Medical Image Analysis Project

Mahmoud Hegazy
IP-Paris
mahmoud.hegazy@ip-paris.fr

Inès Vati
MVA ENS Paris-Saclay
ines.vati@eleves.enpc.fr

1 Introduction

Inès Vati and *Mahmoud Hegazy*

In the area of Convolution Neural Network (CNN) based segmentation, widely employed loss functions, such as Dice or cross-entropy, have been instrumental in delineating segmentation regions. However, a critical drawback emerges when tackling highly unbalanced segmentation. As regions are highly uneven, the values obtained from regional summations vary drastically across classes, spanning several orders of magnitude. This inherent imbalance poses a substantial challenge to training performance and stability.

To address this issue, Kervadec et al. propose a boundary-based loss function in their article. Departing from conventional methodologies, this loss function operates on the space of contours rather than regions. The key advantage lies in its ability to mitigate the difficulty of highly unbalanced regions by penalising mistakes proportionally to the distance between the mistake and the interfaces of regions. This circumvents the pitfalls of unbalanced sums over the regions themselves. Moreover, boundary loss not only addresses imbalance issues but also complements regional information and can be seamlessly integrated with standard regional loss. In addition, it can be implemented with any existing deep network architecture designed for segmentation.

From this perspective, this article introduces the paradigm of boundary-based loss functions for CNN segmentation. We now first introduce the mathematical and notational framework needed to proceed with defining boundary loss. Then, we proceed to discuss other segmentation loss functions and some of the related work. Finally, we will evaluate the method on two different datasets, each presenting a unique set of difficulties.

Problem Statement: We consider a segmentation function f_θ parameterized by θ such that $f_\theta : \mathbb{R}^s \times \mathbb{R}^c \to \mathbb{R}^s \times \Delta^{k-1}$ taking in an image and outputting softmax probabilities over k classes. Here, s represents the dimension of the underlying images. Using $\mathbb{R}^{m\times n}$ to denote $\mathbb{R}^m \times \mathbb{R}^n$, and with the slight abuse of notation, s can be taken as a placeholder for $h \times w$ for 2D or $h \times w \times d$ for 3D tensors. c represents the channel (3 for RGB, 2 for MRI, 1 for grayscale) and $\Delta^{k-1} \subset \mathbb{R}^k$ is the standard simplex.

In addition, let $\langle \cdot, \cdot \rangle$ be the standard inner product, $\| \cdot \|_F$ the Frobenius norm, $\boldsymbol{a}$ be the all vector/matrix/tensor in $\mathbb{R}^d$ of repeated a. For $X \in \mathbb{R}^{m\times n\times p}$, X_i is the component at index $i \in [m] \times [n] \times [p]$

For notational simplicity, we restrict our attention to the case where $k = 2$ for most of this report. The extension of the boundary loss to multiclass segmentation will be later introduced in section 2. For an image X, we denote the ground truth by $g : \mathbb{R}^d \to \{0, 1\}^d$ such that $g(X)_i = 1$ if X_i is in class 1. Also define $c_\theta(X) = f_\theta(X)_{\cdot,1}$ i.e. for an image X, $c_\theta(X)$ is the softmax probability that a pixel/voxel belongs to class 1.

A note on notation: The original article opted mostly for integral-based notation to motivate and define the boundary loss function. We found that it obscured the computational ease of this loss. Thus, we opted for a more vectorised notation, where most operations are written as matrix products

and element-wise operations, that are extremely ubiquitous, fast, and scalable in modern scientific computing.

2 Boundary Loss

Mahmoud Hegazy

At its core, boundary loss is an approach of penalizing mistakes w.r.t to their distance from the boundary between two regions. In practice, this reduces to a voxel-wise multiplication between the network predictions and a pre-computed distance map.

To define the boundary loss, we first introduce the necessary sets and notation. Let I be the set of all indices of an image X. Then, define $G_0(X) \coloneqq \{i|g(X)_i = 0\}$, $G_1(X) \coloneqq I/G_0(X)$, $C_0(X) \coloneqq \{i|c_\theta(X)_i \leq \alpha\}$, and $C_1(X) = I/C_0(X)$ where α is a hyperparameter by the user (typically $1/2$). Define the boundary set $\partial G(X) \coloneqq \{d \in G_1(X)|N(d) \cap G_0(X) \neq \emptyset\}$, where $N(d)$ are the neighbouring indices to d.

Using the boundary set ∂G, it is now possible to define a distance between any point and ∂G. For example, consider a 3D image of width w, height h, and depth d voxels. Then, $I = [w] \times [h] \times [d]$ and for any $x, y \in I$ we can define a Euclidean-like distance on the grid

$$(x, y) = \sqrt{\alpha_1^2(x_1 - y_1)^2 + \alpha_2^2(x_2 - y_2)^2 + \alpha_3^2(x_3 - y_3)^2}, \tag{1}$$

with α_1, α_2, and α_3 representing sampling constants i.e. estimations on how far two voxels are in real life. We can now define the projection onto ∂G and a corresponding distance using d w.r.t to the index as follows

$$\Pi_{\partial G(X)}(i) \coloneqq \text{argmin}_{j \in \partial G(X)}\, d(i, j) \tag{2}$$

