MATHÉMATIQUES
VISION
APPRENTISSAGE

" Boundary Loss for highly unbalanced segmentation "

Kervadec et al. (2020)

Realized By:

Eya Ghamgui

Siwar Mhadhbi

Hamza Meddeb

Academic Year: 2021 - 2022

Contents

Introduction

The paper entitled 'Boundary loss for highly unbalanced segmentation' introduces a new loss function that boosts the performance of image segmentation for some challenging unbalanced datasets. This loss was tested on binary pixel classification. In other words, images are segmented to foreground and background. Besides, datasets are highly unbalanced which indicates that the foreground objects are smaller than the background. Existing loss functions for CNN networks, for example Dice or cross entropy, are based on summation over the segmentation regions. For the case of highly unbalanced segmentations the loss associated to the background have values higher than the loss associated to the foreground object. Therefore, small foreground regions are poorly detected by models trained on such loss functions.

In this paper, authors introduce a new loss metric that is combined with regional loss functions such the ones we mentioned previously to obtain better segmentation performance. This loss is the non-symmetric L_2 distance on the space of contours. The calculation of this lost is based on local differential computations over contour points. Therefore, adding it to regional loss functions to train Deep CNN is difficult to achieve. However, authors of the paper propose an approximation of this distance using the regional softmax probability output of the network, which can be easily combined with standard regional losses and implemented with any existing deep network architecture.

1 Expression of the proposed boundary loss

Before we demonstrate that the loss boundary can be expressed as a function softmax probability output of any CNN architecture, let's explain some annotations used in the paper.

- p: denotes a pixel in a 2D image or a voxel in the case of 3D images.
- $g(p)$: denotes the ground-truth labelling of the pixel p. This function returns 1 if the pixel belongs to the foreground object, otherwise it takes 0.
- $s(p)$: denotes the binary indicator of region S ,which is the segmentation output of the CNN. This function takes the value 1 if the pixel p belongs to S, 0 otherwise.
- S_θ: denotes the softmax probability output of the deep CNN.
- ∂G: denotes the representation of the boundary ground-truth. Here, it is the set of pixels that have a spatial neighbor in background.
- ∂S_θ: denotes the representation of the boundary of the segmentation region defined by the CNN output.

A distance between two contours ∂G, ∂S is defined as below:

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

where $y_{\partial S(p)}$ denotes the projection of the point p that belongs to the contour ∂G on the contour ∂S.

This point is obtained by the intersection of the normal vector to the contour ∂G at the point p and the contour ∂S. This distance is approximated by the following equation:

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

where ΔS denotes the region between the two contours ∂G and ∂S and $D_G(q)$ is the distance between a point q in the region ΔS and the contour ∂G. To obtain this approximation, we will use the following proof:

$$\int_{p}^{y_{\partial S}(p)} 2D_G(q) dq = \int_{0}^{\|y_{\partial S}(p)-p\|} 2D_G dD_G = \|y_{\partial S}(p) - p\|^2$$

The equation above shows that the L_2 distance between a point p of the contour ∂G and the contour ∂S is equal to the integral of the distance map $2D_G(q)$ over the normal segment connecting p and its projection on the contour ∂S.
This approximation means that instead of calculating the L_2 distance between the set of points of the contour ∂G and its projections on the contour ∂S, we calculate the integral of the distance map $2D_G(q)$ over all the points that are in the region ΔS.
Now, we will express the integral over the region ΔS with respect to the integral over the regions G and S. This is done thanks to the level set representation of boundary ∂G which is defined as below:

$$\phi_G(q) = D_G(q) \text{ if } q \in G \text{ and } \Phi_G(q) = D_G(q) \text{ otherwise}$$

Given this function, we can express the L_2 distance between the contours ∂G and ∂S as following:

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

Finally, we replace the binary variable $s(q)$ with the softmax probability output of the CNN.

$$\boxed{L_B(\theta) = \int_\Omega \phi_G(q) s_\theta(q) dq}$$

In the expression above the second term disappears. In fact, it does not depend on the output of the CNN. Thus, we can drop it since its gradient with respect to the parameters of our architecture is equal to zero.

After expressing our boundary loss with respect to the output of any CNN, we will combine it with existing regional loss functions using different techniques that will be more explained in the following section. Also experiments will be represented to compare the effect of adding our loss to other regional loss functions.

2 Experimental discussion

2.1 Chosen datasets

The article selects two challenging brain lesion segmentation datasets that correspond to highly unbalanced classes; Ischemic Stroke Lesion (ISLES) and White Matter Hyperintensities (WMH) benchmark datasets. These datasets contain only 94 ischemic stroke lesion multi-modal scans and only 60 3D T1-weighted scans and 2D multi-slice FLAIR respectively.

