Medical Image Analysis 67 (2021) 101851

Contents lists available at ScienceDirect

Medical Image Analysis

journal homepage: www.elsevier.com/locate/media

Boundary loss for highly unbalanced segmentation

Hoel Kervadec*, Jihene Bouchtiba, Christian Desrosiers, Eric Granger, Jose Dolz, Ismail Ben Ayed

ÉTS Montréal, CRCHUM (University of Montreal Hospital Centre) Canada

ARTICLE INFO

Article history:
Received 1 October 2019
Revised 29 May 2020
Accepted 24 July 2020
Available online 6 October 2020

Keywords:
Boundary loss
Unbalanced data
Semantic segmentation
Deep learning
CNN

ABSTRACT

Widely used loss functions for CNN segmentation, e.g., Dice or cross-entropy, are based on integrals over the segmentation regions. Unfortunately, for highly unbalanced segmentations, such regional summations have values that differ by several orders of magnitude across classes, which affects training performance and stability. We propose a *boundary* loss, which takes the form of a distance metric on the space of contours, not regions. This can mitigate the difficulties of highly unbalanced problems because it uses integrals over the interface between regions instead of unbalanced integrals over the regions. Furthermore, a boundary loss complements regional information. Inspired by graph-based optimization techniques for computing active-contour flows, we express a non-symmetric L_2 distance on the space of contours as a regional integral, which avoids completely local differential computations involving contour points. This yields a boundary loss expressed with the regional softmax probability outputs of the network, which can be easily combined with standard regional losses and implemented with any existing deep network architecture for N-D segmentation. We report comprehensive evaluations and comparisons on different unbalanced problems, showing that our boundary loss can yield significant increases in performances while improving training stability. Our code is publicly available[1].

1. Introduction

Recent years have witnessed a substantial growth in the number of deep learning methods for medical image segmentation (Litjens et al., 2017; Shen et al., 2017; Dolz et al., 2018; Ker et al., 2018). Widely used loss functions for segmentation, e.g., Dice or cross-entropy, are based on *regional* integrals, which are convenient for training deep neural networks. In practice, these regional integrals are summations over the segmentation regions of differentiable functions, each directly invoking the softmax probability outputs of the network. Therefore, standard stochastic optimizers such as SGD are directly applicable. Unfortunately, difficulties occur for highly unbalanced segmentations, for instance, when the size of target foreground region is several orders of magnitude less than the background size. For example, in the characterization of white matter hyperintensities (WMH) of presumed vascular origin, the foreground composed of WMH regions may be 500 times smaller than the background (see the typical example in Fig. 1). In such cases, quite common in medical image analysis, standard regional losses contain foreground and background terms with values that differ considerably, typically by several orders of magnitude, potentially affecting performance and training stability (Milletari et al., 2016; Sudre et al., 2017).

Segmentation approaches based on convolutional neural networks (CNN) are typically trained by minimizing the cross-entropy (CE), which measures an affinity between the regions defined by probability softmax outputs of the network and the corresponding ground-truth regions. The standard regional CE has well-known drawbacks in the context of highly unbalanced problems. It assumes identical importance distribution of all the samples and classes. To achieve good generalization, it requires a large training set with balanced classes. For unbalanced data, CE typically results in unstable training and leads to decision boundaries biased towards the majority classes. Class-imbalanced learning aims to mitigate learning bias by promoting the importance of infrequent labels. In medical image segmentation, a common strategy is to re-balance class prior distributions by down-sampling frequent labels (Havaei et al., 2017; Valverde et al., 2017). Nevertheless, this strategy limits the information of the images used for training. Another common practice is to assign weights to the different classes that are inversely proportional to the frequency of the corresponding labels (Brosch et al., 2015; Ronneberger et al., 2015; Kamnitsas et al., 2017; Long et al., 2015; Yu et al., 2017). In this scenario, the standard cross-entropy (CE) loss is modified so as to assign

* Corresponding author:
E-mail address: hoel@kervadec.science (H. Kervadec).

[1] https://github.com/LIVIAETS/surface-loss

https://doi.org/10.1016/j.media.2020.101851
1361-8415/© 2020 Elsevier B.V. All rights reserved.

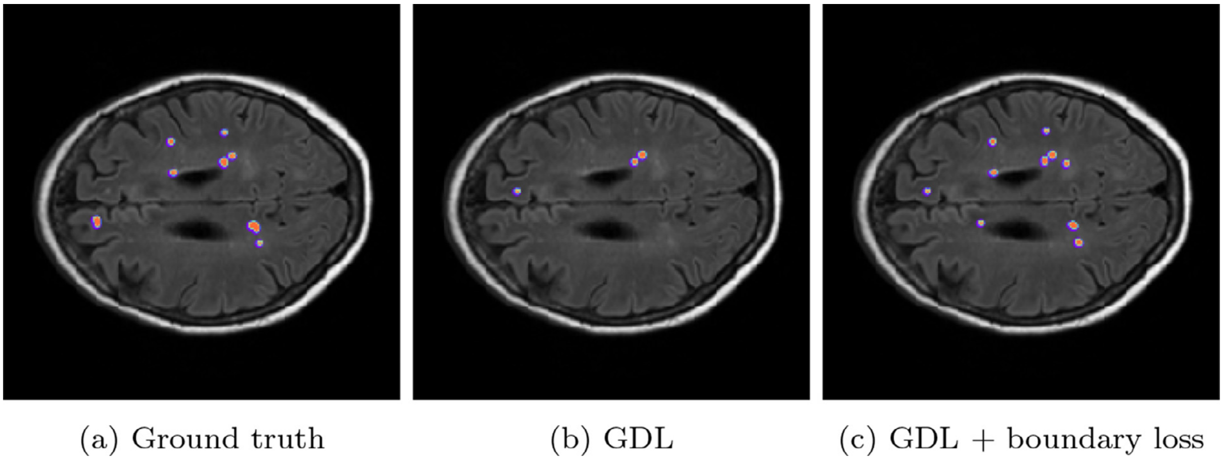


Fig. 1. A visual comparison that shows the positive effect of our boundary loss on a validation data from the WMH dataset. Our boundary loss helped to recover small regions that were otherwise missed by the model trained with the generalized Dice loss (GDL). Best viewed in colors.

more importance to the rare labels. Although effective for some unbalanced problems, such weighting methods may undergo serious difficulties when dealing with highly unbalanced datasets, as seen with WMH segmentation. The CE gradient computed over the few pixels of infrequent labels is typically noisy, and amplifying this noise with a high class weight may lead to instability.

The well-known Dice overlap coefficient was also adopted as a regional loss function, typically outperforming CE in unbalanced medical image segmentation problems (Milletari et al., 2016; 2017; Wong et al., 2018). Sudre et al. (2017) generalized the Dice loss Milletari et al. (2016) by weighting according to the squared inverse of class-label frequency. Despite these improvements over CE (Milletari et al., 2016; Sudre et al., 2017), regional Dice losses may encounter difficulties when dealing with very small structures. In such highly unbalanced scenarios, mis-classified pixels may lead to large decreases of the loss, resulting in unstable optimization. Furthermore, Dice corresponds to the harmonic mean between precision and recall, implicitly using the arithmetic mean of false positives and false negatives. False positives and false negatives are, therefore, equally important when the true positives remain the same, making this loss mainly appropriate when both types of errors are equally high. The recent research in Salehi et al. (2017); Abraham and Khan (2018) investigated losses based on the Tversky similarity index in order to provide a better trade-off between precision and recall. It introduced two parameters that control the importance of false positives and false negatives. Other recent advances in class-imbalanced learning for computer vision problems have been adopted in medical image segmentation. For example, inspired by the concept of focal loss (Lin et al., 2018), Dice and Tvserky losses have been extended to integrate a focal term, which is parameterized by a value that controls the importance of easy and hard training samples (Abraham and Khan, 2018; Wong et al., 2018). Furthermore, the combination of several of these regional losses has been further investigated (Zhu et al., 2019). The main objective of these losses is to balance the classes not only in terms of their relative class sizes, but also by the level of segmentation difficulty.