Dropping the notational dependence on X, we now define the distance-map function $\phi : \mathbb{R}^d \to \mathbb{R}^d$ as

$$\phi(X)_i = \begin{cases} -d(i, \Pi_{\partial G}(i)) & \text{for } i \in G(X) \\ d(i, \Pi_{\partial G}(i)) & \text{otherwise} \end{cases}. \tag{3}$$

To illustrate the distance map, we refer to figure 1. Finally, the boundary distance can be written as

$$\mathcal{L}_B(X, \theta) \coloneqq \langle c_\theta(X), \phi(X) \rangle\,. \tag{4}$$

First, we note the above formulation of ϕ was extracted from the source code. The statement in the paper is more general and leaves some freedom for its implementation. In essence, for a voxel i in $G(X)$, this definition of the boundary loss penalizes any prediction probability of a voxel being outside the boundary. Furthermore, the penalty is scaled the more inside the $G(X)$ lies. Interestingly, in contrast to most common loss functions, the boundary can and will often be negative in training. However, the function remains bounded below and will work out of the box with any first-order solver optimizing θ.

For an N-class segmentation problem, applying boundary loss can be achieved by transforming the problem to a set of N one-vs-all binary segmentation problems. Let $\mathcal{L}_\mathcal{B}^i$ be the loss associated with the i-th problem, then the boundary loss of the N-class problem can be written as $\mathcal{L}_\mathcal{B}(X, \theta) = \sum_{i=1}^N \mathcal{L}_\mathcal{B}^i(X, \theta)$

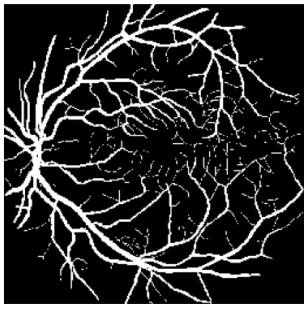
3 Related work, Impact and Discussion

Ines Vati

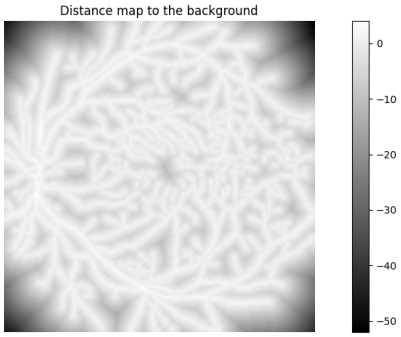
The cross-entropy loss is one of the most common loss functions for classification tasks. In particular, segmentation can be regarded as a voxel-wise classification problem. Under this setting, the cross-entropy loss is defined as

$$\mathcal{L}_{CE}(X, \theta) = -\langle g(X), \log(c_\theta(X)) \rangle - \langle \mathbf{1} - g(X), \log(\mathbf{1} - c_\theta(X)) \rangle\,, \tag{5}$$

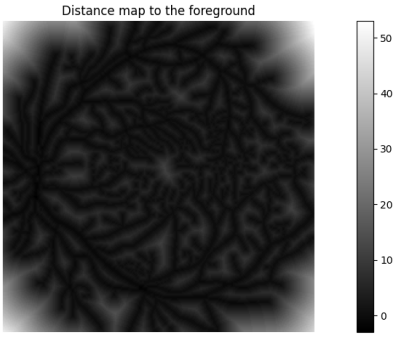
(a) RGB Input Image



(b) Segmentation map



(c) Distance map to G_0



(d) Distance map to G_1

Figure 1: Illustration of distance map on DRIVE Dataset
G_1 corresponds to the set of pixels constituting the vessels.

where the $\log$ is taken element-wise. It measures the affinity between predicted regions $c_\theta(X)$ and the ground-truth regions $g(X)$. However, it assumes identical distribution of all the samples and classes (including background/foreground). Therefore, using it alone is not well-suited for unbalanced segmentation class problems.

Several studies have already been conducted to address the challenges posed by the unbalanced region sizes in medical images. Sudre et al. generalized the Dice loss by weighting according to the squared inverse of class-label frequency, denoted as w_0 and w_1. This approach was shown to outperform the cross-entropy loss. The generalized Dice loss (GDL) is defined as

$$\mathcal{L}_{GDL}(X,\theta) = 1 - 2\frac{w_1 \langle g(X), c_\theta(X)\rangle + w_0 \langle \mathbf{1} - g(X), \mathbf{1} - c_\theta(X)\rangle}{w_1 \langle c_\theta(X) + g_\theta(x), \mathbf{1}\rangle + w_0 \langle 2 - c_\theta(X) - g(X), \mathbf{1}\rangle}, \tag{6}$$

with $w_0 = \|1 - g(X)\|_F^{-2}$ and $w_1 = \|g(X)\|_F^{-2}$.

