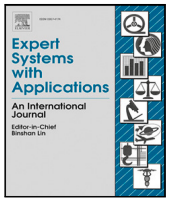
Expert Systems With Applications 238 (2024) 122024

Contents lists available at ScienceDirect

Expert Systems With Applications

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



Non-iterative scribble-supervised learning with pacing pseudo-masks for medical image segmentation

Zefan Yang [a,b,c], Di Lin [d], Dong Ni [a,b,c], Yi Wang [a,b,c,*]

[a] National-Regional Key Technology Engineering Laboratory for Medical Ultrasound, Guangdong Key Laboratory for Biomedical Measurements and Ultrasound Imaging, School of Biomedical Engineering, Health Science Center, Shenzhen University, Shenzhen, Guangdong, China
[b] Smart Medical Imaging, Learning and Engineering (SMILE) Lab, Shenzhen, Guangdong, China
[c] Medical UltraSound Image Computing (MUSIC) Lab, Shenzhen, Guangdong, China
[d] College of Intelligence and Computing, Tianjin University, Tianjin, China

ARTICLE INFO

Keywords:
Scribble-supervised learning
Medical image segmentation
Consistency training
Pseudo-mask
Siamese architecture
Memory bank

ABSTRACT

Scribble-supervised medical image segmentation tackles the limitation of sparse masks. Conventional approaches alternate between: labeling pseudo-masks and optimizing network parameters. However, such iterative two-stage paradigm is unwieldy and could be trapped in poor local optima since the networks undesirably regress to the erroneous pseudo-masks. To address these issues, we propose a non-iterative method where a stream of varying (pacing) pseudo-masks teach a network via consistency training, named PacingPseudo. Our contributions are summarized as follows. First, we design a non-iterative process. This process is achieved gracefully by a siamese architecture that comprises two weight-sharing networks. The siamese architecture naturally allows a stream of pseudo-masks to assimilate a stream of predicted-masks during training. Second, we make the consistency training effective with two necessary designs: (i) entropy regularization to obtain high-confidence pseudo-masks for effective teaching; and (ii) distorted augmentations to create discrepancy between the pseudo-mask and predicted-mask streams for consistency regularization. Third, we devise a new memory bank mechanism that provides an extra source of ensemble features to complement scarce labeled pixels. We evaluate the proposed PacingPseudo on public abdominal organ, cardiac structure, and myocardium datasets, named CHAOS T1&T2, ACDC, and LVSC. Evaluation metrics include the Dice similarity coefficient (DSC) and the 95th percentile of Hausdorff distance (HD95). Experimental results show that PacingPseudo achieves a 68.0% DSC and 14.1 mm HD95 on CHAOS T1, 73.7% DSC and 12.2 mm HD95 on CHAOS T2, 82.9% DSC and 4.3 mm HD95 on ACDC, and 61.4% DSC and 11.9 mm HD95 on LVSC. These results improve the baseline method by ≥3.1% in DSC and ≥14.2 mm in HD95. These results also outcompete previous methods. The fully-supervised method attains a 67.0% DSC and 16.7 mm HD95 on CHAOS T1, 71.2% DSC and 12.6 mm HD95 on CHAOS T2, 84.0% DSC and 3.9 mm HD95 on ACDC, and 72.9% DSC and 7.6 mm HD95 on LVSC. PacingPseudo's performance is comparable to the fully-supervised method on CHAOS T1&T2 and ACDC. Overall, the above results demonstrate the feasibility of PacingPseudo for the challenging scribble-supervised segmentation tasks. *The source code is publicly available at https://github.com/zefanyang/pacingpseudo*.

1. Introduction

The success of deep learning in semantic segmentation still relies on great amounts of fully annotated masks (Han et al., 2023; Isensee, Jaeger, Kohl, Petersen, & Maier-Hein, 2021; Litjens et al., 2017; Qi et al., 2022; Shen, Wu, & Suk, 2017; Uslu & Bharath, 2023). Annotating the segmentation masks inflicts high cost in the field of medical imaging because of the expertise and laborious workload needed in the process. Scribble-supervised medical image segmentation, which trains networks supervised by scribble annotations only, can be a feasible

The code (and data) in this article has been certified as Reproducible by Code Ocean: (https://codeocean.com/). More information on the Reproducibility Badge Initiative is available at https://www.elsevier.com/physical-sciences-and-engineering/computer-science/journals.

* Corresponding author at: National-Regional Key Technology Engineering Laboratory for Medical Ultrasound, Guangdong Key Laboratory for Biomedical Measurements and Ultrasound Imaging, School of Biomedical Engineering, Health Science Center, Shenzhen University, Shenzhen, Guangdong, China.
E-mail addresses: 2016222016@email.szu.edu.cn (Z. Yang), ande.lin1988@gmail.com (D. Lin), nidong@szu.edu.cn (D. Ni), onewang@szu.edu.cn (Y. Wang).

https://doi.org/10.1016/j.eswa.2023.122024
Received 24 January 2023; Received in revised form 2 October 2023; Accepted 2 October 2023
Available online 6 October 2023

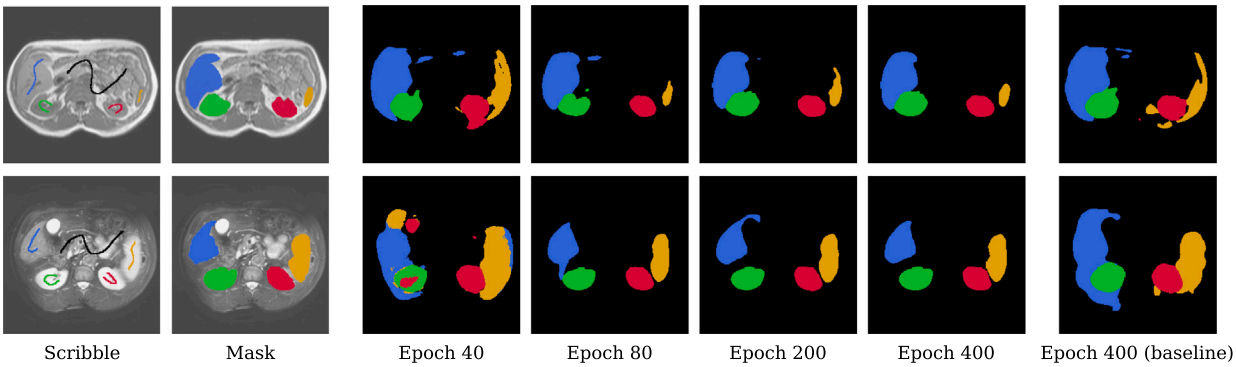


Fig. 1. Two examples showing the evolution of training-time inference predictions. Left (two columns): scribble annotations and ground-truth masks. Middle (four columns): the predictions of our PacingPseudo updating over epochs. Right (one column): the predictions in the final epoch (epoch 400) of the baseline (i.e., a network trained by a partial cross-entropy loss). It can be observed that the predictions of PacingPseudo gradually approximate the ground-truth masks while those of the baseline present inaccuracies. The images are from the training sets and training is supervised solely by scribbles.

way to reduce that burden. Created by dragging a cursor inside target regions, scribbles are flexible to edit structures (Tajbakhsh et al., 2020), but could only provide sparse labeled pixels while leaving vast regions unlabeled, posing a primary challenge in algorithm design.

Conventional scribble-supervised segmentation approaches (Can et al., 2018; Lin, Dai, Jia, He, & Sun, 2016) iterate between two stages: *labeling* pseudo-masks and *optimizing* network parameters; with the masks fixed, optimize the parameters, and vice versa. However, such paradigm has two major drawbacks. Firstly, it could be trapped in poor local optima, due to the reason that the networks probably regress to errors in the initial pseudo-masks and are unable to considerably reduce such mistakes in later iterations. Secondly, it is unwieldy especially when applied on large datasets. To bypass the iterative process, recent studies have attained a non-iterative one. These non-iterative approaches, which use either a regularizer (Tang, Djelouah, Perazzi, Boykov, & Schroers, 2018; Tang, Perazzi, et al., 2018) or knowledge from full masks (Valvano, Leo, & Tsaftaris, 2021) or mixed pseudo-masks (Luo, Hu, et al., 2022; Zhang & Zhuang, 2022), overlooked pure pseudo-masks, as opposed to those artificially mixed ones, for network training.

We argue that such stream can be useful and ask: *In a non-iterative method, how and to what extent, can pure pseudo-masks supervised by scribbles teach a network?* We attempt to answer the first part of the question by means of a siamese architecture (Bromley, Guyon, LeCun, Säckinger, & Shah, 1993) which has two weight-sharing neural networks applied on two inputs, based on the following analysis. **(i)** Set up a non-iterative paradigm. This paradigm, with the siamese architecture, can be achieved by translating the iterative two-stage process into: one network generating pseudo-masks supervised by scribbles (i.e., *labeling*) to assimilate the predicted-masks of the other network (i.e., *optimizing*) during training. **(ii)** Use pseudo-masks to teach a network. The pseudo-mask, supervised by scribbles, acts to regularize network parameters via consistency regularization (a regularizer) that maximizes similarity between it and the predicted-mask. An advantage is that these pseudo-masks are being diversified than of fixed quality, due to the continually updated network parameters which are different when mapping images at each training step. Each image's pseudo-masks are thereby varying between epochs ("pacing"). Fig. 1 shows the predictions of "PacingPseudo" gradually approximate the ground-truths as the network learns from more equivalently improving pseudo-masks through the training process.