More recently, Karimi and Salcudean (2019) proposed a novel loss function that attempts to directly reduce the Hausdorff distance (HD). This relaxed loss based on the HD is shown to bring improvements when combined with the DSC loss. Nevertheless, its main drawback is the high computational cost of computing the distance transforms. Particularly, at each training epoch, the new distance maps have to be recomputed for all the images, which incurs in a computationally costly process. This issue is further magnified in the case of 3D volumes, which heavily increases the computational burden.

1.1. Contributions

In this paper, we propose a *boundary* loss that takes the form of a distance metric on the space of contours (or shapes), not regions. We argue that a boundary loss can mitigate the issues related to regional losses in highly unbalanced segmentation problems. Rather than using unbalanced integrals over the regions, a boundary loss uses integrals over the boundary (interface) between the regions. Furthermore, it provides information that is complementary to regional losses. It is, however, challenging to represent the boundary points corresponding to the regional softmax outputs of a CNN. This difficulty may explain why boundary losses have been avoided in the context of deep segmentation networks. Our boundary loss is inspired by techniques in discrete graph-based optimization for computing gradient flows of curve evolution (Boykov et al., 2006). Following an integral approach for computing boundary variations, we express a non-symmetric L_2 distance on the space of shapes (or contours) as a regional integral, which avoids completely local differential computations involving contour points. This yields a boundary loss expressed as the sum of linear functions of the regional softmax probability outputs of the network. Therefore, it can be easily combined with standard regional losses and implemented with any existing deep network architecture for N-D segmentation.

We evaluated our boundary loss in conjunction with various region-based losses on two challenging and highly unbalanced segmentation problems – the Ischemic Stroke Lesion (ISLES) and the White Matter Hyperintensities (WMH) benchmark datasets. The results indicate that the proposed boundary loss yields a more stable learning process, and can bring significant gains in performances, in terms of Dice and Hausdorff scores.

2. Formulation

Let $I: \Omega \subset \mathbb{R}^{2,3} \rightarrow \mathbb{R}$ denotes a training image with spatial domain Ω, and g: $\Omega \rightarrow \{0, 1\}$ a binary ground-truth segmentation of the image: $g(p) = 1$ if pixel/voxel p belongs to the target region $G \subset \Omega$ (foreground region) and 0 otherwise, i.e., $p \in \Omega \backslash G$ (background region)[2]. Let s_θ: $\Omega \rightarrow [0, 1]$ denotes the softmax proba-

[2] We focus on two-region segmentation to simplify the presentation. However, our formulation extends to the multi-region case in a straightforward manner.

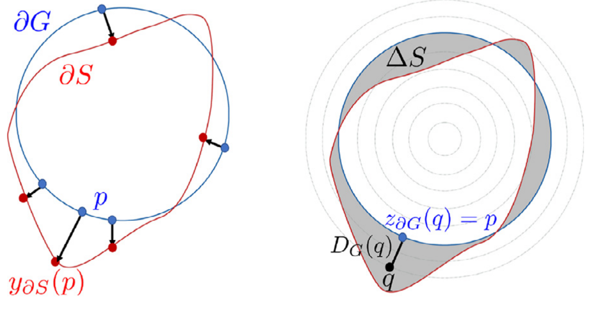
bility output of a deep segmentation network, and $S_\theta \subset \Omega$ the corresponding segmentation region: $S_\theta = \{p \in \Omega \,|\, s_\theta(p) \geq \delta\}$ for some threshold δ. Widely used segmentation loss functions involve a *regional integral* for each segmentation region in Ω, which measures some similarity (or overlap) between the region defined by the probability outputs of the network and the corresponding ground-truth. In the two-region case, we have an integral of the general form $\int_\Omega g(p) f(s_\theta(p))dp$ for the foreground, and of the form $\int_\Omega (1-g(p)) f(1-s_\theta(p))dp$ for the background. For instance, the standard two-region cross-entropy loss corresponds to a summation of these two terms for $f = -\log(\cdot)$. Similarly, the generalized Dice loss (GDL) Sudre et al. (2017) involves regional integrals with $f = 1$, subject to some normalization, and is given as follows for the two-region case:

$$\mathcal{L}_{GDL}(\theta) = 1 - 2\frac{w_G \int_{p\in\Omega} g(p)s_\theta(p)dp + w_B \int_{p\in\Omega}(1-g(p))(1-s_\theta(p))dp}{w_G \int_\Omega [s_\theta(p)+g(p)]dp + w_B \int_\Omega [2-s_\theta(p)-g(p)]dp} \quad (1)$$

where coefficients $w_G = 1/\left(\int_{p\in\Omega} g(p)dp\right)^2$ and $w_B = 1/\left(\int_\Omega (1-g(p))dp\right)^2$ are introduced to reduce the well-known correlation between the Dice overlap and region size.

Regional integrals are widely used because they are convenient for training deep segmentation networks. In practice, these regional integrals are summations of differentiable functions, each invoking directly the softmax probability outputs of the network, $s_\theta(p)$. Therefore, standard stochastic optimizers such SGD are directly applicable. Unfortunately, extremely unbalanced segmentations are quite common in medical image analysis, where, e.g., the size of the target foreground region is several orders of magnitude smaller than the background size. This represents challenging cases because the foreground and background terms have substantial differences in their values, which affects segmentation performance and training stability (Milletari et al., 2016; Sudre et al., 2017).

Our purpose is to build a boundary loss $\text{Dist}(\partial G, \partial S_\theta)$, which takes the form of a distance metric on the space of contours (or region boundaries) in Ω, with ∂G denoting a representation of the boundary of ground-truth region G (e.g., the set of points of G, which have a spatial neighbor in background $\Omega \backslash G$) and ∂S_θ denoting the boundary of the segmentation region defined by the network output. On the one hand, a boundary loss should be able to mitigate the above-mentioned difficulties for unbalanced segmentations: rather than using unbalanced integrals within the regions, it uses integrals over the boundary (interface) between the regions. Furthermore, a boundary loss provides information that is different from and, therefore, complimentary to regional losses. On the other hand, it is not clear how to represent boundary points on ∂S_θ as a differentiable function of regional network outputs s_θ. This difficulty might explain why boundary losses have been mostly avoided in the context of deep segmentation networks.



(a) Differential (b) Integral

Fig. 2. The relationship between *differential* and *integral* approaches for evaluating boundary change (variation).

Our boundary loss is inspired from discrete (graph-based) optimization techniques for computing gradient flows of curve evolution (Boykov et al., 2006). Similarly to our problem, curve evolution methods require a measure for evaluating boundary changes (or variations). Consider the following non-symmetric L_2 distance on the space of shapes, which evaluates the change between two nearby boundaries ∂S and ∂G (Boykov et al., 2006):

$$\text{Dist}(\partial G, \partial S) = \int_{\partial G} \|y_{\partial S}(p) - p\|^2 dp \quad (2)$$

where $p \in \Omega$ is a point on boundary ∂G and $y_{\partial S}(p)$ denotes the corresponding point on boundary ∂S, along the direction normal to ∂G, i.e., $y_{\partial S}(p)$ is the intersection of ∂S and the line that is normal to ∂G at p (See Fig. 2a for an illustration) $\|.\|$ denotes the L_2 norm. In fact, this *differential* framework for evaluating boundary change is in line with standard variational curve evolution methods (Mitiche and Ben Ayed, 2011), which compute the motion of each point p on the evolving curve as a velocity along the normal to the curve at point p. Similarly to any contour distance invoking directly points on contour ∂S, expression (2) cannot be used directly as a loss for $\partial S = \partial S_\theta$. However, it is easy to show that the differential boundary variation in (2) can be approximated using an *integral* approach (Boykov et al., 2006), which avoids completely local differential computations involving contour points and represents boundary change as a regional integral:

$$\text{Dist}(\partial G, \partial S) \approx 2\int_{\Delta S} D_G(q)dq \quad (3)$$

where ΔS denotes the region between the two contours and $D_G : \Omega \rightarrow \mathbb{R}^+$ is a *distance map* with respect to boundary ∂G, i.e., $D_G(q)$ evaluates the distance between point $q \in \Omega$ and the nearest point $z_{\partial G}(q)$ on contour ∂G: $D_G(q) = \|q - z_{\partial G}(q)\|$. Fig. 2b illustrates this integral framework for evaluating the boundary distance in Eq. (2). To clarify approximation (3), notice that integrating the distance map $2D_G(q)$ over the normal segment connecting a point p on ∂G and $y_{\partial S}(p)$ yields $\|y_{\partial S}(p) - p\|^2$, via the following variable change:

$$\int_p^{y_{\partial S}(p)} 2D_G(q)dq = \int_0^{\|y_{\partial S}(p)-p\|} 2D_G dD_G = \|y_{\partial S}(p) - p\|^2$$