However, Dice losses could still face challenges when addressing very small structures. In highly unbalanced scenarios, misclassification of pixels might result in loss reductions, leading to unstable optimization. Another reason that may explain the poor performances of Dice losses when dealing with extreme foreground/background class imbalance is that the Dice similarity coefficient (DSC)[1] corresponds to the harmonic mean between precision and recall. Indeed, using the same notation as

[1]Note that Dice loss is 1 - DSC.

above, we recover:

$$DSC = \frac{2|C_1 \cap G_1|}{|C_1| + |G_1|} = \frac{2}{\frac{|C_1|}{|C_1 \cap G_1|} + \frac{|G_1|}{|C_1 \cap G_1|}}$$
$$= \frac{2}{\frac{TP+FP}{TP} + \frac{TP+FN}{TP}}$$

Recognizing the precision and recall, it can be written as the following harmonic mean

$$DSC = \frac{2}{\frac{1}{\text{Precision}} + \frac{1}{\text{Recall}}}$$
$$= \frac{2}{2 + \frac{FP}{TP} + \frac{FN}{TP}}.$$

When the true positives remain the same, one notices that false positives and false negatives are equally important, making this loss inappropriate when both types of errors are not equally high (in particular when $FP >> FN$). Indeed, easily classified negatives comprise the majority of the loss and dominate the gradient. Lin et al. proposed to reshape the cross-entropy loss function to down-weight "easy" examples and thus focus on hard negatives. The focal loss is defined as

$$\mathcal{L}_{FL}(X, \theta) = -\langle g(X), (1 - c_\theta(X))^\gamma \log(c_\theta(X)) \rangle - \langle 1 - g(X), c_\theta(X)^\gamma \log(1 - c_\theta(X)) \rangle \quad (7)$$

with $\gamma \geq 0$ a tunable *focusing* parameter. It implies that when the predicted probability $c_\theta(X)_i$ of pixel i to belong to C_1 is small, its values have a higher weight in the loss function.

Furthermore, since 2019-2020, incorporating the distance transform maps of image segmentation labels into CNN pipelines, has received significant attention. For example, Ma et al. benchmarked five methods using the signed distance map. In addition to the method studied in this report, they have evaluated two other loss functions: the Hausdorff distance loss (Karimi and Salcudean) and the signed distance map regression loss (Xue et al.). They run 70 experiments to tune each method to achieve the best performance. According to their experimental results, all three losses have the potential to improve the performance of baseline CNNs. However, they observed that the performance gains are not consistent in different datasets and highly depend on implementation details like learning rates, loss functions and so on.

Nonetheless, several works still rely on the proposed boundary loss. For instance, Ribalta Lorenzo et al. studied multi-modal U-Net-based architecture with unsupervised pre-training and surface loss components for brain tumour segmentation which allowed them to seamlessly benefit from all magnetic resonance modalities during the delineation.

The result presented in (Kervadec et al.) shows that the proposed boundary loss does not work alone. It becomes interesting when it is associated with another regional loss ($\mathcal{L}_R$) with a certain weight. They specified that a suitable loss would be

$$\mathcal{L} = \mathcal{L}_R + \alpha \mathcal{L}_B \quad (8)$$

with $\alpha \in \mathbb{R}_+$ a parameter balancing the two losses. The chosen regional loss would be one of the above like the cross-entropy loss or the generalized Dice loss.

4 Experiments

For the experimental settings, we considered two datasets each representing a challenging setting. First, we considered the ACDC dataset [Bernard et al.], which is a multi-class segmentation dataset for heart delineation (left ventricle, right ventricle, myocardium, and background). In addition, we considered the DRIVE dataset [Staal et al.], which is a binary segmentation dataset for retinal vessel extraction.

4.1 ACDC Dataset

Mahmoud Hegazy

The ACDC dataset consists of 3D heart MRI scans of 150 people with 100 reserved for training and 50 for validation. Ground truth manual segmentation is available for all participants. The 3D scans are sliced into a sequence of 2D scans to reduce the computational cost. For our implementation, we relied on the source code provided by the authors[2]. In particular, we were interested in this dataset for two reasons. The first is that it tests the ability of the boundary loss to scale to multi-class problems. In addition, the performance on this dataset was not reported in the article Kervadec et al.. Sadly without access to a dedicated GPU, the computational cost of this experiment is high and we could conduct the experiments on the boundary loss without exploring other benchmarks.

For the experimental setting, we used E-Net Paszke et al. architecture. As an optimization algorithms, we used Adam Kingma and Ba with learning rate 5×10^{-3}, $\beta_1 = 0.9$, $\beta_2 = 0.99$, and a batch size of 8. During the training, we also measure the Dice loss (6) to evaluate the relation between minimizing the boundary loss and the performance w.r.t the Dice loss.

We report the results of the ACDC experiments in figures 2 and 3. Figure 2 displays the boundary loss scaled to $0, 1$ and the dice loss recorded during training. On the training set, minimizing the boundary loss strongly corresponded to smaller values of the Dice loss. However, on the validation set, the Dice loss fluctuated and such a strong relation was not observed. It is not immediately clear to us the reason behind such disparity. We believe that it is due to mistakes committed around the contour regions, as such mistakes are more penalized under the Dice loss. At the same time, such mistakes can be avoided on the training set due to overfitting. Figure 3 illustrates the output of the segmentation function over the validation set.