To answer the second part of the question, which is about improving PacingPseudo's level of performance, we leverage insights on pseudo-labeling and augmentation from consistency training. Firstly, since labeled pixels are scarce in scribble-supervised segmentation, output pseudo-masks remain uncertain. Berthelot, Carlini, Cubuk, et al. (2019), Berthelot, Carlini, Goodfellow, et al. (2019), Sohn et al. (2020), Xie, Dai, Hovy, Luong, and Le (2020) use artificial post-processing (e.g., thresholding, sharpening, or argmax) to obtain high-confidence pseudo labels, whereas MeanTeacher (Yu, Wang, Li, Fu, & Heng, 2019) takes self-ensembling model's predictions as pseudo-masks. However, we empirically find these approaches are of limited effectiveness in our task, but the entropy regularization (Grandvalet & Bengio, 2004), that regularizes pseudo-masks end-to-end, performs satisfactorily. We then provide analysis about these findings. Secondly, augmentation is critical as it creates discrepancy between the pseudo-mask and predicted-mask branches to enable consistency regularization. Previous studies have promoted advanced augmentation techniques (Berthelot, Carlini, Cubuk, et al., 2019; Sohn et al., 2020; Xie, Dai, et al., 2020) or spatial augmentations (Bortsova, Dubost, Hogeweg, Katramados, & Bruijne, 2019; Patel & Dolz, 2022). In contrast, inspired by recent findings in representation learning (Chen, Kornblith, Norouzi, & Hinton, 2020; Grill et al., 2020) where augmentation serves a similar objective to create different views of an image (a positive pair) for assimilation, our study investigates a composition of distorted augmentations, which can be suitable and more convenient for consistency-training-based scribble-supervised segmentation.

We benchmark PacingPseudo on three public medical image datasets: CHAOS T1 and T2 (abdominal multi-organs) (Kavur et al., 2021), ACDC (cardiac structures) (Bernard et al., 2018), and LVSC (myocardium) (Suinesiaputra et al., 2014). Despite its simplicity, PacingPseudo improves the baseline by large margins and consistently outcompetes previous methods in the categories of consistency training, iterative training and non-iterative training. In some cases, PacingPseudo achieves comparable performance with its fully-supervised counterparts using ground-truth segmentation masks.

In conclusion, we list our contributions as follows:

- We design a non-iterative paradigm to bypass the iterative two-stage paradigm proposed by previous methods (Can et al., 2018; Lin et al., 2016). We opt for the siamese architecture that naturally does "labeling" and "optimizing" during training, allowing a stream of pseudo-masks with decreasing errors to reinforce network learning.
- We make pure pseudo-masks sufficient for scribble-supervised learning to avoid using redundant pseudo-mask manipulation operations introduced by previous methods (Lee & Jeong, 2020; Luo, Hu, et al., 2022; Zhang & Zhuang, 2022). We utilize entropy regularization to obtain high-confidence, accurate pseudo-masks. The pseudo-masks teach a network via consistency training. We use distorted augmentations to create discrepancy for consistency training. We further study an open question about the influence of the stop-gradient operation.
- We develop a memory bank mechanism, whereby an extra source of information, the ensemble of embedded labeled pixels across images, is introduced to complement scarce labeled supervision.

2. Related work

2.1. Iterative vs. Non-iterative weakly-supervised segmentation

In this section, we revisit weakly-supervised segmentation studies from an iterative or non-iterative perspective to position our study. Conventional iterative methods pre-process pseudo-masks for network training several times relying on different techniques. For instance, Can et al. (2018), Lin et al. (2016) use graph cuts (Boykov & Kolmogorov, 2004) or dense conditional random fields (DCRFs) (Krähenbühl & Koltun, 2011) to refine pseudo-masks (network inference predictions); Khoreva, Benenson, Hosang, Hein, and Schiele (2017) designs heuristic prior rules to de-noise pseudo-masks for better precision; Papandreou, Chen, Murphy, and Yuille (2015) incorporates background and foreground biases given weak labels via expectation–maximization steps; Roth, Yang, Xu, Wang, and Xu (2021) extends extreme points to pseudo-masks to supervise network training. Other than pre-processing, Dai, He, and Sun (2015) selects a small portion of candidate masks as supervision via a cost for network training in each epoch; Zhao et al. (2018) uses a two-step process where a detector generates proposals to be segmented.

In addition to ours, non-iterative methods do have been proposed for weakly-supervised segmentation (Dolz, Desrosiers, & Ayed, 2021; Kervadec et al., 2019; Lee & Jeong, 2020; Luo, Hu, et al., 2022; Patel & Dolz, 2022; Tang, Djelouah, et al., 2018; Tang, Perazzi, et al., 2018; Valvano et al., 2021; Zhang & Zhuang, 2022). Some studies add a regularizer to bypass pre-processing, based on shallow approaches (e.g., graph cuts) (Tang, Djelouah, et al., 2018; Tang, Perazzi, et al., 2018) or cardinality (Kervadec et al., 2019). Another category of studies transfer knowledge from full masks during training. While Dolz et al. (2021) constrains weakly-supervised predictions to be similar to fully-supervised ones, full masks train an auxiliary discriminator in Valvano et al. (2021). Beyond above approaches, Luo, Hu, et al. (2022), Zhang and Zhuang (2022) use cutout or mixed (i.e., linearly interpolated) pseudo-masks for network training. In contrast, our study argues that pure pseudo-masks (but not those artificial mixtures) can already be effective enough to teach a network and design our method inheriting spirits from recent consistency training.

2.2. Consistency training

Two aspects have been purposefully emphasized in consistency training mechanisms: pseudo-labeling to reduce uncertainty in pseudo-masks, and augmentation defining the neighborhood of an image to create discrepancy to enable consistency regularization. Regarding pseudo-labeling, (Berthelot, Carlini, Cubuk, et al., 2019; Berthelot, Carlini, Goodfellow, et al., 2019; Sohn et al., 2020; Xie, Dai, et al., 2020; Yu et al., 2019; Zou et al., 2020) obtain high-confidence pseudo labels using at least one of the following artificial post-processing operations: **(i)** thresholding: eliminates a distribution whose maximum probability is smaller than a threshold; **(ii)** sharpening: uses a temperature to sharpen a distribution; **(iii)** argmax: truncate a distribution to one-hot encoding. But since none of these proves effective in our task, we seek end-to-end entropy regularization (Grandvalet & Bengio, 2004). Owing to its simplicity, the entropy regularization has been investigated in medical imaging (Dolz et al., 2021; Luo, Liao, et al., 2022). However, while Dolz et al. (2021) reports collapse when incorporating it in point-supervised segmentation, Luo, Liao, et al. (2022) only tests its efficacy upon the baseline. In contrast, we show the entropy regularization not only generally improves over the baseline in scribble-supervised segmentation, but also can regularize low-entropy (i.e., high-confidence) pseudo-masks to reinforce network learning via consistency training.

In terms of augmentation in consistency training, while Berthelot, Carlini, Cubuk, et al. (2019), Sohn et al. (2020), Xie, Dai, et al. (2020) favor advanced augmentation techniques (e.g., RandAugment Cubuk, Zoph, Shlens, & Le, 2020, CTAugment (Berthelot, Carlini, Cubuk, et al., 2019)), some studies (Bortsova et al., 2019; Laradji et al., 2021; Li et al., 2020; Patel & Dolz, 2022) apply a spatial augmentation and its inverse version, and impose transformation equivariance. However, we note augmentation to obtain different views of the same image is not just required in consistency training, but also essential in representation learning (Chen & He, 2021; Chen et al., 2020; Grill et al., 2020; He, Fan, Wu, Xie, & Girshick, 2020). Both consistency training and representation learning share an objective that different views of the same image should be similar in output space. Inspired by this insight, our study explores distortion augmentation, recently popularized in the representation-learning community (Chen et al., 2020; Grill et al., 2020), for scribble-supervised segmentation in medical imaging.

3. Method

Our framework (Fig. 2) uses two weight-sharing neural networks, denoted as $f_\theta(\cdot)$. An input image x undergoes $\omega(\cdot)$ and then $\beta(\cdot)$ to produce two augmented views: a commonly-augmented view $\omega(x)$ and a further-augmented view $\beta\circ\omega(x)$. The predictions of $\omega(x)$ serve as pseudo-masks. Labeled pixels in pseudo-masks are penalized by a partial cross-entropy loss $\mathcal{L}_{pce}$ described in Section 3.1; unlabeled pixels are regularized by an entropy regularization loss $\mathcal{L}_{ent}$ described in Section 3.3. To use the pseudo-masks to guide network training, a consistency regularization loss $\mathcal{L}_{cr}$ described in Section 3.2 maximizes similarity between the predicted-masks of $\beta\circ\omega(x)$ and the pseudo-masks. To regularize network training, we incorporate a memory bank, an auxiliary loss $\mathcal{L}_{aux}$, and a memory loss $\mathcal{L}_{m}$ described in Section 3.4. The overall loss function is described in Section 3.5. The architecture of the network $f_\theta(\cdot)$ is described in Section 3.6. Training details are described in Section 3.7.