2.2 Implementation details

2.2.1 Data preprocessing

As training data, Kervadec et al. (2021) used not only provided 3D scans but also they processed them as a stack of independent 2D images to be fed into the network. This is pertinent as the scans in ISLES dataset contain between 2 and 16 slices, making them ill-suited for 3D convolutions. These aforementioned datasets are relevant to the task of highly unbalanced segmentation. Nonetheless, they are very small and need to be augmented in a pre-processing step in order to train conveniently deep neural networks. This task of **data augmentation** hasn't been treated in this article, yet it has been widely used in many tasks and particularly in medical image segmentation as suggested by Eaton et al. (2018) who implemented a whole data augmentation technique that increased the performance of medical image segmentation.

2.2.2 Architecture and training

The article selects U-Net architecture (Ronneberger et al., 2015) for this task of segmentation with Adam optimizer parametrized by a batch size of 8 and a learning rate of 0.001 halved if the validation performances do not improve during 20 epochs.

2.3 Experiments

In order to evaluate the impact of integrating the proposed boundary loss with different regional losses, Eaton et al. employ the common Dice Similarity Coefficient (DSC) and the modified Hausdorff Distance (HD95) metrics. Their procedure consists in evaluating each time one regional loss (among GDL, U-Net CE, and focal loss) combined with one boundary loss (among HD, and the proposed Boundary loss).

2.3.1 Quantitative evaluation

As shown by Table 1, adding the boundary loss during training improves the performances, in most of the settings. This is most visible when the regional loss is the GDL, and is consistent for both metrics and both datasets. In particular, on the ISLES segmentation task, the strategy yielded about 13% improvement in DSC over using GDL alone, and about 3% improvement over using UNet cross-entropy or focal loss alone, which clearly shows the benefits of employing the proposed boundary loss term.

2.3.2 Qualitative evaluation

From figure 3, we can observe that there are two major types of improvements when employing the proposed boundary loss. First, it helps reduce the number of false positives as shown in the first row. The GDL detects a wrong similar region that is far from the target one. Whereas the boundary loss detects only regions that are included in the target one. This is indeed explained by the fact that the GDL does not use spatial information whereas the introduced boundary loss is based on the distance map from the ground-truth boundary. Moreover, the proposed boundary loss helps recover small regions that are wrongly classified as negative regions by the GDL alone as we can observe in the second and third rows.

2.4 Interpretation and discussion of the weighting factor

The combination of regional and boundary losses is based on the factor α as shows the equation:

$$L_R(\theta) + \alpha L_B(\theta)$$

The factor goes from 0.001 to 2 which would involve both cases of boundary or regional dominant loss. In the underlying article, different strategies were performed to select and schedule α. First strategy is to set α constant during training, but with a proper fine-tuned value. Second strategy is to increase α gradually over time which gives more importance to the regional loss term at the beginning while gradually increasing the impact of the boundary loss term until dominating at the end. As for the third strategy, it involves the rebalance α, where the total loss is rewritten as $(1-\alpha)L_R + \alpha L_B$. The results of these experiments are shown in table 3. We can observe that increasing the weight of constant α yields better performances, up to a certain value $\alpha = 1$, with the performances decreasing starting from $\alpha = 1.5$. However, the re-balancing technique yields the best results.

2.5 New proposition for the combined loss

In this section, we question the formula of the training loss. The article only inspected one single regional loss against one single boundary loss. However, in recent literature, authors tend to combine more losses to improve further the results. A very recent study from 2021, by Ma et al., entitled "Loss odyssey in medical image segmentation", highlighted the improvement of results by combining losses.

Inspired by this article, we propose below a new theoretical formula for the combined loss where the first term becomes a block of regional losses, precisely a combination of Generalized dice loss and weighted cross entropy which is denoted as **Combo loss** in the mentioned article, also the block of boundary loss becomes a combination of the proposed loss and the Hausdorff distance.

$$\underbrace{(GDL + \beta_1\, wCE)}_{L'_R} + \alpha \underbrace{(L_B + \beta_2\, L_{HD})}_{L'_B}$$

This intuition of combining blocks of losses, based on recent literature, is very likely to help increase further the segmentation results.

