0	42.6
	GA-MagicNet	**63.51**	**4.58**	78.9	**85.5**	**87.2**	**50.0**	**49.1**	86.9	**56.2**	**83.4**	**70.3**	**57.4**	**49.1**	**40.8**	38.3	**71.6**	**47.9**

Table 3. Quantitative comparison on 5% & 10% labeled ACDC dataset.

	Methods	DSC ↑	Jaccard ↑	95HD ↓	ASD ↓	DSC ↑	Jaccard ↑	95HD ↓	ASD ↓
	U-Net (fully)	91.44	84.59	4.30	0.99				
		5% Labeled				10% Labeled			
General	MT [38]	47.98	37.47	27.35	11.01	80.15	68.59	14.50	4.01
	UAMT [44]	49.48	38.92	17.29	5.48	80.13	68.75	19.59	6.11
	DePL [41]	47.77	37.05	39.06	16.15	81.04	70.16	12.02	3.51
	CPS [9]	50.11	40.97	8.78	1.83	85.11	75.28	5.42	1.84
	RDrop [44]	46.69	37.44	7.93	1.83	84.21	74.30	4.83	1.44
	SS-Net [45]	65.83	55.38	6.67	2.28	86.78	77.67	6.07	1.40
	BCP [2]	87.59	78.67	**1.90**	**0.67**	88.84	80.62	3.98	1.17
Imbalance	Adsh [13]	52.90	41.06	6.51	0.83	83.03	72.60	7.96	2.34
	CReST [42]	46.11	35.92	38.40	14.74	81.10	70.34	12.82	4.08
	SimiS [8]	63.50	51.73	23.17	7.04	76.35	77.02	7.48	2.41
	Basak *et al.* [3]	52.60	40.30	43.91	16.11	81.70	70.67	7.92	2.24
	CLD [25]	58.49	47.13	28.83	10.99	86.43	77.27	9.69	2.48
	DHC [39]	57.88	46.86	31.18	2.04	85.71	75.95	8.78	2.63
	GA-MT	64.72	52.87	32.02	10.44	85.53	75.62	11.17	4.75
	GA-UAMT	63.72	52.86	25.93	8.44	85.85	76.05	8.93	2.72
	GA-DePL	60.65	49.46	28.44	9.80	84.88	74.95	11.60	3.18
	GA-CPS	66.42	54.36	29.91	9.47	87.46	78.46	9.35	2.20
	GA-RDrop	67.22	56.92	10.98	2.67	87.71	78.87	8.11	1.92
	GA-SS-Net	73.39	63.34	6.64	2.45	88.14	79.44	4.18	1.30
	GA-BCP	**88.24**	**79.60**	3.91	1.11	**89.31**	**81.27**	**3.32**	**1.01**

4.4 Ablation Study

We conduct ablation studies on Synapse dataset with 20% labeled data to validate the effectiveness of each module.

Effectiveness of Each Component in GA Loss. We conduct ablation studies to show the impact of each component in GA loss in Table 4. The first row indicates the MagicNet baseline model trained for 17k iterations. **GD**, **ClsE** and **DWE** indicate the proposed GADice loss, class equilibrium via hard instance mining, and dynamic weight equilibrium in our proposed GACE, which increase

Table 4. Ablation study on 20% labeled Synapse dataset. **GD**: GADice loss. **ClsE**: class equilibrium in GACE. **DWE**: dynamic weight equilibrium in GACE.

#	GD	ClsE	DWE	Avg. DSC	Average Dice of Each Class												
					Sp	RK	LK	Ga	Es	Li	St	Ao	IVC	PSV	PA	RAG	LAG
0				42.33 ± 8.5	51.1	75.4	66.9	16.7	0.0	86.9	55.5	75.9	76.1	6.7	39.2	0.0	0.0
1	✓			44.98 ± 0.4	74.4	89.3	88.0	0.0	0.0	88.1	59.0	50.7	69.3	26.2	39.7	0.0	0.0
2		✓		56.46 ± 7.2	81.4	91.3	89.7	10.1	12.9	89.0	59.7	75.9	79.1	61.9	46.0	16.8	20.1
3			✓	66.96 ± 1.2	81.0	91.6	90.6	31.7	41.7	88.6	65.1	77.8	81.9	66.2	50.3	47.3	56.6
4	✓	✓		59.40 ± 6.0	82.8	91.1	89.6	17.7	7.5	91.6	62.5	78.1	81.9	66.9	47.3	35.0	20.2
5	✓		✓	67.18 ± 2.0	81.9	91.1	90.6	31.1	32.5	89.8	67.5	79.7	81.6	67.0	50.0	50.6	59.8
6		✓	✓	67.99 ± 0.7	83.2	91.9	91.0	29.9	47.8	90.0	65.6	79.1	82.0	63.6	47.6	50.5	61.9
7	✓	✓	✓	68.43 ± 0.5	81.4	92.4	90.8	33.5	53.3	89.1	60.9	79.1	82.1	66.7	48.7	50.3	61.4