Thus, from approximation (3), the non-symmetric L_2 distance between contours in Eq. (2) can be expressed as a sum of regional integrals based on a *level set* representation of boundary ∂G:

$$\begin{aligned}\frac{1}{2}\text{Dist}(\partial G, \partial S) &= \int_S \phi_G(q)dq - \int_G \phi_G(q)dq \\ &= \int_\Omega \phi_G(q)s(q)dq - \int_\Omega \phi_G(q)g(q)dq \end{aligned} \quad (4)$$

where s: $\Omega \rightarrow \{0, 1\}$ is binary indicator function of region S: $s(q) = 1$ if $q \in S$ belongs to the target and 0 otherwise. $\phi_G : \Omega \rightarrow \mathbb{R}$ denotes the level set representation of boundary ∂G: $\phi_G(q) = -D_G(q)$ if $q \in G$ and $\phi_G(q) = D_G(q)$ otherwise. Now, for $S = S_\theta$, i.e., replacing binary variables $s(q)$ in Eq. (4) by the softmax probability outputs of the network $s_\theta(q)$, we obtain the following boundary loss which, up to a constant independent of θ, approximates boundary distance $\text{Dist}(\partial G, \partial S_\theta)$:

$$\mathcal{L}_B(\theta) = \int_\Omega \phi_G(q)s_\theta(q)dq \quad (5)$$

Notice that we omitted the last term in Eq. (4) as it is independent of network parameters. The level set function ϕ_G is pre-computed directly from the ground-truth region G. In practice, our boundary loss in Eq. (5) is the sum of linear functions of the regional softmax probability outputs of the network. Therefore, it can be easily combined with standard regional losses ($\mathcal{L}_R$) and implemented with any existing deep network architecture for N-D segmentation:

$$\mathcal{L}_R(\theta) + \alpha \mathcal{L}_B(\theta), \tag{6}$$

where $\alpha \in \mathbb{R}$ is a parameter balancing the two losses.

It is worth noting that our boundary loss uses ground-truth boundary information via pre-computed level-set function $\phi_G(q)$, which encodes the distance between each point q and ∂G. In Eq. (5), the softmax for each point q is weighted by the distance function. Such distance-to-boundary information is omitted in widely used regional losses, where all the points within a given region are treated equally, independently of their distances from the boundary.

Notice that the global minimum (smallest possible value) of our boundary loss (5) is reached when all the negative values in the distance function are included in the sum (i.e., the softmax predictions for the pixels within the ground-truth foreground are equal to 1) and all the positive values are omitted (i.e., the softmax predictions within the background are equal to zero). This means that the global optimum is reached for softmax predictions that correspond exactly to the ground truth, which confirms the meaningfulness of our boundary loss. It is also worth noticing that the gradient of our loss is ϕ_G multiplied the gradient of the softmax predictions. This results in negative factors for the pixels in G, which encourages s_θ to increase during SGD, with the magnitude (strength) of the factors depending on the distance between the pixel and the ground-truth boundary (the further the pixel from the boundary, the higher the magnitude of the factor). Positive factors for pixels within the background ($\Omega \backslash G$) encourage the softmax predictions to decrease.

As we will see in our experiments, it is important to use our boundary loss in conjunction with a regional loss for the following technical facts. As discussed earlier, the global optimum of our boundary loss corresponds to a strictly negative value, with the softmax probabilities yielding a non-empty foreground region. However, an empty foreground, with approximately null values of the softmax probabilities almost everywhere, corresponds to very low gradients. Therefore, this trivial solution is close to a local minimum or a saddle point. This is why we integrate our boundary loss with a regional loss: the regional loss guides training during the first epochs and avoids getting stuck in such trivial solutions. In the next section, we will discuss various scheduling strategies for updating the weight of the boundary loss during training, with the boundary loss becoming very dominant, almost acting alone, towards the end of the training process. It is also worth noting that this behaviour of boundary terms is conceptually similar to the behaviour of classical and popular contour-based energies for segmentation, e.g., level set Geodesic Active Contours (GAC) (Caselles et al., 1997) or discrete Markov Random Fields (MRFs) for boundary regularization and edge alignment (Boykov and Funka-Lea, 2006), which require additional regional terms to avoid trivial empty-region solutions.

3. Experiments

In this section, we perform two sets of experiments. First, we perform comprehensive evaluations demonstrating to positive effect of integrating our boundary loss with different regional losses $\mathcal{L}_R$. Then, we perform a study on the different strategies for selecting and scheduling weight α in (6), showing its impact on performances and good default values for new applications.

3.1. Datasets

To evaluate the proposed boundary loss, we selected two challenging brain lesion segmentation tasks, each corresponding to highly unbalanced classes.

ISLES: The training dataset provided by the ISLES organizers is composed of 94 ischemic stroke lesion multi-modal scans. In our experiments, we split this dataset into training and validation sets containing 74 and 20 examples, respectively. Each scan contains Diffusion maps (DWI) and Perfusion maps (CBF, MTT, CBV, Tmax and CTP source data), as well as the manual ground-truth segmentation. The spatial resolution goes from $0.8mm \times 0.8mm \times 4mm$ to $1mm \times 1mm \times 12mm$. More details can be found in the ISLES website[3].

WMH: The public dataset of the White Matter Hyperintensities (WMH)[4] MICCAI 2017 challenge contains 60 3D T1-weighted scans and 2D multi-slice FLAIR acquired from multiple vendors and scanners in three different hospitals. The spatial resolution goes from $0.95mm \times 0.95mm \times 3mm$ to $1.21mm \times 1mm \times 3mm$ for each volume. In addition, the ground truth for the 60 scans is provided. From the whole set, 50 scans were used for training, and the remaining 10 for validation.

3.2. Compared losses

As stated previously, our proposed boundary loss can be combined with any standard regional loss. In the following experiments, we evaluated different popular ones:

GDL (Sudre et al., 2017) We use the binary case of this loss, described in Eq. (1). This is also the baseline loss that we use for the experiments on the selection of α. An important advantage of this loss is that it is hyper-parameter free.

Distance weighted cross-entropy (Ronneberger et al., 2015) UNet original paper proposed this loss as a way to integrate spatial information during the training. It is a modified weighted cross-entropy loss, where the weight for each pixel depends both on the class distribution, and its distance to the two cells closest boundaries. We adapted it for our case, where we take into account only one distance:

$$\mathcal{L}_{\text{UNET}}(\theta) = -\int_C \int_\Omega u_c(p) \log s_\theta^c(p) dp dc,$$

where C is the set of classes and $s_\theta^c(p)$ are the network predictions for class c. $u_c(p)$ is defined as:

$$u_c(p) = g_c(p) \left[w_c + w_0 e^{\frac{-D_G(p)^2}{2\sigma^2}} \right],$$

where $w_c = \frac{\int_\Omega g_c(p)dp}{|\Omega|}$, and $w_0 = 10$ and $\sigma = 5$ are two hyper-parameters. We kept the paper's default values.