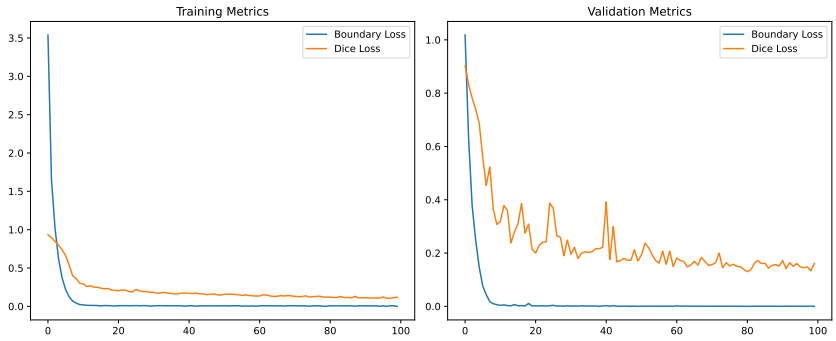


Figure 2: Boundary loss and Dice loss on the training and validation sets of ACDC dataset. As the boundary loss on ACDC is negative and on a different scale, the boundary loss was called to $(0, 1)$ for visual comparison

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

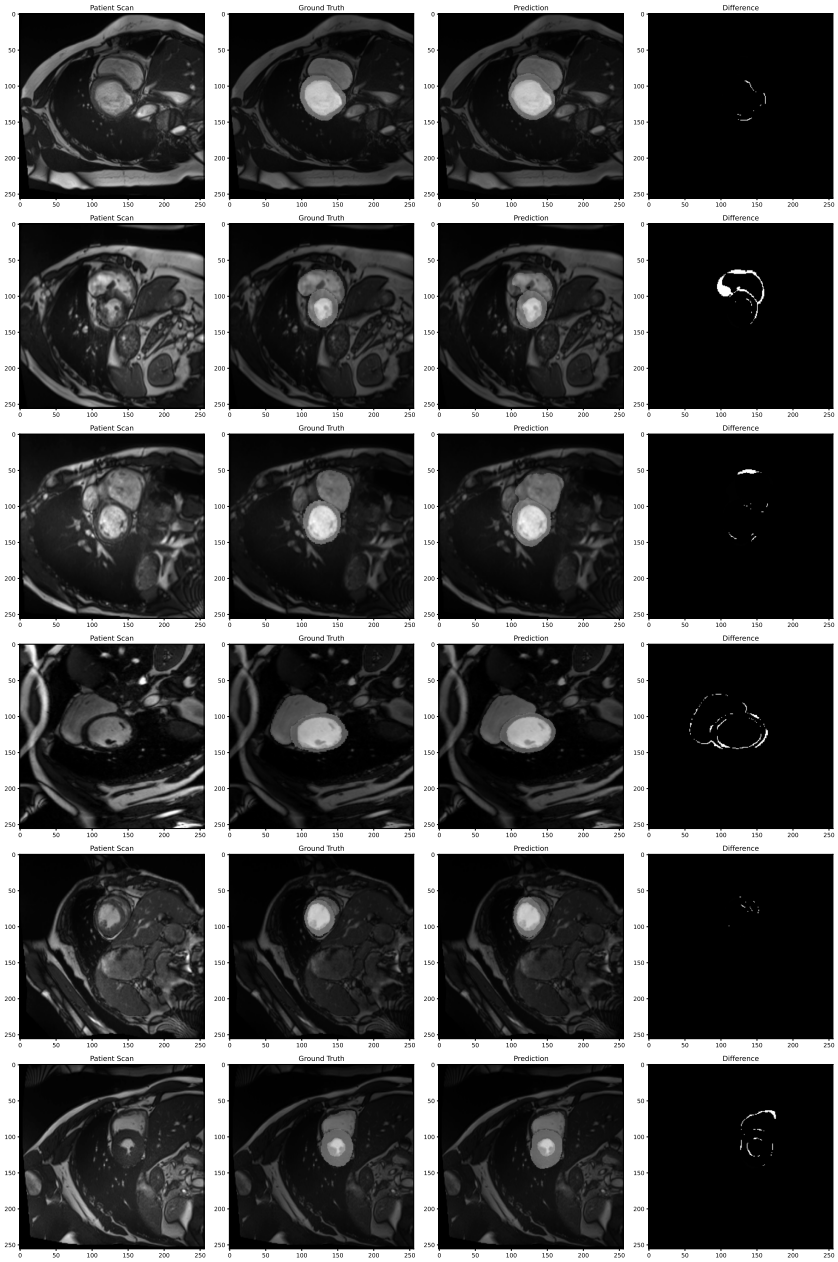


Figure 3: Output of the segmentation function on the validation set. 6 images out of the validation set were chosen at random. The first column corresponds to the input 2D scans, the second column to the ground truth segmentation, the third to the prediction of the CNN trained with boundary loss, and the last column illustrates the difference between the prediction and the ground truth.

4.2 DRIVE Dataset

Ines VATI

4.2.1 Experimental Design

We also trained a model with different losses to evaluate the applicability and performance of the proposed method on the DRIVE dataset. We chose this dataset as it challenges deep learning methods due to its small size. In addition, it was not tested by the authors. The code can be found here https://github.com/InesVATI/Medical-Image-Analysis-BoundaryLoss.