Table 5. (a) Comparison of different γ for balancing the volume weight $\left(\frac{1}{N_K}\right)^{\gamma}$ and class weight $\left(\frac{1}{N^c}\right)^{1-\gamma}$ in GACE Eq. (11). **(b)** Comparison of the proportion of Hard Instance Mining in GACE Eq. (11). k refers to k% worst instances. **(c)** Results with three designs of GADice loss for class absence issue. **0**: classic Dice loss. $\langle\rangle$: stop gradient.

(a)

γ	DSC ↑	ASD ↓
1	59.40±6.0	20.00±15.8
0.75	67.28±1.0	3.64±0.8
0.5	68.43±0.5	3.11±0.2
0.25	68.40±0.8	3.70±1.0
0	52.22±1.3	9.80±2.5

(b)

k	DSC ↑	ASD ↓
100	67.18±2.0	7.20±6.9
50	68.27±0.7	3.63±0.4
30	68.32±1.6	3.56±0.3
10	68.43±0.5	3.11±0.2
5	67.82±1.1	3.11±0.1

(c)

	DSC ↑	ASD ↓
0	67.99±0.7	3.22±0.5
$\frac{1}{N^2}$	68.14±1.0	3.10±0.3
$\frac{\langle p_{i,c}^2\rangle}{N}$	68.24±0.3	3.31±0.3
$\frac{\langle p_{i,c}\rangle}{N}$	68.43±0.5	3.11±0.2

the performance from 42.33% to 44.98%, 56.46%, and 66.96%, respectively. It's impressive that there is a significant performance gain by only applying **DWE**, showing the importance of balancing the volume weight and class weight. We can see that combining GADice loss with hard instance mining can improve 2.94% (56.46% to 59.40%, #4), and GADice loss with weight balance leads to 67.18% (#5). The performance improved to 67.99% (#6) when only applying GACE. Finally, our proposed GA loss provides a significant improvement to 68.43%.

Importance of Weight Equilibrium Factor γ. As illustrated in Table Table 4, the dynamic weight equilibrium γ of GACE is the most significant contributor to our GA loss. In this section, we examine how it affects performance by adjusting the volume weight $\left(\frac{1}{N_K}\right)^{\gamma}$ and class weight $\left(\frac{1}{N^c}\right)^{1-\gamma}$, as shown in Table 5(a). We decrease γ from 1.0 to 0.0, where $\gamma = 1$ indicates only volume weight is applied in GACE, where $\gamma = 0$ indicates only class weight is adopted. It can be observed that when γ decreases, the performance improves rapidly as the importance of class weight increases. However, when gamma decreases to 0 for only applying the class weight, the performance drops significantly due to the severely imbalanced class distribution in the batch, resulting in fluctuations in network optimization. Best performance is achieved when $\gamma = 0.5$ demonstrates the importance of balancing volume and class weight.

Proportion (k%) of Hard Instance Mining. Here, we validate another hyper-parameter k in our GACE, which controls the proportion of Hard Instance Mining. We study the results when k is set to 100, 50, 30, 10, 5. The results are shown in Table 5(b). The performance increases gradually when decreasing the number of instances mined, and being best at $k = 10$. Of course, selecting too few instances can be detrimental to optimization. However, there is still an improvement even when only 5% of the worst instances are selected.

Design Choices of GADice Loss. As illustrated in Sect. 3.2, we propose to introduce a gradient of $\frac{\langle p_{i,c} \rangle}{N}$ for class absence issue, and demonstrate its effectiveness in Table 4. Here, we design two more strategies of GADice loss, $\frac{1}{N^2}$ and $\frac{\langle p_{i,c}^2 \rangle}{N}$, to compare different designs with GA-MagicNet on 20% labeled Synapse dataset. The results are shown in Table 5(c). It can be seen that the performance improved after adding the gradient for the condition of class absence, while $\frac{\langle p_{i,c} \rangle}{N}$ works favorably for the semi-supervised medical image segmentation. Probability-aware is effective as performance of both $\frac{\langle p_{i,c} \rangle}{N}$ and $\frac{\langle p_{i,c}^2 \rangle}{N}$ are better than $\frac{1}{N^2}$.