Focal loss (Lin et al., 2018) The idea of this loss is to give hard examples a more important weight:

$$\mathcal{L}_{\text{FOCAL}} = -\int_C \int_\Omega (1 - s_\theta^c(p))^\gamma g_c(p) \log s_\theta^c(p) dp dc,$$

with $\gamma = 2$ as default hyper-parameter. Therefore, during training, pixels correctly classified with a high confidence will have little to no influence.

Hausdorff loss (Karimi and Salcudean, 2019) This closely related loss is also designed to minimize some distance between the two boundaries, but through a different path. We refer to this loss as $\mathcal{L}_{HD}$.

$$\mathcal{L}_{HD} = \int_\Omega (g(p) - s_\theta(p))^2 (D_G(p)^\beta + D_S(p)^\beta) dp,$$

where D_S denotes the distance function from predicted boundary S, after thresholding s_θ. β is a hyper-parameter, which the authors of Karimi and Salcudean (2019) set to 2 following a grid search. Unlike our boundary loss, computing D_S cannot be done in a single

[3] http://www.isles-challenge.org

[4] http://wmh.isi.uu.nl

Table 1
Average DSC and HD95 values (and standard deviation over three independent runs) achieved on the validation subset. Best results highlighted in bold.

Loss	ISLES		WMH	
	DSC	HD95 (mm)	DSC	HD95 (mm)
$\mathcal{L}_B$	NA	NA	NA	NA
$\mathcal{L}_{HD}$	NA	NA	0.638 (NA)	4.578 (NA)
GDL	0.511 (0.016)	5.320 (1.742)	0.768 (0.051)	3.634 (2.570)
w/ $\mathcal{L}_B$ (2D)	0.644 (0.026)	4.795 (3.712)	0.793 (0.006)	2.039 (1.834)
w/ $\mathcal{L}_B$ (3D)	**0.659 (0.001)**	2.725 (2.196)	**0.818 (0.003)**	**1.702 (1.982)**
w/ $\mathcal{L}_{HD}$	0.582 (0.015)	4.126 (1.634)	0.805 (0.015)	2.151 (2.100)
UNet cross-entropy Ronneberger et al. (2015)	0.608 (0.025)	4.572 (0.675)	0.757 (0.015)	4.355 (3.388)
w/ $\mathcal{L}_B$ (2D)	0.631 (0.016)	5.961 (2.291)	0.756 (0.022)	2.887 (2.629)
Focal loss Lin et al. (2018)	0.631 (0.046)	4.989 (2.775)	0.808 (0.026)	1.816 (1.370)
w/ $\mathcal{L}_B$ (2D)	0.650 (0.019)	**1.770 (0.549)**	0.786 (0.031)	2.258 (2.513)

step before training. The distance needs to be re-computed at each epoch during training, for all the images. It also requires to store the whole volume Ω in memory, as we cannot compute the distance map for only a subset of Ω. These might be important computational and memory limitations, more so when dealing with large images, as is the case for 3D distance maps.

3.3. 2D and 3D distance maps

While the main experiments resort to a distance map computed from each individual 2D slice, we evaluate the proposed boundary loss with a distance map computed from the whole initial 3D segmentation mask. Eq. (5) enables us to have only a subset of Ω at each update, making it possible to use a 3D distance map with mini-batches of 2D slices.

3.4. Selection of α

We study several strategies for selecting α, and its effect on the performances. On top of a constant pre-selected α, we evaluated simple scheduling strategies to update it during the training.

***Constant* α** The simplest method would be to use a constant during the whole training, but this might require careful tuning of its value.

***Increase* α** We start with a low value of $\alpha > 0$, and increase it gradually at the end of each epoch. The weight of the regional loss $\mathcal{L}_R$ remains constant over time. At the end of the training, the two losses have the same weight.

***Rebalance* α** First we rewrite our combined loss as $(1-\alpha)\mathcal{L}_R + \alpha\mathcal{L}_B$. As for the increase strategy, we start with a low $\alpha > 0$, and increase it over time. In this way, we give more importance to the regional loss term at the beginning while gradually increasing the impact of the boundary loss term. Note that we make sure that the weight for $\mathcal{L}_R$ never reaches 0; the two losses are used at all times during training.

3.5. Implementation details

Data pre-processing While the scans are provided as 3D images, we process them as a stack of independent 2D images, which are fed into the network. In fact, the scans in some datasets, such as ISLES, contain between 2 and 16 slices, making them ill-suited for 3D convolutions in those cases. The scans were normalized between 0 and 1 before being saved as a set of 2D matrices, and re-scaled to 256 × 256 pixels if needed. When several modalities were available, all of them were concatenated before being used as input to the network. We did not use any data augmentation in our experiments.

Architecture and training We employed UNet (Ronneberger et al., 2015) as deep learning architecture in our experiments. To train our model, we employed Adam optimizer, with a learning rate of 0.001 and a batch size equal to 8. The learning rate is halved if the validation performances do not improve during 20 epochs. We did not use early stopping.

To compute the level set function ϕ_G in Eq. (5), we used standard SciPy functions[5]. Note that, for slices containing only the background region, we used a zero-distance map, assuming that the regional loss is sufficient in those cases. For the increase and rebalance α scheduling strategies, we start with $\alpha = 0.01$ and increase it by 0.01 at the end of each epoch. For all the experiments comparing different losses, we use the same rebalance strategy, with the same hyper-parameters.

In addition, we evaluated the performance when the boundary loss is the only objective, i.e., $\alpha = 0$.

For our implementation, we used PyTorch (Paszke et al., 2017), and ran the experiments on a machine equipped with an NVIDIA GTX 1080 Ti GPU with 11GBs of memory. Our code (data pre-processing, training and testing scripts) is publicly available[6]. As Karimi and Salcudean (2019) did not release their code, we relied on the re-implementation from Ma et al. (2020)[7].

Evaluation For evaluation purposes, we employ the common Dice Similarity Coefficient (DSC) and modified Hausdorff Distance[8] (HD95) metrics.

3.6. Results

3.6.1. Comparison of regional losses

In this section, we detail the results that we obtained when using different regional losses $\mathcal{L}_R$.

Quantative evaluation Table 1 reports the DSC and HD performances for our experiments using four different choices of $\mathcal{L}_R$, with each regional term used either alone or in conjunction with our boundary loss in Eq. (6), on the ISLES and WMH datasets. In most of the settings, adding the boundary loss during training improves the performances, as reflected in the significantly better DSC and HD values. For instance, on the ISLES segmentation task, adding the boundary loss yielded about 13% improvement in DSC over using Generalized Dice loss alone, and about 3% improvement over using UNet cross-entropy or focal loss alone. The discrepancy of the improvements the boundary loss brings might be due to the difference in the difficulty of the tackled tasks. The more difficult the tasks (i.e., when regional terms have difficulty achieving good performances), the larger the gain boundary loss brings (as it

[5] https://docs.scipy.org/doc/scipy-0.14.0/reference/generated/scipy.ndimage.morphology.distance_transform_edt.html

[6] https://github.com/LIVIAETS/surface-loss

[7] https://github.com/JunMa11/SegWithDistMap

[8] We report the 95th percentile distance value instead of the maximum-distance value.