3.1. Partial cross-entropy

Training a network with the partial cross-entropy loss $\mathcal{L}_{pce}$ is the baseline in scribble-supervised segmentation. The loss $\mathcal{L}_{pce}$ only penalizes the predictions of labeled pixels and ignores those of unlabeled pixels, which is written as:

$$\mathcal{L}_{pce} = \frac{1}{N}\sum_{i=0}^{N-1} \texttt{CrossEntropy}(y_i, f_\theta(\omega(x))_i), \tag{1}$$

where $x \in \mathbb{R}^{H\times W}$ denotes an input image and N denotes the number of pixels in x. $y \in \mathbb{R}^{H\times W\times K}$ denotes a scribble-annotated label, where K is the number of classes (including the background). $y_i \in \mathbb{R}^K$ is either a one-hot vector for a labeled pixel or a zero vector for an unlabeled pixel. In such a setting, no gradient of the unlabeled pixels in $\mathcal{L}_{pce}$ is back-propagated. $f_\theta(\omega(x))_i \in \mathbb{R}^K$ denotes a pre-softmax logit, where $f_\theta(\cdot)$ is a network with trainable parameters θ and $\omega(\cdot)$ is a common augmentation operation. $\texttt{CrossEntropy}(p, q) = -p \cdot \log \texttt{softmax}(q)$ denotes a cross-entropy function, where p and q are K-dimensional vectors. Let $q' = \texttt{softmax}(q)$ that is $q'_i = \exp(q_i)/\sum_{j=0}^{K}\exp(q_j)$ for the ith channel.

3.2. Consistency regularization

However, training the network $f_\theta(\cdot)$ with solely the partial cross-entropy loss $\mathcal{L}_{pce}$ (i.e., the baseline) is rarely satisfactory, as shown in Section 5.2. Based on our motivation of designing a non-iterative method, we use the siamese architecture to perform pseudo-mask generation and network optimization during training. Specifically, to generate the pseudo-mask, we use the prediction of the commonly-augmented view $\tilde{y} \in \mathbb{R}^{H\times W\times K}$ that is $\tilde{y}_i \in \mathbb{R}^K = \texttt{softmax}(f_\theta(\omega(x))_i)$. To perform the network optimization, we use the consistency regularization loss $\mathcal{L}_{cr}$ to maximize cross-entropy similarity between the pseudo-mask $\tilde{y}$ and the predicted-mask $f_\theta(\beta\circ\omega(x))$:

$$\mathcal{L}_{cr} = \frac{1}{N}\sum_{i=0}^{N-1} \texttt{CrossEntropy}(\tilde{y}_i, f_\theta(\beta\circ\omega(x))_i), \tag{2}$$

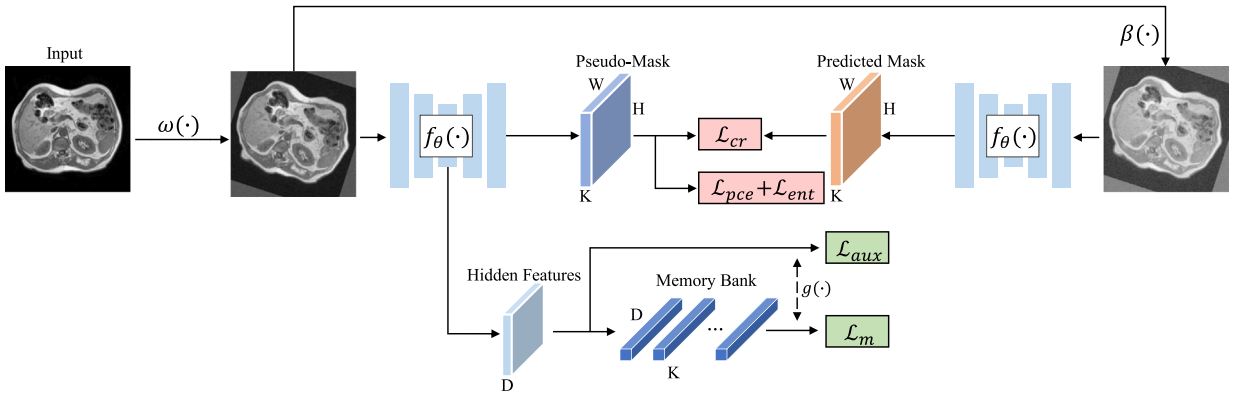


Fig. 2. Overview of our scribble-supervised segmentation framework. We use a siamese architecture that contains two weight-sharing neural networks $f_\theta(\cdot)$. The network processes the commonly-augmented (i.e., $\omega(\cdot)$) view of the input image and produces the pseudo-mask, which is used to assimilate the predicted-mask of the further-augmented (i.e., $\beta(\cdot)$) view. We incorporate a memory bank, which contains the ensemble features of each semantic class mapped by a shared prediction head $g(\cdot)$ in an auxiliary path. The notations $\mathcal{L}_{cr}$, $\mathcal{L}_{pce}$, $\mathcal{L}_{ent}$, $\mathcal{L}_{aux}$, and $\mathcal{L}_m$ denote the consistency regularization loss, partial cross-entropy loss, entropy regularization loss, auxiliary loss and memory loss, respectively, which are described in details in Section 3.

where $\beta(\cdot)$ is a further augmentation operation (described in Section 4.5) that applies additional augmentation on $\omega(x)$. In our method, we do not apply a stop-gradient operation on the pseudo-mask $\tilde{y}$ in the loss $\mathcal{L}_{cr}$, but many scribble-supervised or semi-supervised approaches did (Luo, Hu, et al., 2022; Xie, Dai, et al., 2020; Yu et al., 2019). Whether or not to apply stop-gradient leads to a different optimization trajectory, which we empirically find to considerably influence scribble-supervised segmentation. We therefore discuss our understanding of the stop-gradient operation in Section 5.4.

3.3. Entropy regularization

Our another motivation is to obtain effective low-entropy (i.e., high-confidence) pseudo-masks. To achieve this aim, our method incorporates the entropy regularization loss $\mathcal{L}_{ent}$. $\mathcal{L}_{ent}$ is based on the concept of Shannon entropy. Its value reflects the degree of uncertainty in a probability distribution. Specifically, $\mathcal{L}_{ent}$ has a low value when a distribution is nearly deterministic, but it has a high value when a distribution is nearly uniform. The loss $\mathcal{L}_{ent}$ is written as:

$$\mathcal{L}_{ent} = -\frac{1}{N}\sum_{i=0}^{N-1} \tilde{y}_i \cdot \log \tilde{y}_i, \tag{3}$$

where $\tilde{y}_i \in \mathbb{R}^K$ is the softmax distribution of a pseudo-mask pixel. We train $f_\theta(\cdot)$ with the loss $\mathcal{L}_{ent}$ end-to-end, compared with Xie, Dai, et al. (2020), Yu et al. (2019) that require the artificial post-processing operations (thresholding and sharpening) to obtain low-entropy pseudo-masks to avoid degeneration. We empirically show that the former works satisfactorily, whereas the latter are not effective. Related to this finding, we discuss the entropy regularization loss from a decision boundary's perspective to explain its power in Section 5.3.

3.4. Memory bank

To complement scarce labeled supervision from scribbles, we introduce extra information that represents the ensemble of embedded labeled pixels across images, analogous to class prototypes in Snell, Swersky, and Zemel (2017), to regularize network learning. Specifically, we maintains a memory bank containing the ensemble features of each semantic class, implemented as momentum moving averages of labeled-pixel features. We build the memory bank $M \in \mathbb{R}^{K\times D}$ based on encoder features $f_e(\omega(x)) \in \mathbb{R}^{\frac{H}{8}\times\frac{W}{8}\times C}$, where $f_e(\cdot)$ denotes the encoder in the network $f_\theta(\cdot)$ and $C = 512$ denotes the number of channels. $f_e(\omega(x))$ are upsampled and projected to hidden features $z \in \mathbb{R}^{H\times W\times D}$, where $D = 64$. Considering a set of indices pointing to the labeled pixels of class k in the label y as $\mathcal{I}_k = \{i | y_{ik} = 1, \forall i \in \{0, \dots, N-1\}\}$, the mean features of the labeled pixels $m_k \in \mathbb{R}^D$ are computed by:

$$m_k = \sum_{i\in\mathcal{I}_k} s_{ik} z_i, \tag{4}$$

$$s_{ik} = \frac{1-\texttt{sim}(M_k, z_i)}{\sum_{j\in\mathcal{I}_k}(1-\texttt{sim}(M_k, z_j))}, \tag{5}$$

where s_{ik} is a weight scalar indicating the relative importance of a pixel representation $z_i \in \mathbb{R}^D$. Let $\texttt{sim}(p, q) = p/\|p\| \cdot q/\|q\|$ denote the dot product between ℓ_2 normalized p and q, namely cosine similarity. s_{ik} is reversely proportional to the cosine similarity between M_k and z_i, according to (5). This prioritizes dissimilar representations for update:

$$M_k \leftarrow \alpha M_k + (1-\alpha) m_k, \tag{6}$$

where $M_k \in \mathbb{R}^D$ is initialized to a zero vector (detached) and updated by the momentum moving average of m_k. $\alpha = 0.9$ is a momentum coefficient. We incorporate a weight-sharing prediction head $g(\cdot)$ to map the hidden features z and the memory bank M. The auxiliary loss $\mathcal{L}_{aux}$ penalizes structured predictions, whilst the memory loss $\mathcal{L}_m$ punishes the semantic predictions derived from the memory bank features. The losses are computed by:

$$\mathcal{L}_{aux} = \frac{1}{N}\sum_{i=0}^{N-1} \texttt{CrossEntropy}(y_i, g(z)_i), \tag{7}$$

$$\mathcal{L}_m = \frac{1}{K}\sum_{k=0}^{K-1} \texttt{CrossEntropy}(\hat{y}_k, g(M)_k), \tag{8}$$

where $\hat{y} \in \mathbb{R}^{K\times K}$ (a unit matrix) represents the label of M. The loss $\mathcal{L}_m$ influences the weights of $g(\cdot)$ in back-propagation, which then affects pre-softmax logits $g(z)$ from the encoder $f_e(\cdot)$ to perform regularization.