Table 6. Effectiveness of GA loss on labeled, unlabeled data. GA_U-CPS: only apply GA loss on unlabeled data. GA_L-CPS: only apply GA loss on labeled data. GA-CPS: apply GA loss on both labeled and unlabeled data.

	20% labeled Synapse		5% labeled AMOS	
	DSC ↑	ASD↓	DSC ↑	ASD ↓
CPS [9]	48.49 ± 1.2	38.93 ± 0.9	41.08	20.37
GA_U-CPS	56.46 ± 2.1	13.49 ± 4.0	53.84	21.36
GA_L-CPS	59.12 ± 2.8	4.93 ± 0.6	53.58	11.46
GA-CPS	66.39 ± 0.9	5.25 ± 0.8	57.17	10.18

Effectiveness of GA Loss on Labeled, Unlabeled Data. To explore whether the improvements come from labeled or unlabeled data, we validate our GA loss on CPS [9] with 20% labeled Synapse and 5% labeled AMOS. CPS utilizes CE + Dice loss as a supervised loss on labeled data, and CE as a consistent loss on pseudo-labels generated from unlabeled data. As shown in Table 6, **GA_U-CPS** and **GA_L-CPS** represent we only replace the consistent loss, or supervised loss with our GA loss in CPS, which increases the DSC from 49.78% to 56.46% and 59.12% on 20% labeled Synapse dataset, respectively. It can be seen that applying GA loss on both labeled and unlabeled data provides a significant improvement to 66.39%. Similar results are shown in 5% labeled AMOS dataset.

More discussions about 1) comparison of labeled, unlabeled training data, validation data, and testing data, 2) input patch size, and 3) variants of Dice loss are addressed in supplementary material.

5 Conclusion

We identify the intrinsic gradient-bias of class-imbalanced SSLMIS that results from the extrinsic data-bias. We propose a GA loss to alleviate the intrinsic gradient-bias by class equilibrium via hard instance mining, dynamic weight equilibrium between the batch's volume weight and class weight in CE, and adding a probability-aware gradient in Dice loss. Extensive experiments on three public datasets demonstrate the effectiveness and superiority of our method. **Limitations.** Although we have identified the significance of dynamic weight equilibrium and found $\gamma = 0.5$ exhibits promising performance, we have not theoretically analyzed the optimal value of γ, or whether it can be adjusted during training based on performance. This could potentially be a future direction.

References

1. Abraham, N., Khan, N.M.: A novel focal tversky loss function with improved attention U-Net for lesion segmentation. In: 2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019), pp. 683–687. IEEE (2019)
2. Bai, Y., Chen, D., Li, Q., Shen, W., Wang, Y.: Bidirectional copy-paste for semi-supervised medical image segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11514–11524 (2023)
3. Basak, H., Ghosal, S., Sarkar, R.: Addressing class imbalance in semi-supervised image segmentation: a study on cardiac MRI. In: Wang, L., Dou, Q., Fletcher, P.T., Speidel, S., Li, S. (eds.) MICCAI 2022. LNCS, vol. 13438, pp. 224–233. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-16452-1_22
4. Bernard, O., et al.: Deep learning techniques for automatic MRI cardiac multi-structures segmentation and diagnosis: is the problem solved? IEEE Trans. Med. Imaging **37**(11), 2514–2525 (2018)
5. Cai, H., Li, S., Qi, L., Yu, Q., Shi, Y., Gao, Y.: Orthogonal annotation benefits barely-supervised medical image segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3302–3311 (2023)
6. Chen, B., Jiang, J., Wang, X., Wan, P., Wang, J., Long, M.: Debiased self-training for semi-supervised learning. In: Advances in Neural Information Processing Systems 35, pp. 32424–32437 (2022)
7. Chen, D., Bai, Y., Shen, W., Li, Q., Yu, L., Wang, Y.: MagicNet: semi-supervised multi-organ segmentation via magic-cube partition and recovery. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 23869–23878 (2023)
8. Chen, H., et al.: An embarrassingly simple baseline for imbalanced semi-supervised learning. arXiv preprint arXiv:2211.11086 (2022)
9. Chen, X., Yuan, Y., Zeng, G., Wang, J.: Semi-supervised semantic segmentation with cross pseudo supervision. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2613–2622 (2021)