3 Advantages of Boundary Loss

3.1 Easily combined with standard regional losses in deep network architecture for N-D segmentation

Representing the boundary points corresponding to the softmax regional outputs of a convolutional neural network is very challenging. Therefore, the boundary loss introduced in this paper is considered as an interesting result. This loss is an integral approach for computing boundary variations. In practice, it is computed as a sum of linear functions of the regional softmax probability outputs of the network. This formulation allows it to be very simply integrated into regional neural networks and easily combined with other standard regional losses.

3.2 Improving the performance of the unbalanced segmentation

Boundary loss is shown to have many advantages when combined with regional loss functions. It aims at recovering very small and far regions that are poorly segmented. When regional losses have difficulty in converging to an optimal solution, boundary loss solves this issue by adding additional information as discussed in the previous Experimental section. Moreover, another approach used when calculating the distance is that instead of using a 2D slice, we use a 3D slice to calculate the distance map. This method improved the performance by about 1% of the DSC. Added to that, this approach improved the model learning for the WMH dataset. This can be explained by the high correlation between the slices.

3.3 Computationally efficient loss

The boundary loss is defined as an element-wise product between two matrices (the pre-computed level set function $\Phi_G(q)$ and the softmax regional outputs s(q)). This formulation makes the added complexity very negligeable. As shown in the experiments, the computational time increased by 0.003 sper patch for the ISLES dataset and remains equal for the WMH dataset.

3.4 Outperforming the Hausdorff loss

Distances in the boundary loss are pre-computed before training. However, for Hausdorff Loss, these distances are recomputed at each epoch during the training of the model for all images, and then stored in memory. This is a memory and computational time limitation for large images, 3D distances maps and multi-class problems. This loss showed a 10% slowdown in learning. Furthermore, the boundary loss validation performance is quite similar for the WMH dataset and much better on the ISLES dataset when combined with the GDL loss.

3.5 Improving stability

The boundary loss has improved the stability of the GDL. When we add this loss to the boundary loss, the validation performances of the model become more stable at the end of the training. Thus, it is easier to tune the model parameters and choose the right model for generalization.

4 Limitations of Boundary Loss

4.1 Limitation to two regional segmentation

In this study, the authors limited their work to the binary case of segmentation. All the related work and performances are carried out on images containing a single object. In the last section, we will discuss in detail their future perspective for the multi-class case.

4.2 Conjunction with regional losses

This boundary loss does not seem to work alone. It should be associated with other regional losses. In fact, by minimizing the loss function, we solve a gradient descent algorithm that is based on updating steps using the computation of the gradient. A very small value of the gradient corresponds to zero values of the softmax probabilities almost everywhere, causing the network to collapse rapidly in empty foreground regions. This case, according to authors, is called a local minimum or a saddle point. This is why, regional losses are useful to guide the convergence of the loss and prevent it from getting stuck in these points.

4.3 Qualitative limitations

This method, although very efficient and improving the segmentation of highly unbalanced cases, still has drawbacks and poor segmentation quality on some examples. The figure 1 shows an example of images where the addition of boundary loss does not improve performance. Here, the algorithm converges to an empty solution which corresponds, according to the authors' intuition, to a local minimum or a saddle point. We can say that the boundary loss term is dominant here and that adding the GDL does not help the algorithm get stuck into one of these points.

5 Future Perspectives

5.1 Multi-class case

In this paper, the authors proposed the multi-class segmentation case as an extension that can be added to their study. We found that they have recently finished adapting their method to handle the multi-class case. They have indicated that this is a simple change in the loss formula. It is based on calculating the distance for each class then combining the results. The figure 2 depicts dataset with four classes. They showed that the boundary loss can be used alone in the multi-class case and that it is able to give very good performances close to those obtained using the GDL and CE losses.

5.2 Regularization

The segmentation of the boundary loss is smoother than that of the other regional losses. This spatial regularization is not studied in this paper. It may be studied in the future and incorporated into other applications that require smoother contours, such as applications with challenging imaging noise, which may prevent regional losses from generating smooth contours. For instance, this can be used for Ultrasound imaging.

5.3 Many more potential applications

There are several other applications that are able to benefit from this type of loss, for instance:

- Imagery Semantic Segmentation: "Boundary Loss for Remote Sensing Imagery Semantic Segmentation" Bokhovkin et al. (2019)
- Vehicle segmentation: "Vehicle lane markings segmentation and keypoint determination using deep convolutional neural networks" Raja et al. (2021)
- Object identification: "A global method to identify trees outside of closed-canopy forests with medium-resolution satellite imagery" Brandt et al. (2020)

Conclusion

Image segmentation plays an important role in the field of image processing, helping to understand images and recognize objects. However, most existing methods do not seem to be able to solve the problem of highly unbalanced foreground/background regions. This paper discusses a new boundary loss that incorporates boundary information. It has proven to be a very powerful improvement for highly unbalanced segmentation problems. Its main strength is its ability to be easily combined with any standard regional loss and then used in deep neural networks.