3.5. Overall loss

We denote an overall loss function as:

$$\mathcal{L} = \mathcal{L}_{pce} + r(t)\mathcal{L}_{cr} + r(t)\mathcal{L}_{ent} + \lambda_1\mathcal{L}_{aux} + \lambda_2\mathcal{L}_m, \tag{9}$$

where $r(t)$ is an essential warm-up function applied on $\mathcal{L}_{cr}$ and $\mathcal{L}_{ent}$ to filter out noise in the early stage, and λ_1 and λ_2 are coefficients balancing loss terms. We use an exponential form of $r(t)$ that is $r(t) = \exp(-\eta(1-t/T))$, where t is an epoch index, and T and η are warm-up hyper-parameters. Specifically, during the first T epochs, $r(t)$ ramps up from a small positive value to 1, and then remains unchanged until training is completed. And the hyper-parameter η controls the speed of the warm-up process. A larger value of η makes the warm-up process slower. We set $T = 80$ and $\eta = 8$ (unless otherwise specified) which work well in our experiments.

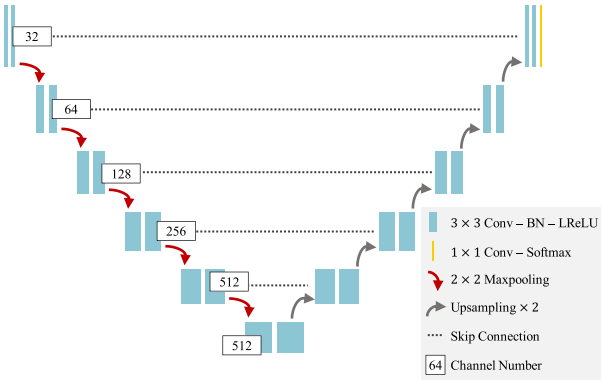


Fig. 3. Network architecture. We use the encoder–decoder architecture with skip-connections as our backbone network. Each stage contains two convolutional layers in sequence. A convolutional layer comprises a 3 × 3 convolution, batch normalization, and LeakyReLU.

Algorithm 1 Pseudocode of PacingPseudo

Require: Image x, scribble-annotated label y
Require: Backbone network $f_\theta(\cdot)$
Require: Common augmentation operation $\omega(\cdot)$
Require: Further augmentation operation $\beta(\cdot)$
for t in [1, num_epochs] **do**
  for each minibatch **do**
    \\ Compute pesudo-masks using $\omega(\cdot)$
    $\tilde{y} = f_\theta(\omega(x))$
    \\ Compute predicted-masks using $\beta \circ \omega(\cdot)$
    $y' = f_\theta(\beta \circ \omega(x))$
    \\ Compute memory bank features
    $M_k \leftarrow \alpha M_k + (1-\alpha) m_k$
    Compute $\mathcal{L}_{pce}$, $\mathcal{L}_{cr}$, $\mathcal{L}_{ent}$, $\mathcal{L}_{aux}$, $\mathcal{L}_m$
    Update θ using Adam
  end for
end for
return θ

3.6. Backbone

We use the UNet's encoder–decoder architecture (Ronneberger, Fischer, & Brox, 2015) as our backbone network with a few modifications, as shown in Fig. 3. We set the encoder depth (i.e., the number of encoder stages) to a value that an encoder output has a spatial size smaller than 8 × 8, following Isensee et al. (2021). Specifically, we set the encoder depth to 6, leading to an output stride (i.e., the downsampling scale) of 32 suitable for the spatial resolutions of our datasets. We set the initial number of channels to 32 and the maximum number of channels to 512. This means the largest channel number is set to 512 regardless of the encoder depth. A convolutional layer is composed of a 3 × 3 convolution, batch normalization, and LeakyReLU (with a negative slope of 0.01). A prediction head contains a 1 × 1 convolution and softmax operation. We use maxpooling for downsampling and bilinear interpolation for upsampling.

3.7. Training

We use Adam as our optimizer. The Adam weight decay is 0.0003. The Adam learning rate is 0.0001, decayed by a polynomial policy. Specifically, the learning rate is multiplied by a scale $(1 - \frac{t}{\text{num\_epochs}})^{0.9}$, where t is an epoch index. We use a mini-batch of 12. On CHAOS and ACDC, we train for 400 epochs and set $T = 80$ for the ramp-up function $r(t)$. On LVSC, because of its larger scale, we train for 40 epochs and set $T = 8$. We obtain loss function's hyperparameters λ_1 and λ_2 using grid search in {0.01, 0.1, 1}. Based on grid search results, we set $\lambda_1 = 0.01$ and $\lambda_2 = 1$. We train networks on four 11 GB GPUs. Pseudocode is presented in Algorithm 1. *The source code is available at* https://github.com/zefanyang/pacingpseudo.

4. Experimental settings

4.1. Datasets

CHAOS T1 and T2. The CHAOS dataset (Kavur et al., 2021) provides 20 patients for multi-organ segmentation. The patients have two MRI sequences (T1-DUAL (in-phase and out-phase) and T2-SPIR) acquired by a 1.5T Phillips MRI. T1 in-phase and out-phase are well-aligned and use a shared ground-truth, while T2-SPIR uses an independent one. The ground-truths include four targets: liver (LIV), left kidney (LK), right kidney (RK) and spleen (SPL). This dataset has 1917 slices (~32 slices per sequence). We manually delineate scribbles (including the background) based on the ground-truth using the one-pixel brush in ITK-SNAP (see Fig. 4).[1] In pre-processing, we first resample images to 1.62 × 1.62 mm^2 (the median spacing), and then center crop or pad them to 256 × 256 pixels (the median size). We train separate networks for T1-DUAL (CHAOS T1) and T2-SPIR (CHAOS T2).

ACDC. The ACDC dataset (Bernard et al., 2018) provides 100 patients for heart structure segmentation. The patients have cine-MRI sequences at the end-diastolic and end-systolic instant. Segmentation targets include right ventricle (RV), myocardium (MYO), and left ventricle (LV). This dataset has 1902 slices (~10 slices per sequence). We use scribble annotations provided by Valvano et al. (2021). We resample images to 1.51 × 1.51 mm^2 (the median spacing) and center crop or pad them to 224 × 224 pixels (the median size).

LVSC. The LVSC dataset (Suinesiaputra et al., 2014) provides 100 patients for myocardium (MYO) segmentation. This dataset contains "2D+time" sequences. We incorporate slices in short-axis view. This dataset is relatively large, consisting of 29,086 images (~291 slices per patient). We use artificial scribble annotations generated by skeletonizing the target and background, and then expanding the unlabeled regions. We resample images to 1.48 × 1.48 mm^2 (the median spacing) and center crop or pad them to 256 × 256 pixels (the median size).

Overall, the scribbles occupy ~10% of the foreground and ~0.5% of the background on the above datasets. In our datasets, annotating scribbles takes 10–25 s per slice, whereas annotating ground-truth masks take 50–70 s per slice.

4.2. Comparison methods

We compare PacingPseudo with three types of methods in medical imaging and computer vision. First, to show the effectiveness of our consistency training mechanism, we compare ours with consistency training methods (UDA (Xie, Dai, et al., 2020) and Mean-Teacher (Yu et al., 2019)). Second, to support our motivation of circumventing the iterative process, we compare ours with iterative training methods (NoisyStudent (Xie, Luong, Hovy, & Le, 2020) and IterativeTraining (Can et al., 2018)). Third, to illustrate the advantage of our non-iterative process, we compare ours with non-iterative training methods (EntMin (Luo, Liao, et al., 2022), RegularizedLoss (Tang, Perazzi, et al., 2018), MixedPseudo (Luo, Hu, et al., 2022), and Scribble2Label (Lee & Jeong, 2020)). We note, among them, IterativeTraining, EntMin, RegularizedLoss, and MixedPseudo are from the field of scribble-supervised segmentation. Implementation details that highlight their primary difference are as follows:

[1] http://www.itksnap.org/

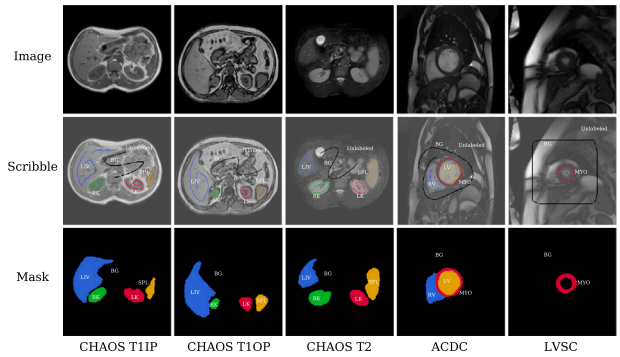


Fig. 4. Examples of images, scribbles and masks. CHAOS and ACDC use manually delineated scribbles, while LVSC uses artificial scribbles. T1IP and T1OP denote T1 in-phase and out-phase respectively. The color-to-structure correspondences are fixed in this paper.