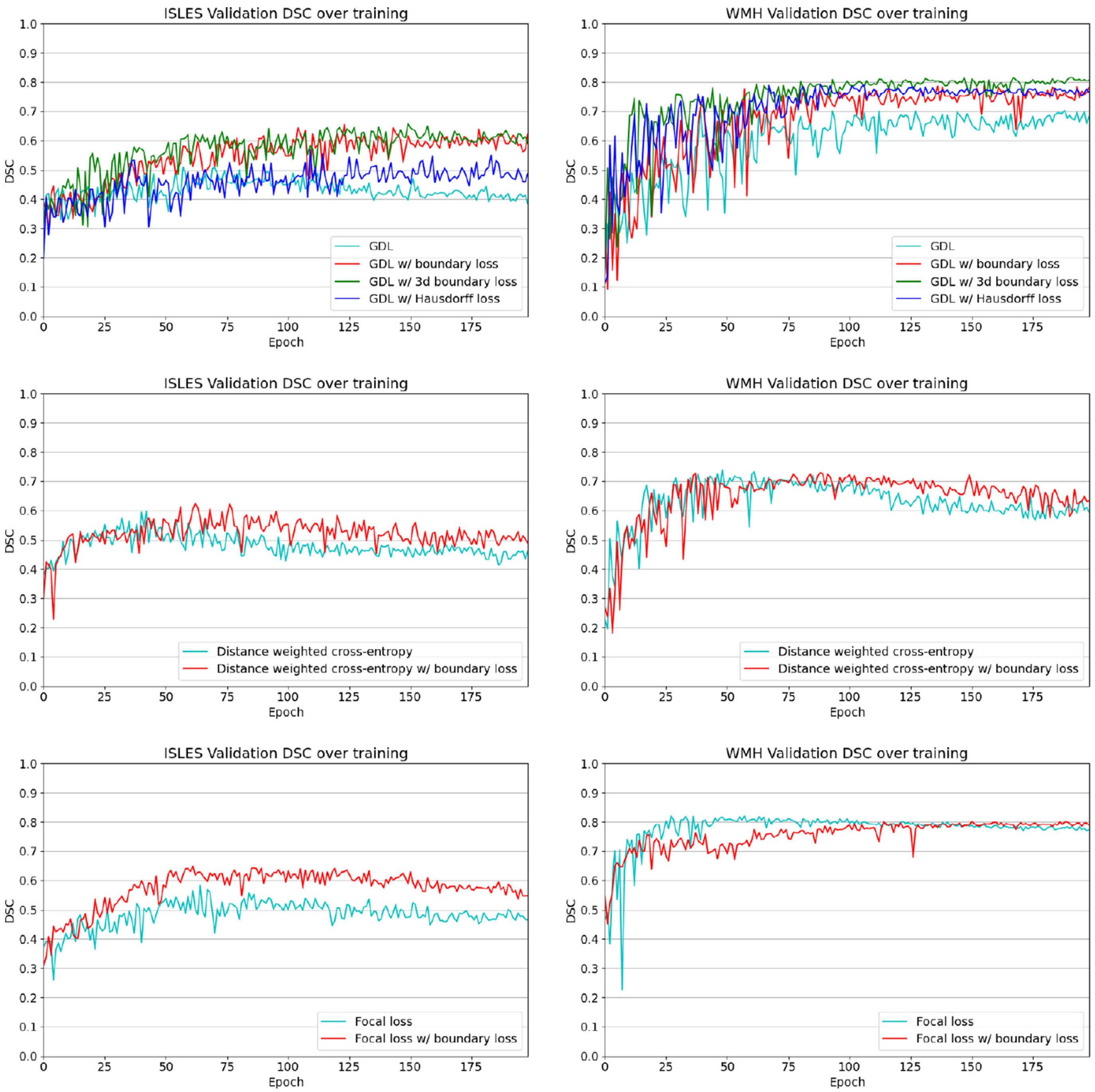


Fig. 3. Evolution of DSC values on validation subsets, for different base losses, on both ISLES and WMH. Best viewed in colors. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

complements regional information). GDL/ISLES is a noticeable case, where boundary loss corrected substantially the performance of the GDL regional loss, making it the winning competitor (although, without boundary information, it is the worse-performing regional loss).

The mixed results with the UNet cross-entropy (improvement on ISLES, but stall on WMH), and the difference on the HD95 metrics can potentially be explained by a toxic interplay between the two losses: both of them are trying to use the distance from the boundary information, potentially counter-acting each others, and introducing instability.

Computing the distance map from the 3D volume rather than from the 2D slices gives a small boost in performance (about 1% DSC), and is more noticeable on the training curve for WMH (Fig. 3). This difference could be explained by the spacing between the slices on the z axis: they are quite close (and correlated) in the case of WMH. However, in the case of ISLES, the big spacing (around 1cm) makes slices quite independent. Adding 3D information in this case is less helpful.

While the Hausdorff loss (Karimi and Salcudean, 2019) also improves the results over the GDL alone (around 7% on ISLES), its performance is not always at the same level as boundary loss (similar performance on WMH, but lower on ISLES). This is consistent with the findings of Ma et al. (2020), which found that the differences in performances are dataset dependent.

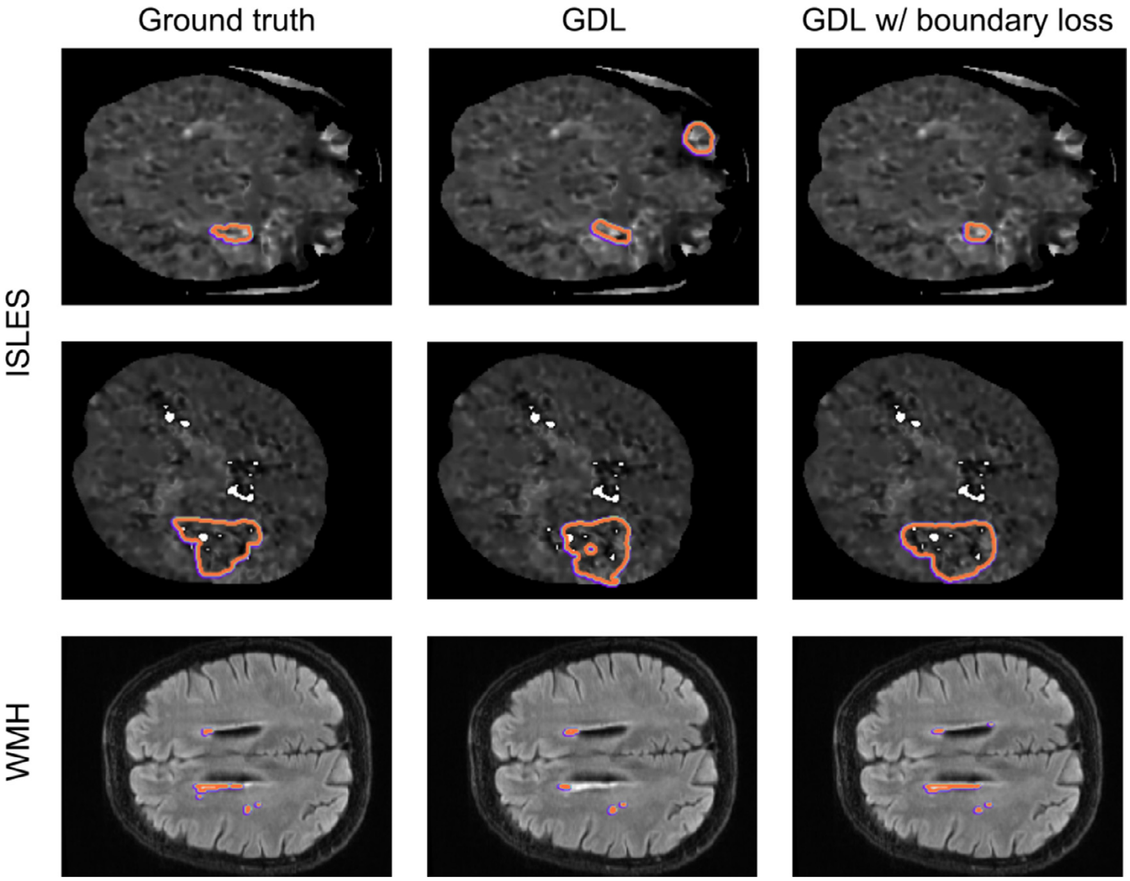


Fig. 4. Visual comparison on two different datasets from the validation set.