10. Chen, D.-D., Wang, W., Gao, W., Zhou, Z.H.: Tri-net for semi-supervised deep learning. In: Proceedings of Twenty-Seventh International Joint Conference on Artificial Intelligence, pp. 2014–2020 (2018)
11. Drozdzal, M., Vorontsov, E., Chartrand, G., Kadoury, S., Pal, C.: The importance of skip connections in biomedical image segmentation. In: Carneiro, G., et al. (eds.) LABELS/DLMIA -2016. LNCS, vol. 10008, pp. 179–187. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46976-8_19
12. Gonzalez-Jimenez, A., Lionetti, S., Gottfrois, P., Gröger, F., Pouly, M., Navarini, A.A.: Robust T-loss for medical image segmentation. In: Greenspan, H., et al. (eds.) MICCAI 2023. LNCS, vol. 14222, pp. 714–724. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-43898-1_68
13. Guo, L.Z., Li, Y.F.: Class-imbalanced semi-supervised learning with adaptive thresholding. In: International Conference on Machine Learning, pp. 8082–8094. PMLR (2022)
14. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. In: Advances in Neural Information Processing Systems 33, pp. 6840–6851 (2020)
15. Hong, Y., Han, S., Choi, K., Seo, S., Kim, B., Chang, B.: Disentangling label distribution for long-tailed visual recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6626–6636 (2021)
16. Ji, Y., et al.: AMOS: a large-scale abdominal multi-organ benchmark for versatile medical image segmentation. In: Advances in Neural Information Processing Systems 35, pp. 36722–36732 (2022)
17. Karimi, D., Salcudean, S.E.: Reducing the hausdorff distance in medical image segmentation with convolutional neural networks. IEEE Trans. Med. Imaging **39**(2), 499–513 (2019)
18. Kervadec, H., Bouchtiba, J., Desrosiers, C., Granger, E., Dolz, J., Ayed, I.B.: Boundary loss for highly unbalanced segmentation. In: International Conference on Medical Imaging with Deep Learning, pp. 285–296. PMLR (2019)
19. Kervadec, H., de Bruijne, M.: On the dice loss gradient and the ways to mimic it. arXiv preprint arXiv:2304.04319 (2023)
20. Kodym, O., Španěl, M., Herout, A.: Segmentation of head and neck organs at risk using CNN with batch dice loss. In: Brox, T., Bruhn, A., Fritz, M. (eds.) GCPR 2018. LNCS, vol. 11269, pp. 105–114. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-12939-2_8
21. Lai, Z., Wang, C., Cheung, S.C., Chuah, C.N.: SAR: self-adaptive refinement on pseudo labels for multiclass-imbalanced semi-supervised learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4091–4100 (2022)
22. Landman, B., Xu, Z., Igelsias, J.E., Styner, M., Langerak, T.R., Klein, A.: 2015 MICCAI multi-atlas labeling beyond the cranial vault workshop and challenge. In: Proceedings of the MICCAI Multi-Atlas Labeling Beyond Cranial Vault—Workshop Challenge (2015)
23. Lee, D.H., et al.: Pseudo-label: the simple and efficient semi-supervised learning method for deep neural networks. In: Workshop on Challenges in Representation Learning, ICML, Atlanta, vol. 3, p. 896 (2013)
24. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2980–2988 (2017)
25. Lin, Y., Yao, H., Li, Z., Zheng, G., Li, X.: Calibrating label distribution for class-imbalanced barely-supervised knee segmentation. In: Wang, L., Dou, Q., Fletcher,

P.T., Speidel, S., Li, S. (eds.) MICCAI 2022. LNCS, vol. 13438, pp. 109–118. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-16452-1_11