- Consistency training methods
  - **UDA**: uses a siamese architecture trained with $\mathcal{L}_{pce}$ (1) and consistency regularization. It eliminates distributions whose maximum probability is lower than a 0.95 threshold, and then sharpens them via a 0.5 temperature.
  - **MeanTeacher**: uses a teacher-student architecture trained with $\mathcal{L}_{pce}$ and consistency regularization. The momentum of the teacher is set to 0.999. The same thresholding and sharpening as UDA is implemented.
- Iterative training methods
  - **NoisyStudent**: adds extra noises via $\beta(\cdot)$ to images, and via dropout (with a 0.5 probability) after max-pooling at encoder stage 5 and 6 when training a network in iterations. Pre-processing follows the setting below.
  - **IterativeTraining**: pre-processes pseudo-masks using DCRFs in iterations. The hyper-parameters of DCRFs are $w_1 = 5$, $w_2 = 10$, $\sigma_\alpha = 2$, $\sigma_\beta = 0.1$, $\sigma_\gamma = 5$.
- Non-iterative training methods
  - **EntMin**: trains a network with both $\mathcal{L}_{pce}$ and $\mathcal{L}_{ent}$ (3) but dose not use low-entropy pseudo-masks to reinforce network learning.
  - **RegularizedLoss**: proposes a DCRF regularizer. It first pre-trains a network with $\mathcal{L}_{pce}$, and then fine-tunes the network with both $\mathcal{L}_{pce}$ and the DCRF regularizer.
  - **MixedPseudo**: interpolates and applies argmax on predictions from two decoders to produce hard pseudo-masks, and uses them as supervision via the Dice loss.
  - **Scribble2Label**: uses both scribble annotations and a moving-averaged pseudo-mask to supervise network predictions during training.

4.3. Ablation analyses

In Section 5.2, to quantify the gap between our method and its fully-supervised counterpart, we compare the two and analyze factors that contribute to their performance difference. In Section 5.3, we ablate the network components and provide our understanding of their effectiveness. In Section 5.4, we study the gradient flows of pseudo-masks in consistency regularization and discuss why no stop-gradient is better. In Section 5.5, we show how the further augmentation operation's strength influence performance and discuss its role in our method. In Section 5.6, to clarify the impact of scribble coverage over targets, we evaluate the robustness of our method to different scribble lengths. Lastly, in Section 5.7, we visualize some results to illustrate the limitations of our method.

4.4. Evaluation protocols

We evaluate methods using *five-fold cross-validation*. Images are split at patient level. We quantify the quality of predictions using two metrics: the Dice similarity coefficient (DSC) and the 95th percentile of Hausdorff distance (HD95) (Wang et al., 2019). While DSC (%) measures relative overlap between the ground-truth and the prediction (higher is better), HD95 (mm) quantifies the longest distance over the shortest distances between the surface pixels of the ground-truth and the prediction (lower is better).

4.5. Settings of augmentation operations

The common augmentation operation $\omega(\cdot)$ comprises geometry and noise augmentations. Specifically, an input image is first normalized to zero mean and unit variance. Then, it undergoes resizing, elastic deformation, rotation, horizontal and vertical flip, and Gaussian noises. Last, the image is randomly cropped (or padded) to the original input size. The magnitudes and probabilities of the above augmentations follow the same setting in Isensee et al. (2021).

In terms of the further augmentation operation $\beta(\cdot)$, we focus on a composition of color-distorted augmentations described in SimCLR (Chen et al., 2020) and curate them to suit scribble-supervised segmentation in medical imaging. The set of operations which $\beta(\cdot)$ is sampled from is as follows:

- **Brightness**: increases pixel intensities by a value drawn from a uniform distribution $U(-0.8\delta, +0.8\delta)$.
- **Contrast**: multiplies pixel intensities by a scale drawn from $U(1-0.8\delta, 1+0.8\delta)$ and then clips them to the original min–max range.
- **GammaAugment**: applies a min–max normalization and then raises pixel intensities to the power of γ drawn from $U(1-0.8\delta, 1+0.8\delta)$.

The notation $\delta \in (0, 1]$ denotes the strength of augmentations, which is set to 1 by default. Brightness, contrast and gamma augmentation are conducted sequentially and each operation is applied with a 0.8 probability.

5. Results and discussion

5.1. Comparison with previous methods

5.1.1. Comparative analysis of quantitative results

As seen in Table 1, PacingPseudo achieves overall best segmentation results on the three experimental datasets and are significantly better than a majority of results of the comparison methods in both DSC and HD95 (p-value <0.05 in two-sample t-tests). Moreover, due to the differences in image modalities and target shapes, while the performance difference is relatively small on ACDC, PacingPseudo outperforms the comparison methods by large margins on CHAOS T1&T2 and LVSC. Segmentation results are illustrated in Fig. 5, wherein, obviously, ours are most similar to the ground-truths compared with those of the comparison methods.

Next, we provide analysis related to the experimental results of the comparison methods. The main difference between UDA and PacingPseudo is that while UDA uses the artificial post-processing operations to manipulate pseudo-masks, PacingPseudo uses the end-to-end loss $\mathcal{L}_{ent}$ (3). The better performance of PacingPseudo suggests that the artificial operations are not effective for scribble-supervised segmentation. Our explanation is that artificially processed pseudo-masks (which, for example, cover only high-confidence regions) are not what a network predicts, which leads to too serious inconsistency to be handled via consistency regularization and causes unsatisfactory performance. Besides, MeanTeacher obtains the worst performance compared with UDA and PacingPseudo (both using the siamese architecture). We conjecture two factors attributed to its failure: (i) the moving-averaged model's

Table 1
Comparison with previous methods. Results are based on five-fold cross-validation and shown in MEAN$_{SD}$. The best results are underlined, and the **boldface** indicates the results are statistically different with ours ($p < 0.05$).