Using the boundary loss alone does not yield the same competitive results as a joint loss (i.e., boundary and region), making the network collapse quickly into empty foreground regions, i.e., softmax predictions close to zero[9]. We believe that this is due to the following technical facts. In theory, the global optimum of the boundary loss corresponds to a negative value, as a perfect overlap sums only over the negative values of the distance map. In this case, the softmax probabilities correspond to a non-empty foreground. However, an empty foreground (null values of the softmax probabilities almost everywhere) corresponds to low gradients. Therefore, this trivial solution is close a local minimum or a saddle point. This is not the case when we use our boundary loss in conjunction with a regional loss, which guides the training during the first epochs and avoids getting stuck in such a trivial solution. The scheduling method then increases the weight of the boundary loss, with the latter becoming very dominant towards the end of the training process. This behaviour of boundary terms is conceptually similar to the behaviour of classical and popular contour-based energies for level set segmentation, e.g., geodesic active contours (Caselles et al., 1997), which also require additional regional terms to avoid trivial solutions (i.e., empty foreground regions).

The learning curves depicted in Fig. 3 show the gap in performances between using a regional loss $\mathcal{L}_R$ alone and when augmented with our boundary loss, for different choices of $\mathcal{L}_R$. In most of the settings, the difference becomes significant at convergence. This behaviour is most visible when $\mathcal{L}_R = \mathcal{L}_{\text{GDL}}$, and is consistent for both metrics and both dataset, which clearly shows the benefits of employing the proposed boundary loss term.

Qualitative evaluation Qualitative results are depicted in Fig. 4. Inspecting these results visually, we can observe that there are two major types of improvements when employing the proposed boundary loss. First, as the methods based on DSC losses, such as GDL, do not use spatial information, prediction errors are treated equally. This means that the errors for pixels/voxels in an already detected object have the same importance as the errors produced in completely missed objects. On the contrary, as our boundary loss is based on the distance map from the ground-truth boundary ∂G, it will penalize much more such cases, helping to recover small and far regions. This effect is best illustrated in Figs. 1 and 4 (third row). False positives (first row in Fig. 4) will be far away from the closest foreground, getting a much higher penalty than with the GDL alone. This helps in reducing the number of false positives. Additional qualitative results for other base losses, and their combination with the proposed boundary loss, are depicted in Figs. 5 and 6. These figures also show failure cases (*last column*) of the boundary loss.

Computational complexity It is worth mentioning that, as the proposed boundary loss term involves an element-wise product between two matrices – i.e., the pre-computed level-set function ϕ_G and the softmax output $s_\theta(p)$ – the complexity that it adds is negligible as showed in Table 2. Contrary, the Hausdorff loss Karimi and Salcudean (2019) introduces around 10% of slowdown in the training process. This will be further magnified if we generalize to multi-class problems, where an individual distance map should be computed for each class.

[9] For this reason, we do not report metrics in this case, as it would be meaningless.

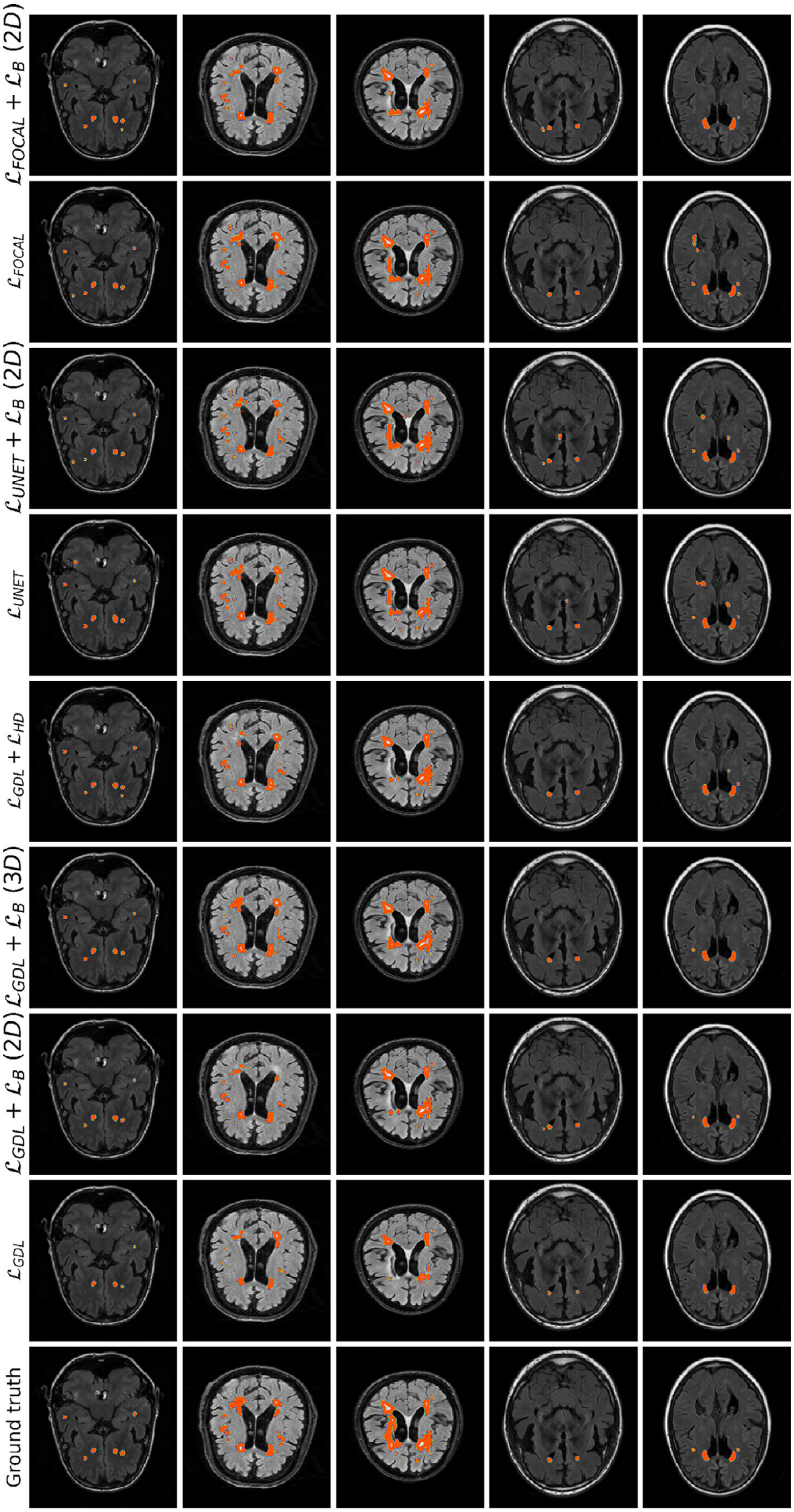

$\mathcal{L}_{FOCAL} + \mathcal{L}_B$ (2D)
$\mathcal{L}_{FOCAL}$
$\mathcal{L}_{UNET} + \mathcal{L}_B$ (2D)
$\mathcal{L}_{UNET}$
$\mathcal{L}_{GDL} + \mathcal{L}_{HD}$
$\mathcal{L}_{GDL} + \mathcal{L}_B$ (3D)
$\mathcal{L}_{GDL} + \mathcal{L}_B$ (2D)
$\mathcal{L}_{GDL}$
Ground truth

Fig. 5. Visual comparison on the WMH dataset for different training losses. The last column depicts a failure case, where the proposed loss does not enhance the regional loss performance. Best viewed in colors. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