26. Luo, X.: SSL4MIS (2020). https://github.com/HiLab-git/SSL4MIS
27. Ma, J., et al.: Loss odyssey in medical image segmentation. Med. Image Anal. **71**, 102035 (2021)
28. Milletari, F., Navab, N., Ahmadi, S.A.: V-Net: fully convolutional neural networks for volumetric medical image segmentation. In: 2016 Fourth International Conference on 3D Vision (3DV), pp. 565–571. IEEE (2016)
29. Oh, Y., Kim, D.J., Kweon, I.S.: DASO: distribution-aware semantics-oriented pseudo-label for imbalanced semi-supervised learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9786–9796 (2022)
30. Peiris, H., Hayat, M., Chen, Z., Egan, G., Harandi, M.: Uncertainty-guided dual-views for semi-supervised volumetric medical image segmentation. Nat. Mach. Intell. **5**(7), 724–738 (2023)
31. Qi, W., Wu, H., Chan, S.: MDF-Net: a multi-scale dynamic fusion network for breast tumor segmentation of ultrasound images. IEEE Trans. Image Process. **32**, 4842–4855 (2023)
32. Rizve, M.N., Duarte, K., Rawat, Y.S., Shah, M.: In defense of pseudo-labeling: an uncertainty-aware pseudo-label selection framework for semi-supervised learning. arXiv preprint arXiv:2101.06329 (2021)
33. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015, Part III. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
34. Salehi, S.S.M., Erdogmus, D., Gholipour, A.: Tversky loss function for image segmentation using 3D fully convolutional deep networks. In: Wang, Q., Shi, Y., Suk, H.-I., Suzuki, K. (eds.) MLMI 2017. LNCS, vol. 10541, pp. 379–387. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-67389-9_44
35. Sudre, C.H., Li, W., Vercauteren, T., Ourselin, S., Jorge Cardoso, M.: Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. In: Cardoso, M.J., et al. (eds.) DLMIA/ML-CDS-2017. LNCS, vol. 10553, pp. 240–248. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-67558-9_28
36. Taghanaki, S.A., et al.: Combo loss: handling input and output imbalance in multi-organ segmentation. Comput. Med. Imaging Graph. **75**, 24–33 (2019)
37. Tang, H., et al.: Clinically applicable deep learning framework for organs at risk delineation in CT images. Nat. Mach. Intell. **1**(10), 480–491 (2019)
38. Tarvainen, A., Valpola, H.: Mean teachers are better role models: weight-averaged consistency targets improve semi-supervised deep learning results. In: Advances in Neural Information Processing Systems 30 (2017)
39. Wang, H., Li, X.: DHC: dual-debiased heterogeneous co-training framework for class-imbalanced semi-supervised medical image segmentation. In: Greenspan, H., et al. (eds.) MICCAI 2023. LNCS, vol. 14222, pp. 582–591. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-43898-1_56
40. Wang, H., Li, X.: Towards generic semi-supervised framework for volumetric medical image segmentation. In: Advances in Neural Information Processing Systems 36 (2024)
41. Wang, X., Wu, Z., Lian, L., Yu, S.X.: Debiased learning from naturally imbalanced pseudo-labels. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 14647–14657 (2022)

42. Wei, C., Sohn, K., Mellina, C., Yuille, A., Yang, F.: CReST: a class-rebalancing self-training framework for imbalanced semi-supervised learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10857–10866 (2021)
43. Wong, K.C.L., Moradi, M., Tang, H., Syeda-Mahmood, T.: 3D segmentation with exponential logarithmic loss for highly unbalanced object sizes. In: Frangi, A.F., Schnabel, J.A., Davatzikos, C., Alberola-López, C., Fichtinger, G. (eds.) MICCAI 2018. LNCS, vol. 11072, pp. 612–619. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-00931-1_70
44. Wu, L., et al.: R-Drop: regularized dropout for neural networks. In: Advances in Neural Information Processing Systems 34, pp. 10890–10905 (2021)
45. Wu, Y., Wu, Z., Wu, Q., Ge, Z., Cai, J.: Exploring smoothness and class-separation for semi-supervised medical image segmentation. In: Wang, L., Dou, Q., Fletcher, P.T., Speidel, S., Li, S. (eds.) MICCAI 2022. LNCS, vol. 13435, pp. 34–43. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-16443-9_4
46. Wu, Z., Shen, C., van den Hengel, A.: Bridging category-level and instance-level semantic image segmentation. arXiv preprint arXiv:1605.06885 (2016)
47. You, C., Zhou, Y., Zhao, R., Staib, L., Duncan, J.S.: SimCVD: simple contrastive voxel-wise representation distillation for semi-supervised medical image segmentation. IEEE Trans. Med. Imaging **41**(9), 2228–2237 (2022)
48. Yu, L., Wang, S., Li, X., Fu, C.-W., Heng, P.-A.: Uncertainty-aware self-ensembling model for semi-supervised 3D left atrium segmentation. In: Shen, D., et al. (eds.) MICCAI 2019. LNCS, vol. 11765, pp. 605–613. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-32245-8_67
49. Zhu, W., et al.: AnatomyNet: deep learning for fast and fully automated whole-volume segmentation of head and neck anatomy. Med. Phys. **46**(2), 576–589 (2019)