	Method	CHAOS T1					CHAOS T2					ACDC				LVSC
		LIV	RK	LK	SPL	*Average*	LIV	RK	LK	SPL	*Average*	RV	MYO	LV	*Average*	MYO
DSC	UDA	**49.0**$_{36}$	**59.1**$_{30}$	**51.4**$_{34}$	**20.8**$_{26}$	**45.1**$_{15}$	**43.6**$_{37}$	**69.2**$_{28}$	**59.9**$_{33}$	**25.3**$_{34}$	**49.5**$_{17}$	**76.0**$_{31}$	**81.5**$_{19}$	**87.3**$_{21}$	**81.6**$_{05}$	**46.7**$_{21}$
	MeanTeacher	**42.8**$_{36}$	**49.4**$_{35}$	**51.9**$_{36}$	**27.3**$_{32}$	**42.9**$_{10}$	**35.3**$_{33}$	**61.8**$_{39}$	**54.0**$_{38}$	**30.3**$_{37}$	**45.3**$_{13}$	**68.6**$_{33}$	**66.1**$_{24}$	**85.7**$_{22}$	**73.5**$_{09}$	**38.8**$_{20}$
	NoisyStudent	**52.0**$_{37}$	**54.3**$_{34}$	**55.8**$_{34}$	**27.9**$_{31}$	**47.5**$_{11}$	**18.3**$_{23}$	**49.5**$_{38}$	**58.7**$_{36}$	**30.4**$_{35}$	**39.2**$_{16}$	**75.5**$_{31}$	**81.4**$_{18}$	87.5$_{21}$	**81.5**$_{05}$	**41.8**$_{18}$
	IterativeTraining	**54.9**$_{37}$	**56.4**$_{33}$	**55.4**$_{35}$	**31.7**$_{32}$	**49.6**$_{10}$	**19.2**$_{24}$	**51.3**$_{38}$	**60.9**$_{35}$	**26.8**$_{32}$	**39.6**$_{17}$	76.5$_{31}$	82.3$_{18}$	87.8$_{20}$	82.2$_{05}$	**42.5**$_{17}$
	EntMin	**62.2**$_{36}$	**56.1**$_{35}$	**56.3**$_{37}$	**40.3**$_{35}$	**53.7**$_{08}$	**52.6**$_{39}$	**65.3**$_{36}$	**62.6**$_{38}$	**39.9**$_{39}$	**55.1**$_{10}$	**75.9**$_{30}$	82.0$_{18}$	87.8$_{20}$	**81.9**$_{05}$	**51.2**$_{23}$
	RegularizedLoss	**56.5**$_{38}$	**52.9**$_{34}$	**52.1**$_{37}$	**36.4**$_{36}$	**49.5**$_{08}$	**39.3**$_{39}$	**24.8**$_{37}$	**47.9**$_{40}$	**24.3**$_{36}$	**34.1**$_{10}$	**75.0**$_{31}$	82.2$_{18}$	88.0$_{19}$	**81.8**$_{05}$	**59.4**$_{28}$
	MixedPseudo	**51.9**$_{36}$	**49.8**$_{31}$	**46.9**$_{33}$	**22.5**$_{28}$	**42.8**$_{12}$	**48.4**$_{38}$	**64.5**$_{29}$	**58.5**$_{33}$	**21.7**$_{30}$	**48.3**$_{16}$	75.8$_{31}$	81.9$_{18}$	88.3$_{19}$	**82.0**$_{05}$	**48.0**$_{20}$
	Scribble2Label	**49.7**$_{35}$	**51.6**$_{31}$	**47.3**$_{34}$	**26.8**$_{27}$	**43.9**$_{12}$	**37.5**$_{36}$	**48.6**$_{37}$	**53.4**$_{35}$	**33.6**$_{37}$	**43.3**$_{09}$	**73.6**$_{31}$	**81.3**$_{18}$	**87.3**$_{21}$	**80.7**$_{07}$	**42.3**$_{22}$
	PacingPseudo	80.5$_{27}$	66.5$_{33}$	63.6$_{36}$	61.3$_{36}$	68.0$_{07}$	79.1$_{30}$	78.7$_{30}$	77.1$_{32}$	59.8$_{42}$	73.7$_{07}$	77.7$_{29}$	82.5$_{17}$	88.4$_{19}$	82.9$_{04}$	61.4$_{22}$
HD95	UDA	**90.1**$_{97}$	**18.0**$_{22}$	**19.0**$_{18}$	**74.2**$_{50}$	**50.3**$_{32}$	**142.2**$_{99}$	**11.3**$_{11}$	**13.0**$_{15}$	**39.5**$_{26}$	**51.6**$_{54}$	**8.5**$_{17}$	**10.9**$_{33}$	**8.3**$_{30}$	**9.3**$_{01}$	**24.5**$_{20}$
	MeanTeacher	**133.2**$_{83}$	**23.0**$_{29}$	**17.7**$_{23}$	**50.2**$_{41}$	**56.0**$_{46}$	**185.8**$_{66}$	**31.8**$_{62}$	**17.2**$_{29}$	**39.2**$_{40}$	**68.5**$_{68}$	**49.5**$_{67}$	**106.3**$_{69}$	**17.5**$_{41}$	**57.8**$_{37}$	**44.8**$_{35}$
	NoisyStudent	**66.7**$_{79}$	**15.0**$_{16}$	**13.5**$_{15}$	**45.2**$_{33}$	**35.1**$_{22}$	**243.5**$_{81}$	**63.2**$_{108}$	**11.5**$_{12}$	**55.9**$_{72}$	**93.5**$_{89}$	**10.1**$_{23}$	**14.5**$_{36}$	**8.1**$_{24}$	**10.9**$_{03}$	**29.6**$_{21}$
	IterativeTraining	**40.8**$_{56}$	**14.5**$_{12}$	**21.2**$_{30}$	**37.8**$_{29}$	**28.6**$_{11}$	**238.1**$_{86}$	**53.2**$_{94}$	**12.0**$_{16}$	**77.1**$_{96}$	**95.1**$_{86}$	**8.0**$_{19}$	**9.7**$_{27}$	**6.7**$_{21}$	**8.1**$_{01}$	**26.1**$_{18}$
	EntMin	**34.1**$_{47}$	**16.5**$_{22}$	**14.7**$_{22}$	**29.5**$_{26}$	**23.7**$_{08}$	**51.5**$_{62}$	**11.8**$_{22}$	**9.1**$_{16}$	**24.5**$_{33}$	**24.2**$_{17}$	**7.3**$_{12}$	**5.3**$_{13}$	**4.1**$_{11}$	**5.6**$_{01}$	**22.7**$_{27}$
	RegularizedLoss	**47.0**$_{62}$	**17.1**$_{19}$	**14.9**$_{16}$	**31.8**$_{34}$	**27.7**$_{13}$	**131.9**$_{96}$	**136.4**$_{93}$	**16.9**$_{30}$	**81.9**$_{92}$	**91.8**$_{48}$	**9.2**$_{18}$	**6.3**$_{18}$	**4.7**$_{14}$	**6.7**$_{02}$	**12.7**$_{23}$
	MixedPseudo	**51.9**$_{63}$	**21.0**$_{19}$	**19.9**$_{17}$	**50.2**$_{37}$	**35.8**$_{15}$	**56.8**$_{69}$	**14.6**$_{23}$	**15.6**$_{28}$	**42.9**$_{30}$	**32.5**$_{18}$	**8.1**$_{15}$	**5.7**$_{14}$	**4.3**$_{12}$	**6.0**$_{02}$	**18.6**$_{16}$
	Scribble2Label	**41.7**$_{59}$	**18.2**$_{17}$	**15.3**$_{14}$	**46.4**$_{35}$	**30.4**$_{17}$	**59.7**$_{64}$	**17.8**$_{26}$	**16.3**$_{30}$	**52.8**$_{67}$	**36.7**$_{23}$	**10.2**$_{16}$	**6.5**$_{16}$	**7.3**$_{13}$	**8.0**$_{02}$	**27.3**$_{18}$
	PacingPseudo	17.7$_{30}$	11.1$_{18}$	11.1$_{19}$	16.6$_{19}$	14.1$_{03}$	20.9$_{30}$	5.6$_{09}$	6.5$_{14}$	15.7$_{28}$	12.2$_{06}$	5.7$_{07}$	3.8$_{06}$	3.4$_{06}$	4.3$_{01}$	11.9$_{15}$

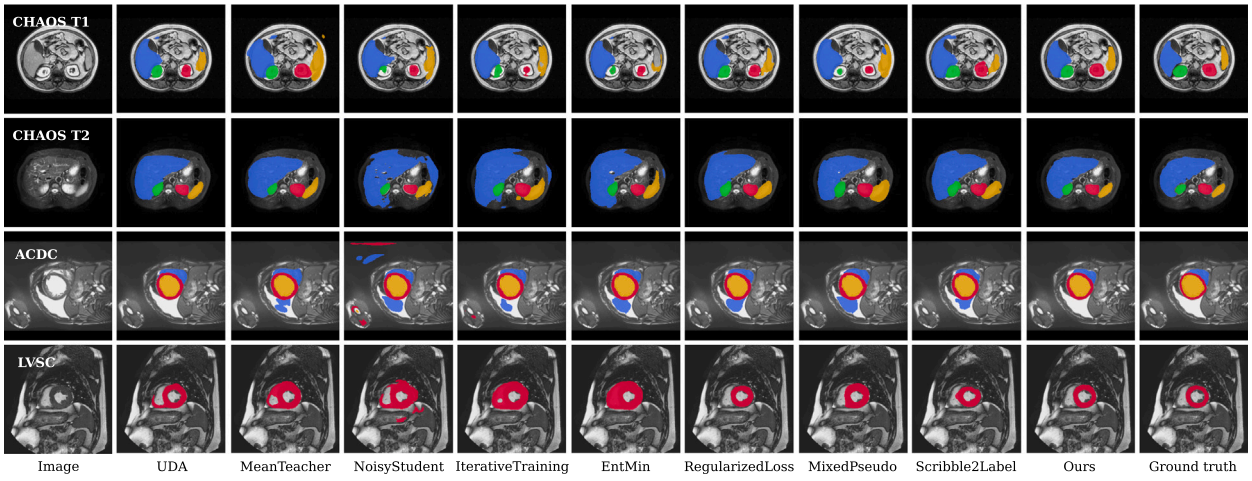


Fig. 5. Qualitative comparison of the segmentations from PacingPseudo and previous methods. They all are supervised solely by scribble annotations.

predictions are not quality pseudo-mask as opposed to our end-to-end regularized ones, and (ii) the stop-gradient operation inherently applied on pseudo-masks hampers its performance, as discussed in Section 5.4.[2] Lastly, PacingPseudo outperforms the iterative NoisyStudent and IterativeTraining and the non-iterative EntMin, RegularizedLoss, MixedPsuedo, and Scribble2Label. While these methods use pre-processed pseudo-masks, regularizers, or interpolated hard pseudo-masks, our method uses regularized high-confidence pseudo-masks to teach a network and shows better performance.

5.1.2. Comparative analysis of computational efficiency

The computational efficiency of the proposed method and previous methods is significantly influenced by two factors: the network architecure and optimization process. The network architecture can be single or dual. The optimization process can be iterative or non-iterative. UDA and MeanTeacher use dual networks and non-iterative optimization as PacingPseudo does. They hence have similar computational costs. However, PacingPseudo performs considerably better than UDA and MeanTeacher, highlighting the superiority of our method. NoisyStudent and IterativeTraining utilize a single network and three-step iterative process with DCRF or noising pre-processing. The iterative process renders them unwieldy. Worse still, their performance is much worse than our PacingPseudo's. The remainder of methods (EntMIN, RegularizedLoss, MixedPseduo, and Scribble2Label) all use a single network and non-iterative optimization. They differ from PacingPseudo in the use of the single network architecture, which can lead to decrease in computational costs, but comes at a cost of performance loss.