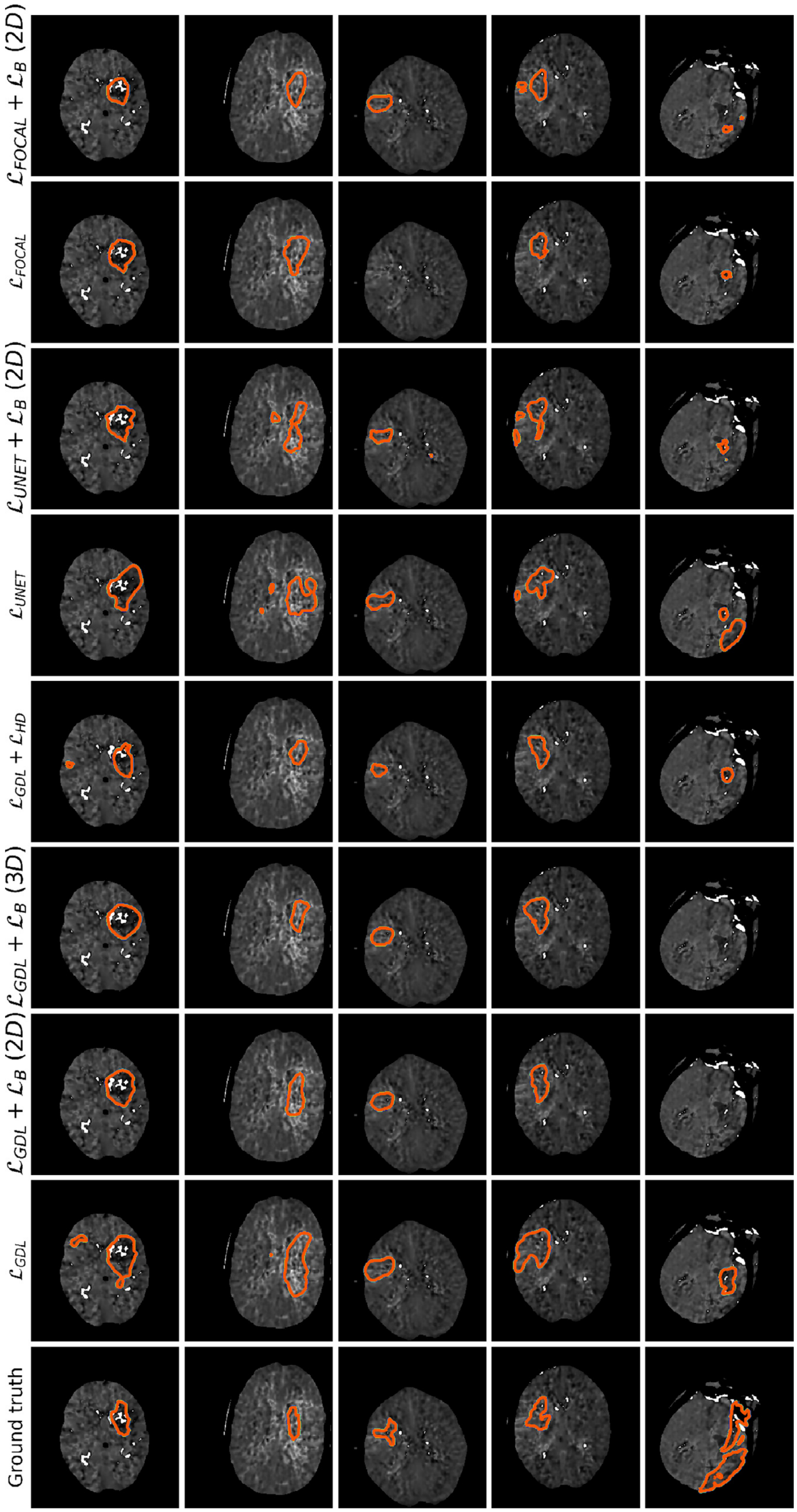

$\mathcal{L}_{FOCAL} + \mathcal{L}_B$ (2D)
$\mathcal{L}_{FOCAL}$
$\mathcal{L}_{UNET} + \mathcal{L}_B$ (2D)
$\mathcal{L}_{UNET}$
$\mathcal{L}_{GDL} + \mathcal{L}_{HD}$
$\mathcal{L}_{GDL} + \mathcal{L}_B$ (3D)
$\mathcal{L}_{GDL} + \mathcal{L}_B$ (2D)
$\mathcal{L}_{GDL}$
Ground truth

Fig. 6. Visual comparison on the ISLES dataset for different training losses. The last column depicts a failure case, where the proposed loss does not enhance the regional loss performance. Best viewed in colors. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

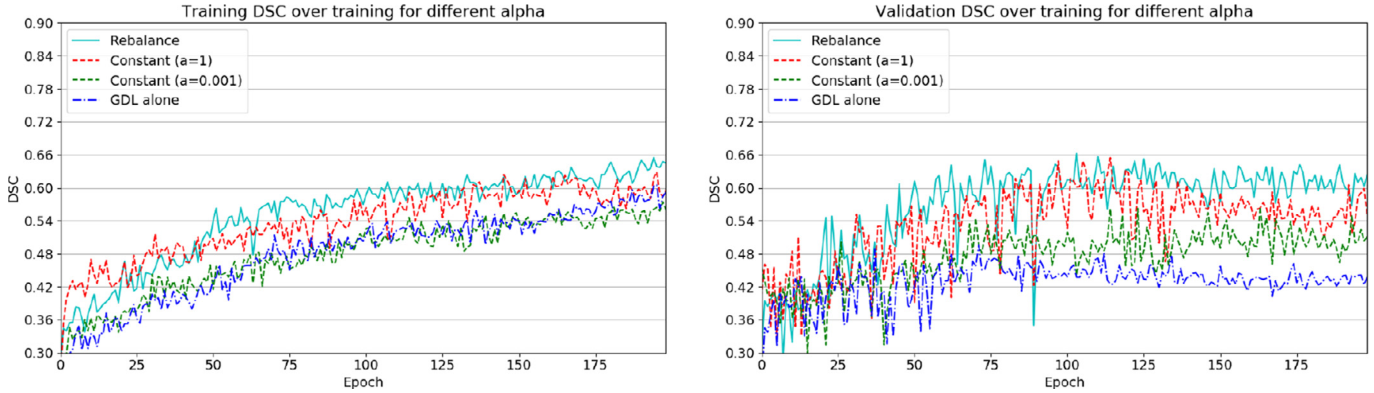


Fig. 7. Comparison of the training and validation DSC curves for different α selection strategies. For readability, not all settings from Table 3 have been included. Best viewed in colors. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 2
Training time required by different losses. We report the average and standard deviation batch time in seconds for each method.

Loss	Time (s) per batch	
	ISLES (Batch size = 4)	WMH (Batch size = 8)
GDL	0.187 (0.129)	0.345 (0.132)
w/ $\mathcal{L}_B$	0.190 (0.128)	0.345 (0.129)
w/ $\mathcal{L}_{HD}$	0.210 (0.108)	0.392 (0.092)

Table 3
Results on ISLES validation set for different α.

Strategy		ISLES	
	DSC	HD95	
GDL only		0.511 (0.016)	5.320 (1.742)
Constant α	$\alpha = 0.001$	0.545 (0.020)	4.778 (1.546)
	$\alpha = 0.01$	0.566 (0.019)	5.052 (1.395)
	$\alpha = 0.05$	0.606 (0.015)	5.326 (1.712)
	$\alpha = 0.1$	0.605 (0.010)	5.762 (1.782)
	$\alpha = 0.5$	0.604 (0.006)	9.234 (10.463)
	$\alpha = 1$	*0.628* (0.023)	*2.462* (0.706)
	$\alpha = 1.5$	0.565 (0.074)	3.335 (1.164)
	$\alpha = 2$	0.549 (0.084)	20.275 (16.603)
Increase α		*0.622* (0.004)	*4.952* (1.773)
Rebalance α		***0.644*** (0.026)	***4.795*****(3.712)**

3.6.2. Selection of α

Table 3 reports the performances of the proposed approach on the ISLES segmentation task for different α values and scheduling techniques. Fig. 3 shows a subset of the learning curves related to α selection strategies in Table 3. This is an indication that, while our boundary loss can benefit from a tuned balance between the two losses, even a sub-optimal α can already provide improvement over the regional loss alone. Observe that increasing the weight of constant α yields better performances, up to a certain value, with the performances decreasing starting from $\alpha = 1.5$. With $\alpha = 2$, the performance is similar to a network trained with the boundary loss alone. In contrast, using any of the two proposed scheduling strategies (increasing α or re-balancing) yields better results than any constant α, without having to explore many configurations.