The images for the DRIVE database were obtained from a diabetic retinopathy screening program in The Netherlands. Twenty RGB 2D images of size (584×565) were available along with a manual segmentation of the vasculature. Twenty other retinal images were also given for testing but their segmentation map were not made available. We thought this dataset would be interesting to evaluate and compare the method given that the vessels are thin and far less profuse than the background as can be seen in the figure 5.

We train the model on Google Colab as it provides a T4 GPU and 15 GB of RAM. However, the memory was not enough for training the network on a batch size higher than 2. Thus we downsampled the image to a size of (256×256).

We also perform data augmentation by making horizontal random flips to make sure the network sees enough data. As in the article, we employed UNet as a deep learning architecture in our experiments. For the DRIVE dataset, we train a UNet network for 150 epochs with a batch size of 8[3]. We trained a model with a UNet architecture with 4 different losses :

- with the boundary loss associated with the generalized loss: $\mathcal{L}_1 = \mathcal{L}_{GDL} + \alpha\mathcal{L}_B$
- with the generalized loss alone : $\mathcal{L}_2 = \mathcal{L}_{GDL}$
- with the cross-entropy loss alone : $\mathcal{L}_3 = \mathcal{L}_{CE}$
- with the cross-entropy associated with the boundary loss : $\mathcal{L}_4 = \mathcal{L}_{CE} + \alpha\mathcal{L}_B$

For each experiment, the weight parameter is fixed to $\alpha = 0.05$. We chose the parameter value that gave suitable results in the experiments conducted by Kervadec et al.. They stated that even sub-optimal α can provide an improvement over the regional loss used alone.

4.2.2 Results

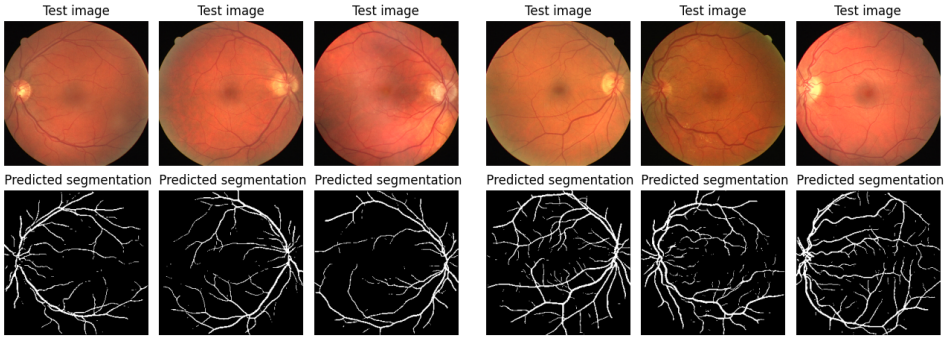
Quantitative evaluation. During training, the dice score between ground truth and predicted segmentation map was computed on the training and validation data (see fig. 10). The table 1 provides the dice score on the validation data at the end of the training. The values are of the same order of magnitude as the values obtained in the article (Kervadec et al.). However, we do not observe a significant improvement over a "standalone" regional loss. To improve those results on the DRIVE dataset, we could use a scheduling strategy for α. Indeed, Kervadec et al. noticed that increasing α or re-balancing the loss weights during training yields slightly better results than any constant α.

Qualitative evaluation. As test segmentation maps were not available, we could not compute the dice coefficient for those unseen data. However, we used test data for qualitative evaluation and computed the predicted segmentation map on random test images for each configuration. As illustrated by figure 4, the model achieves the recovery of the thin structure of the vessels. The vessels in the segmentation map computed with the model trained with $\mathcal{L}_1$ and $\mathcal{L}_4$ appear less dense than those obtained with the other models. Instead of using a fixed number of iterations to train the models, we should have used an early stopping and allowed each model to converge.

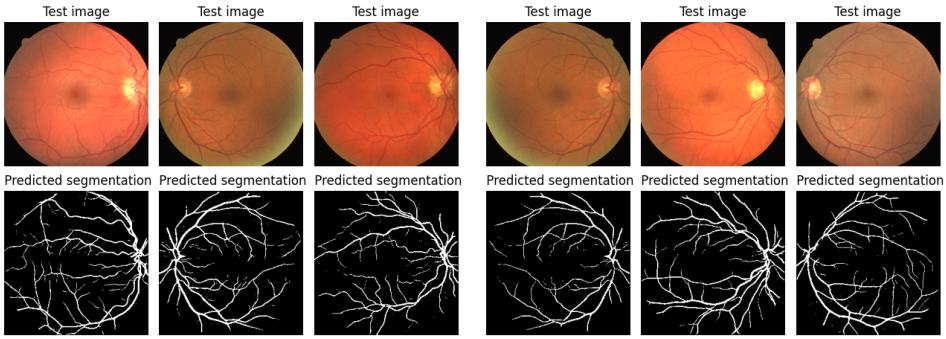
Loss	DSC
$\mathcal{L}_{GDL} + \alpha\mathcal{L}_B$	0.788
$\mathcal{L}_{GDL}$	0.793
$\mathcal{L}_{CE} + \alpha\mathcal{L}_B$	0.763
$\mathcal{L}_{CE}$	0.798

Table 1: Metric after training on DRIVE validation data

[3] In their article, they used a batch size of 8 for the WMH dataset and 4 for the ISLE dataset. They trained on 20 epochs



(a) $\mathcal{L}_1$ (b) $\mathcal{L}_2$



(c) $\mathcal{L}_3$ (d) $\mathcal{L}_4$

Figure 4: Evaluation of the trained model on test images

References

Olivier Bernard, Alain Lalande, Clement Zotti, Frederick Cervenansky, Xin Yang, Pheng-Ann Heng, Irem Cetin, Karim Lekadir, Oscar Camara, Miguel Angel Gonzalez Ballester, et al. Deep learning techniques for automatic mri cardiac multi-structures segmentation and diagnosis: is the problem solved? *IEEE transactions on medical imaging*, 37(11):2514–2525.