5.2. Comparison with the baseline and full supervision

We compare PacingPseudo with the baseline and fully-supervised counterparts. While the baseline trains a network with the partial cross-entropy loss $\mathcal{L}_{pce}$ (1), the fully-supervised counterparts perform network training with either $\mathcal{L}_{ce}$ or with $\mathcal{L}_{ce} + \mathcal{L}_{Dice}$ (an improved implementation), where $\mathcal{L}_{ce}$ and $\mathcal{L}_{Dice}$ denote the cross-entropy loss and Dice loss (Milletari, Navab, & Ahmadi, 2016) respectively and are both based on the ground-truth segmentation masks.

As seen in Table 2, PacingPseudo improves the baseline by large margins in both DSC and HD95. More important, PacingPseudo attains comparable performance with its fully-supervised counterparts on CHAOS T1&T2 and ACDC. To better understand this finding, we compare and analyze their mechanisms. The key difference lies in the mask. While the full supervision can access full masks, our method uses only scribbles and leaves the remaining regions unlabeled. But

[2] The stop-gradient is applied on pseudo-masks as the teacher model simply uses moving-averaged parameters and need not be updated by gradients.

Table 2
Comparison with the baseline and full supervision. Results are based on five-fold cross-validation and shown in MEAN$_{SD}$. The best results are underlined, and the **boldface** indicates the results are statistically different with ours ($p < 0.05$).

	Method	CHAOS T1					CHAOS T2					ACDC				LVSC
		LIV	RK	LK	SPL	*Average*	LIV	RK	LK	SPL	*Average*	RV	MYO	LV	*Average*	MYO
DSC	Baseline $\mathcal{L}_{pce}$	**45.1**$_{37}$	**48.7**$_{34}$	**49.5**$_{36}$	**22.8**$_{29}$	**41.5**$_{11}$	**18.4**$_{22}$	**51.3**$_{37}$	**46.8**$_{38}$	**23.9**$_{32}$	**35.1**$_{14}$	**73.1**$_{32}$	**79.6**$_{20}$	**86.7**$_{21}$	**79.8**$_{06}$	**36.9**$_{18}$
	Full sup. $\mathcal{L}_{ce}$	77.9$_{30}$	64.4$_{37}$	64.7$_{38}$	**53.1**$_{40}$	**65.0**$_{09}$	77.8$_{33}$	75.2$_{35}$	**65.3**$_{41}$	62.8$_{42}$	70.3$_{06}$	**74.8**$_{33}$	83.2$_{19}$	88.7$_{20}$	82.2$_{06}$	**71.9**$_{29}$
	Full sup. $\mathcal{L}_{ce}$+$\mathcal{L}_{Dice}$	78.6$_{30}$	65.9$_{37}$	67.4$_{39}$	**55.8**$_{40}$	**67.0**$_{08}$	77.9$_{34}$	74.5$_{36}$	**67.6**$_{40}$	64.9$_{42}$	71.2$_{05}$	77.4$_{32}$	**84.9**$_{19}$	**89.8**$_{20}$	84.0$_{05}$	**72.0**$_{29}$
	PacingPseudo	80.5$_{27}$	66.5$_{33}$	63.6$_{36}$	61.3$_{36}$	68.0$_{07}$	79.1$_{30}$	78.7$_{30}$	77.1$_{32}$	59.8$_{42}$	73.7$_{07}$	77.7$_{29}$	82.5$_{17}$	88.4$_{19}$	82.9$_{04}$	61.4$_{22}$
HD95	Baseline $\mathcal{L}_{pce}$	**120.5**$_{99}$	**20.2**$_{23}$	**17.0**$_{20}$	**57.1**$_{44}$	**53.7**$_{42}$	**257.6**$_{65}$	**32.7**$_{52}$	**17.2**$_{26}$	**49.8**$_{38}$	**89.3**$_{98}$	**18.5**$_{37}$	**24.5**$_{48}$	**12.5**$_{34}$	**18.5**$_{05}$	**37.7**$_{25}$
	Full sup. $\mathcal{L}_{ce}$	**31.1**$_{52}$	**21.0**$_{37}$	12.8$_{25}$	**21.4**$_{28}$	**21.6**$_{06}$	20.5$_{33}$	**21.1**$_{44}$	**12.8**$_{28}$	13.7$_{23}$	17.0$_{04}$	**7.4**$_{14}$	4.3$_{09}$	**4.0**$_{10}$	**5.2**$_{02}$	7.4$_{14}$
	Full sup. $\mathcal{L}_{ce}$+$\mathcal{L}_{Dice}$	**23.4**$_{39}$	13.5$_{27}$	**8.8**$_{19}$	**21.1**$_{3}$	**16.7**$_{06}$	20.0$_{33}$	**9.7**$_{26}$	8.9$_{19}$	11.7$_{19}$	12.6$_{04}$	5.8$_{11}$	**3.1**$_{05}$	**2.9**$_{05}$	**3.9**$_{01}$	**7.6**$_{15}$
	PacingPseudo	17.7$_{30}$	11.1$_{18}$	11.1$_{19}$	16.6$_{19}$	14.1$_{03}$	20.9$_{30}$	5.6$_{09}$	6.5$_{14}$	15.7$_{28}$	12.2$_{06}$	5.7$_{07}$	3.8$_{06}$	3.4$_{06}$	4.3$_{01}$	11.9$_{15}$

Table 3
Ablation analyses: network components. The best results are underlined, and the **boldface** indicates the results are statistically different with ours ($p < 0.05$).

	Method	CHAOS T1	CHAOS T2	ACDC	LVSC
DSC	Baseline	**41.5**$_{11}$	**35.1**$_{14}$	**79.8**$_{06}$	**36.9**$_{18}$
	EntMin	**53.7**$_{08}$	**55.1**$_{10}$	**81.9**$_{05}$	**51.2**$_{23}$
	+ Memory	**56.8**$_{08}$	**59.6**$_{11}$	82.4$_{05}$	**54.6**$_{22}$
	+ Consistency	68.0$_{07}$	73.7$_{08}$	82.9$_{04}$	61.4$_{22}$
HD95	Baseline	**53.7**$_{42}$	**89.3**$_{98}$	**18.5**$_{05}$	**37.7**$_{25}$
	EntMin	**23.7**$_{08}$	**24.2**$_{17}$	**5.6**$_{01}$	**22.7**$_{27}$
	+ Memory	**23.3**$_{10}$	**22.6**$_{17}$	**5.0**$_{01}$	**14.7**$_{14}$
	+ Consistency	14.1$_{03}$	12.2$_{06}$	4.3$_{01}$	11.9$_{15}$

Table 4
Ablation analyses: with and without stop-gradient. The best results are underlined.

	Method	CHAOS T1	CHAOS T2	ACDC	LVSC
DSC	w/ stop-grad.	53.8$_{14}$	59.7$_{12}$	82.6$_{04}$	52.8$_{25}$
	w/o stop-grad.	68.0$_{07}$	73.7$_{08}$	82.9$_{04}$	61.4$_{22}$
HD95	w/ stop-grad.	26.2$_{18}$	18.1$_{09}$	5.4$_{01}$	21.1$_{24}$
	w/o stop-grad.	14.1$_{03}$	12.2$_{06}$	4.3$_{01}$	11.9$_{15}$

we compensate this deficiency by using regularized high-confidence pseudo-masks (as opposed to one-hot full masks) to teach a network. Another difference is the network architecture. While the full supervision uses a single network, our method leverages weigh-sharing networks which give a natural advantage that augmentations in the predicted-mask branch can always act to expand the training set. These factors could have contributed to narrowing their performance gap. Beyond the above finding, our method, however, does not catch up with its fully-supervised counterparts on LVSC. This may be because scribble annotations provide limited boundary constraint to control the extent of the thin, ringed myocardium, which is the limitations of our method discussed in Section 5.7. But, overall, PacingPseudo could be promising to bridge the gap between the scribble-supervised and fully-supervised segmentation.

5.3. Ablation: Network components

We study the efficacy of the network components in this subsection. In Table 3, EntMin incorporates the entropy regularization loss $\mathcal{L}_{ent}$ (3) into the baseline, then the "+Memory" entry implements the memory bank setting in an auxiliary path, and the "+Consistency" uses the siamese architecture and adds the consistency regularization loss $\mathcal{L}_{cr}$ (2).

As shown in Table 3, it is surprising that simply adding the loss $\mathcal{L}_{ent}$ (the EntMin) brings considerable improvements over the baseline. We conjecture that this is because the loss $\mathcal{L}_{pce}$ in the baseline constrains only a few pixels to be one-hot, which leads to uncertainty. Adding the loss $\mathcal{L}_{ent}$, however, incorporates all pixels and gradually regularizes them to be confident (Dolz et al., 2021), which in effect encourages a network to learn a beneficial low-density decision boundary (Chapelle & Zien, 2005) that could separate classes clearly and hence improve validation accuracy. Then, the "+Memory" can further boost the performance. This supports our motivation of introducing a source of ensemble features to complement scarce supervision. The "+Consistency" brings another performance leaps in both metrics. This validates the effectiveness of using a stream of regularized high-confidence pseudo-masks to reinforce network learning in a non-iterative manner (via a siamese architecture).