In this report, we started by presenting the previous attempts to improve this unbalance problem in segmentation. After that, we went into theoretical details about different losses used in this study, from known losses to the new boundary loss. In a second part, we discussed the dataset and data pre-processing. We showed also, the modeling part and the qualitative and quantitative results. Then, we tackled the advantages and limitations of this new loss. We found that this loss with conjunction with regional losses improves the performance of the model as well as its training stability. However, it presents limitations. Finally, we end up with some future perspectives of this loss.

Appendix

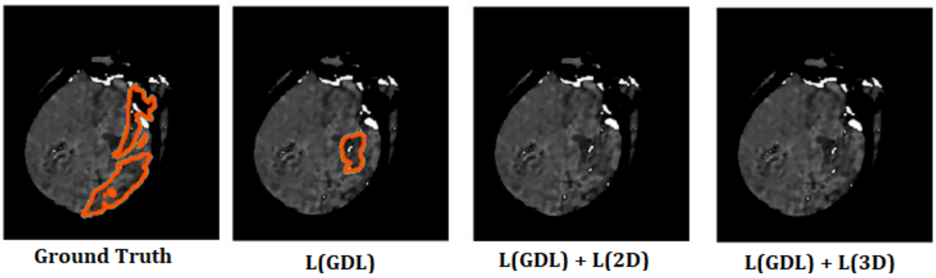

Figure 1: Example of miss-segmented image when using the boundary loss

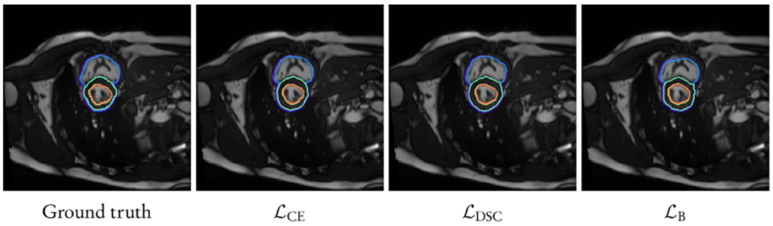

Figure 2: Results on the ACDC, a 4-classes dataset, when training with different losses. Unlike in the binary case, here the boundary loss is able to learn to segment the object properly.

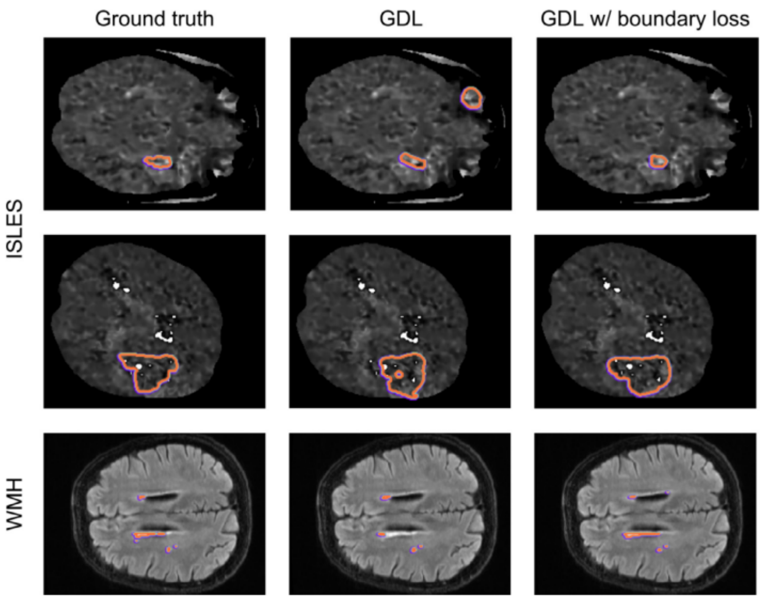

Figure 3: Visual comparison on two different datasets from the validation set

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)

13%

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)**

Bibliography

[1] https://arxiv.org/pdf/1812.07032.pdf

[2] https://deepai.org/publication/boundary-loss-for-highly-unbalanced-segmentation

[3] https://www.frontiersin.org/articles/10.3389/fnins.2019.00353/full

[4] https://github.com/LIVIAETS/boundary-loss

[5] https://openreview.net/pdf?id=rkBBChjiG

[6] https://www.sciencedirect.com/science/article/pii/S1361841521000815