Davood Karimi and Septimiu E. Salcudean. Reducing the hausdorff distance in medical image segmentation with convolutional neural networks. URL http://arxiv.org/abs/1904.10030. version: 1.

Hoel Kervadec, Jihene Bouchtiba, Christian Desrosiers, Eric Granger, Jose Dolz, and Ismail Ben Ayed. Boundary loss for highly unbalanced segmentation. 67:101851. ISSN 13618415. doi: 10.1016/j.media.2020.101851. URL http://arxiv.org/abs/1812.07032.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr DollÃ¡r. Focal loss for dense object detection. URL http://arxiv.org/abs/1708.02002.

Jun Ma, Zhan Wei, Yiwen Zhang, Yixin Wang, Rongfei Lv, Cheng Zhu, Chen Gaoxiang, Jianan Liu, Chao Peng, Lei Wang, Yunpeng Wang, and Jianan Chen. How distance transform maps boost segmentation CNNs: An empirical study. In *Proceedings of the Third Conference on Medical Imaging with Deep Learning*, pages 479–492. PMLR. URL https://proceedings.mlr.press/v121/ma20b.html. ISSN: 2640-3498.

Adam Paszke, Abhishek Chaurasia, Sangpil Kim, and Eugenio Culurciello. Enet: A deep neural network architecture for real-time semantic segmentation. *arXiv preprint arXiv:1606.02147*.

Pablo Ribalta Lorenzo, Michal Marcinkiewicz, and Jakub Nalepa. Multi-modal u-nets with boundary loss and pre-training for brain tumor segmentation. In Alessandro Crimi and Spyridon Bakas, editors, *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries*, Lecture Notes in Computer Science, pages 135–147. Springer International Publishing. ISBN 978-3-030-46643-5. doi: 10.1007/978-3-030-46643-5_13.

Joes Staal, Michael D Abràmoff, Meindert Niemeijer, Max A Viergever, and Bram Van Ginneken. Ridge-based vessel segmentation in color images of the retina. *IEEE transactions on medical imaging*, 23(4):501–509.

Carole H. Sudre, Wenqi Li, Tom Vercauteren, SÃ©bastien Ourselin, and M. Jorge Cardoso. Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. volume 10553, pages 240–248. doi: 10.1007/978-3-319-67558-9_28. URL http://arxiv.org/abs/1707.03237.

Yuan Xue, Hui Tang, Zhi Qiao, Guanzhong Gong, Yong Yin, Zhen Qian, Chao Huang, Wei Fan, and Xiaolei Huang. Shape-aware organ segmentation by predicting signed distance maps. URL http://arxiv.org/abs/1912.03849.