From the learning curves (Fig. 7), we can notice that the GDL alone and the GDL with a small constant $\alpha = 0.001$ have a similar training DSC over time, but that their validation DSC are significantly different. A similar behaviour can be observed by examining the results with constant $\alpha = 1$ and the rebalanced α: while the rebalancing training DSC is slightly higher during the whole training, the validation DSC becomes significantly better around half the training time, where the high constant α performances starts decreasing.

The rebalancing strategy was used in all other experiments, and as showed in Table 1, proved to be a good default strategy to integrate the boundary loss with another regional loss.

4. Conclusion and future works

We proposed a boundary loss term that can be easily combined with any standard regional loss, to tackle segmentation tasks in highly unbalanced scenarios. Furthermore, the proposed term can be implemented with any existing deep network architecture and for any N-D segmentation problem. Our experiments on two challenging and highly unbalanced datasets demonstrated the benefits of including our boundary loss during training. It consistently improved the performances, and by a large margin on one data set, with enhanced training stability.

In this work, we evaluated the proposed boundary loss in the context of class imbalance. However, there are other interesting avenues for extending and evaluating our approach. For instance, our boundary loss has a spatial regularization effect because it is based on distance-to-boundary information. In particular, we observed experimentally that it yield contours, which are, typically, smoother than those obtained with regional losses. Focused on the important problem of unbalanced segmentation, our experiments did not fully investigate the benefits of such a spatial regularization. An interesting future research avenue will be to explore such a regularization effect in applications with challenging imaging noise, which may prevent regional losses from generating smooth contours, e.g., ultrasound imaging. Another limitation of our formulation and experiments is that they were limited to binary (two-region) segmentation problems. It will be interesting to investigate extensions of boundary loss to the multi-region scenario, with competing distance maps from multiple structures and various/complex topological constraints (e.g., one structure fully included within another).

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work is supported by the National Science and Engineering Research Council of Canada (NSERC), Discovery Grants (DG) program, and by the ETS Research Chair on Artificial Intelligence in Medical Imaging.

References

Abraham, N., Khan, N. M., 2018. A novel focal tversky loss function with improved attention U-Net for lesion segmentation. arXiv preprint arXiv:1810.07842.

Boykov, Y., Funka-Lea, G., 2006. Graph cuts and efficient n-d image segmentation. Int. J. Comput. Vis. 70, 109–131.

Boykov, Y., Kolmogorov, V., Cremers, D., Delong, A., 2006. An integral solution to surface evolution PDEs via geo-cuts. In: European Conference on Computer Vision. Springer, pp. 409–422.

Brosch, T., Yoo, Y., Tang, L.Y., Li, D.K., Traboulsee, A., Tam, R., 2015. Deep convolutional encoder networks for multiple sclerosis lesion segmentation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, pp. 3–11.

Caselles, V., Kimmel, R., Sapiro, G., 1997. Geodesic active contours. Int. J. Comput. Vis. 22, 61–79.

Dolz, J., Desrosiers, C., Ben Ayed, I., 2018. 3D fully convolutional networks for subcortical segmentation in MRI: a large-scale study. NeuroImage 170, 456–470.

Havaei, M., Davy, A., Warde-Farley, D., Biard, A., Courville, A., Bengio, Y., Pal, C., Jodoin, P.-M., Larochelle, H., 2017. Brain tumor segmentation with deep neural networks. Med. Image Anal. 35, 18–31.

Kamnitsas, K., Ledig, C., Newcombe, V.F., Simpson, J.P., Kane, A.D., Menon, D.K., Rueckert, D., Glocker, B., 2017. Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation. Med. Image Anal. 36, 61–78.

Karimi, D., Salcudean, S.E., 2019. Reducing the hausdorff distance in medical image segmentation with convolutional neural networks. IEEE Trans. Med. Imaging.

Ker, J., Wang, L., Rao, J., Lim, T., 2018. Deep learning applications in medical image analysis. IEEE Access 6, 9375–9389.

Lin, T.-Y., Goyal, P., Girshick, R., He, K., Dollár, P., 2018. Focal loss for dense object detection. IEEE Trans. Pattern Anal. Mach.Intell..

Litjens, G., Kooi, T., Bejnordi, B.E., Setio, A.A.A., Ciompi, F., Ghafoorian, M., Van Der Laak, J.A., Van Ginneken, B., Sánchez, C.I., 2017. A survey on deep learning in medical image analysis. Med. Image Anal. 42, 60–88.

Long, J., Shelhamer, E., Darrell, T., 2015. Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3431–3440.

Ma, J., Wei, Z., Zhang, Y., Wang, Y., Lv, R., Zhu, C., Chen, G., Liu, J., Peng, C., Wang, L., et al., 2020. How distance transform maps boost segmentation cnns: An empirical study. Medical Imaging with Deep Learning.

Milletari, F., Ahmadi, S.-A., Kroll, C., Plate, A., Rozanski, V., Maiostre, J., Levin, J., Dietrich, O., Ertl-Wagner, B., Bötzel, K., et al., 2017. Hough-CNN: deep learning for segmentation of deep brain regions in MRI and ultrasound. Comput. Vis. Image Underst. 164, 92–102.

Milletari, F., Navab, N., Ahmadi, S.-A., 2016. V-Net: Fully convolutional neural networks for volumetric medical image segmentation. In: 3D Vision (3DV), 2016 Fourth International Conference on. IEEE, pp. 565–571.

Mitiche, A., Ben Ayed, I., 2011. Variational and Level Set Methods in Image Segmentation. Springer.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A., 2017. Automatic differentiation in pytorch.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-Net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, pp. 234–241.

Salehi, S.S.M., Erdogmus, D., Gholipour, A., 2017. Tversky loss function for image segmentation using 3D fully convolutional deep networks. In: International Workshop on Machine Learning in Medical Imaging. Springer, pp. 379–387.

Shen, D., Wu, G., Suk, H.-I., 2017. Deep learning in medical image analysis. Ann. Rev. Biomed. Eng. 19, 221–248.

Sudre, C.H., Li, W., Vercauteren, T., Ourselin, S., Cardoso, M.J., 2017. Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. In: Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support. Springer, pp. 240–248.

Valverde, S., Cabezas, M., Roura, E., González-Villà, S., Pareto, D., Vilanova, J.C., Ramio-Torrenta, L., Rovira, À., Oliver, A., Lladó, X., 2017. Improving automated multiple sclerosis lesion segmentation with a cascaded 3D convolutional neural network approach. NeuroImage 155, 159–168.

Wong, K.C., Moradi, M., Tang, H., Syeda-Mahmood, T., 2018. 3D segmentation with exponential logarithmic loss for highly unbalanced object sizes. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, pp. 612–619.

Yu, L., Yang, X., Chen, H., Qin, J., Heng, P.-A., 2017. Volumetric convnets with mixed residual connections for automated prostate segmentation from 3D MR images.. In: AAAI, pp. 66–72.

Zhu, W., Huang, Y., Zeng, L., Chen, X., Liu, Y., Qian, Z., Du, N., Fan, W., Xie, X., 2019. Anatomynet: deep learning for fast and fully automated whole-volume segmentation of head and neck anatomy. Med. Phys. 46 (2), 576–589.