Appendices

A DRIVE Database

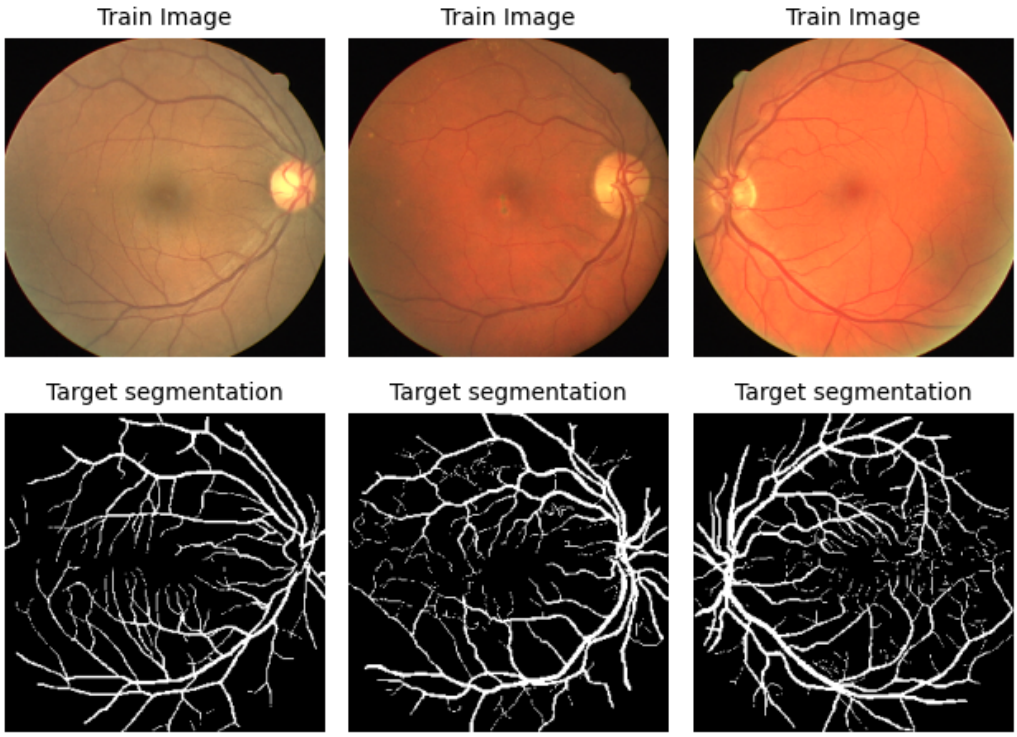


Figure 5: Examples of DRIVE database

B Learning curves for DRIVE dataset

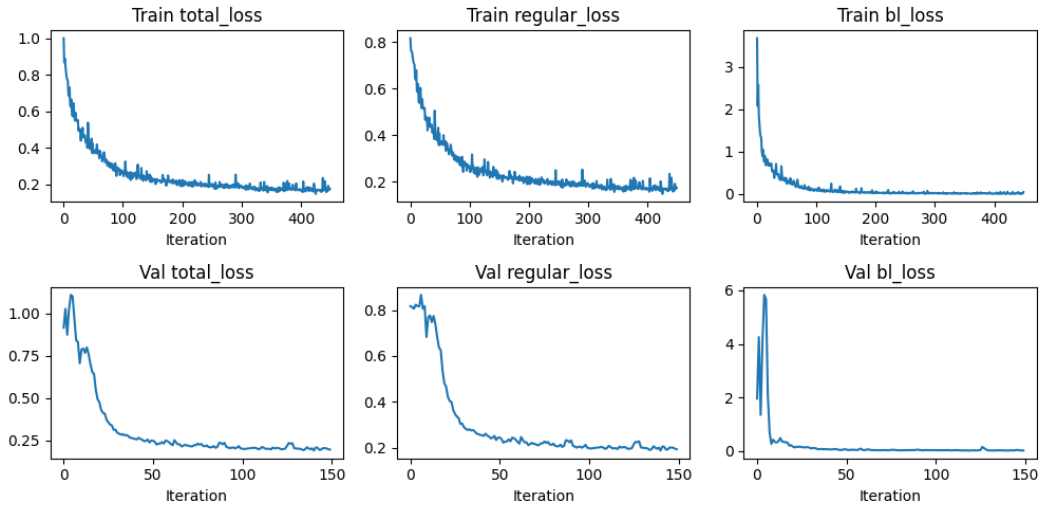


Figure 6: Loss evolution during training of DRIVE dataset with $\mathcal{L}_1$. The "regular_loss" is $\mathcal{L}_{GDL}$

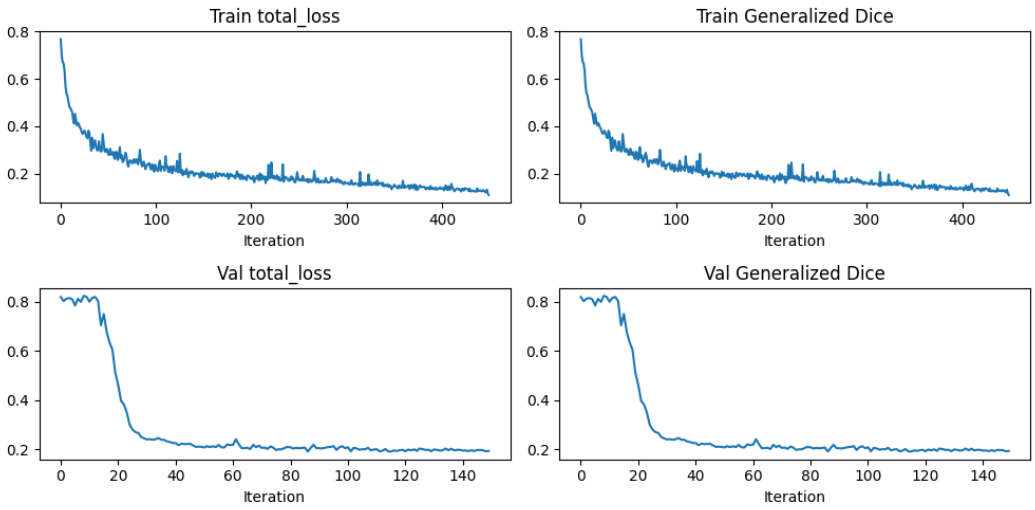


Figure 7: Loss evolution during training of DRIVE dataset with $\mathcal{L}_2$ (without boundary loss)

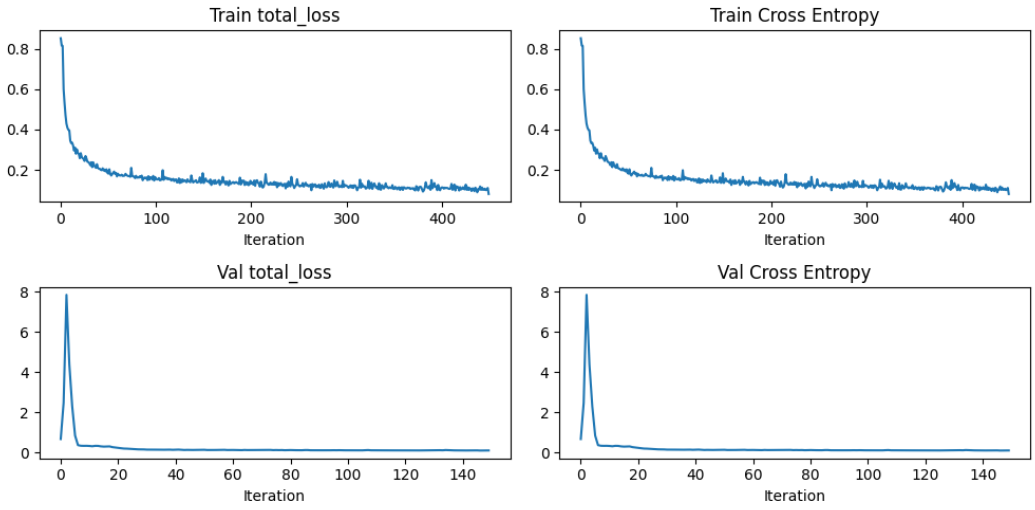


Figure 8: Loss evolution during training of DRIVE dataset with $\mathcal{L}_3$

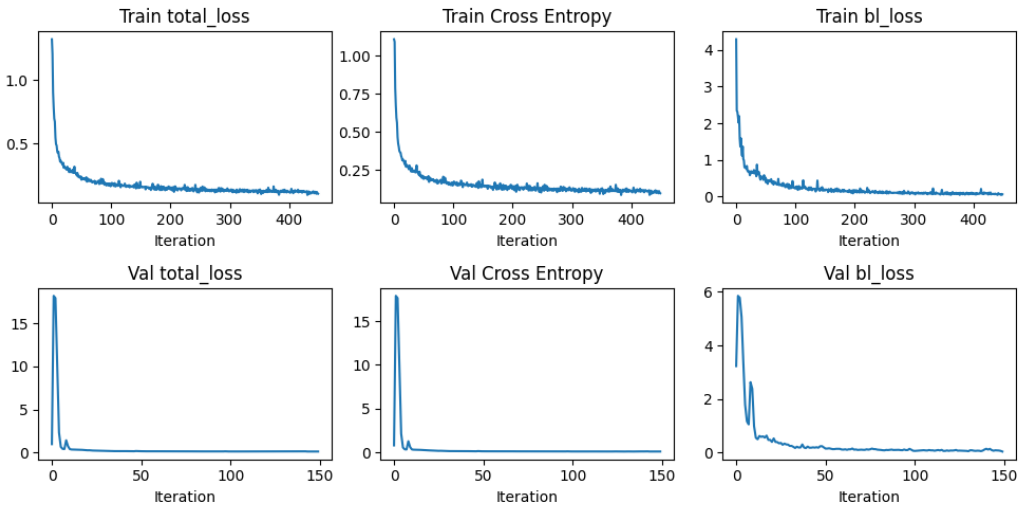
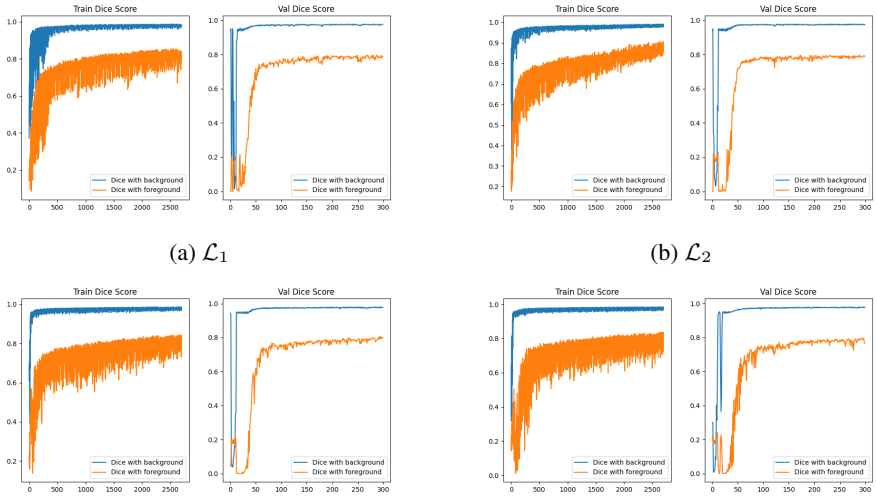


Figure 9: Loss evolution during training of DRIVE dataset with $\mathcal{L}_4$

C Dice score on DRIVE dataset



(a) $\mathcal{L}_1$ (b) $\mathcal{L}_2$

(c) $\mathcal{L}_3$ (d) $\mathcal{L}_4$

Figure 10: Evolution of the dice score during training on DRIVE dataset
The dice score for the foreground and background are plotted for the training and validation dataset. $\mathcal{L}_1$, $\mathcal{L}_2$ and $\mathcal{L}_3$ are the different losses evaluated during the experiments described above. $\mathcal{L}_3$ is the "baseline" loss. In our case, it is the cross-entropy loss function.