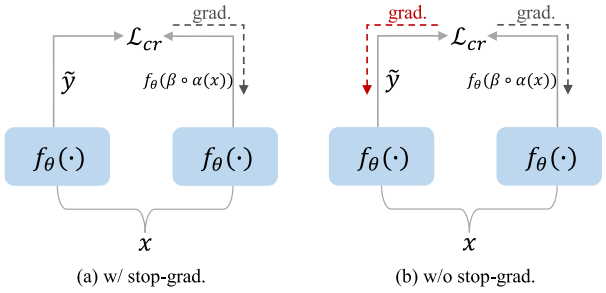


Fig. 6. Stop-gradient on Pseudo-Masks. (a) With the stop-gradient, the gradients of the pseudo-mask $\tilde{y}$ are prevented from back-propagation, while (b) without the stop-gradient, the gradients of $\tilde{y}$ are back-propagated. The above framework illustrates $\mathcal{L}_{cr}$ only for clarity.

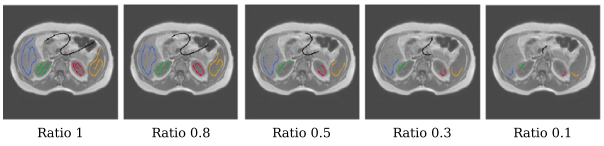
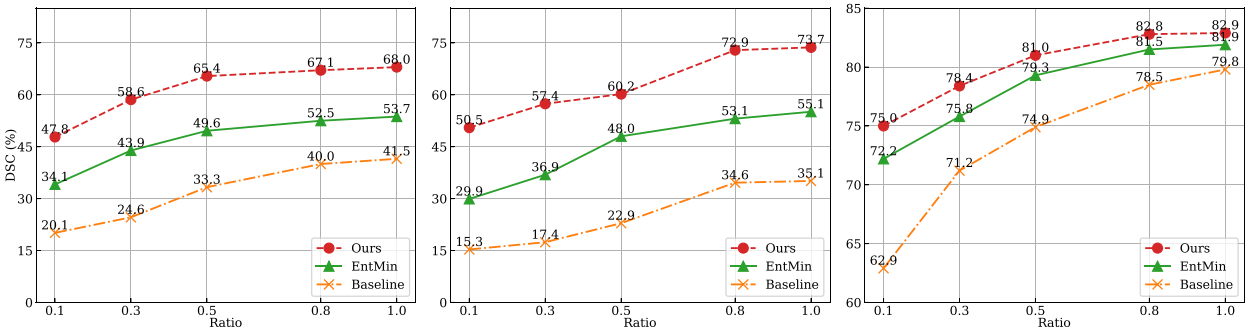


Fig. 7. Scribbles of different lengths. The scribbles are shorten by iteratively pruning their endpoints. It can be observed that scribbles gradually shrink to small line segments.

5.4. Ablation: Stop-gradient on pseudo-masks

Regarding the designs of pseudo-masks, we have discussed why artificial operations (e.g., thresholding and sharpening) are not effective in Section 5.1. In this subsection, we investigate another question, whether or not to apply the stop-gradient operation on pseudo-masks. Specifically, we design two settings: one with the stop-gradient on the pseudo-mask $\tilde{y}$ in the loss $\mathcal{L}_{cr}$ (2) (Luo, Hu, et al., 2022; Xie, Dai, et al., 2020; Yu et al., 2019) and the other without the stop-gradient (ours) (see Fig. 6).

The two settings present different optimization trajectories. With the stop-gradient, $\tilde{y}$ is treated as a constant and the network $f_\theta(\cdot)$ only receives the gradients of $f_\theta(\beta \circ \omega(x))$ in $\mathcal{L}_{cr}$. On the other hand, without stop-gradient, the network $f_\theta(\cdot)$ jointly receives the gradients of both



(a) Dice scores by ratios on CHAOS T1 (b) Dice scores by ratios on CHAOS T2 (c) Dice scores by ratios on ACDC

Fig. 8. Robustness to different scribble ratios. The ratio quantifies the pixel number in the pruned scribbles to that in the original scribbles.

$\tilde{y}$ and $f_\theta(\beta\circ\omega(x))$ in $\mathcal{L}_{cr}$.[3] As seen in Table 4, the entry without stop-gradient (ours) performs better than the one with stop-gradient. Our explanation for this finding is that without stop-gradient, the network is jointly updated by $\mathcal{L}_{cr}$ to produce consistent $\tilde{y}$ and $f_\theta(\beta\circ\omega(x))$, which could encourage beneficial smooth optimization via the consistency regularization. In contrast, with stop-gradient, the network is updated by $\mathcal{L}_{cr}$ to produce solely better $f_\theta(\beta\circ\omega(x))$, which could lead to misalignment between $\tilde{y}$ and $f_\theta(\beta\circ\omega(x))$ and undesirably impede the optimization.

5.5. Strengths of the further augmentation operation

In this subsection, we verify whether the further augmentation operation $\beta(\cdot)$ can influence performance. Our assumption is that lowering the augmentation strength, which in effect contracts the neighborhood of the training set, would lead to worse performance. To gather evidence, we gradually decrease the strength δ from 1 (the default setting) to 1/2, 1/4, and 1/8. Note a smaller value of δ shrinks the range of a uniform distribution from which the augmentation magnitude is drawn, and at the extreme of $\delta = 0$, no further augmentation is applied.

As seen in Table 5, decreasing the strength δ apparently deteriorates performance in both metrics on CHAOS T1&T2 and LVSC. These results support our assumption and suggest the important role of the further augmentation operation in our method. However, on ACDC, while it can be observed that the HD95 results present a similar declining trend, the DSC results are quite stable. This could be because our performance in DSC on ACDC is easily saturated, since the baseline already achieves competitive results (see Table 3).

5.6. Robustness to scribble lengths

In this subsection, we study how the number of scribble pixels covering over targets influences performance, to reassure a concern that is high occupation of target regions by scribbles can easily produce satisfactory performance. Specifically, we evaluate our method's robustness to scribble length. We train a network supervised by scribble annotations of 0.1×, 0.3×, 0.5× and 0.8× original scribble pixels (Fig. 7). To shorten a scribble, we detect and prune its endpoints; if the endpoint does not exist, we prune a random scribble pixel. This process is iterated until a length requirement (e.g., 0.8× original scribble pixels) is met. Experiments are conducted on CHAOS T1&T2 and ACDC with real-world scribble annotations. We compare our method with both the baseline and EntMin (seen as the improved baseline).

[3] A derivation is given: $\nabla_\theta \mathcal{L}_{cr} = -\frac{1}{N}\sum_{i=0}^{N-1}\sum_{k=0}^{K-1}(\log y'_{ik}\nabla_\theta \tilde{y}_{ik} + \frac{\tilde{y}_{ik}}{y'_{ik}}\nabla_\theta y'_{ik})$, where $y' = \texttt{softmax}(f_\theta(\beta\circ\omega(x)))$. With stop-gradient, the first term in $\nabla_\theta \mathcal{L}_{cr}$ is removed, which is the derivatives of the pseudo-mask pixel $\tilde{y}_{ik}$ with respect to the parameters θ.

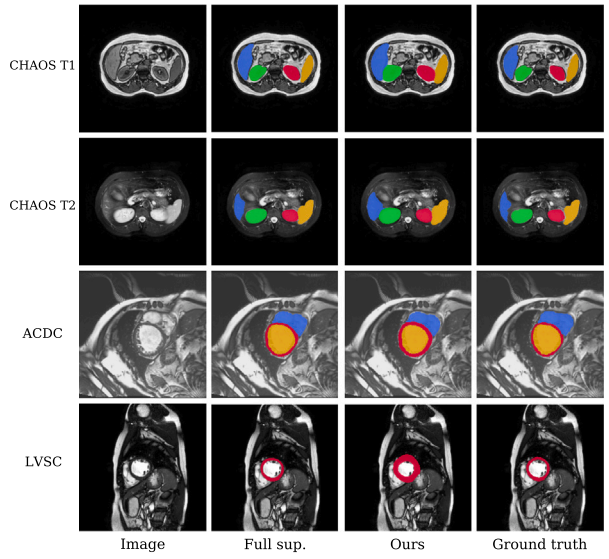


Fig. 9. Limitations of our scribble-supervised segmentation method. Some of our results are expanded outside the ground-truth boundaries.

As illustrated in Fig. 8, our method consistently outperforms the baseline and EntMin under different scribble ratios. Furthermore, although it can be observed that a smaller ratio produce a lower DSC result, this trend is stable at certain intervals. To be specific, the DSC result drops by a trivial margin from 1.0× to 0.5×scribble pixels on CHAOS T1, and from 1.0× to 0.8×scribble pixels on CHAOS T2 and ACDC. This finding suggests that, within certain worse variations in scribble quality, our method can still achieve good performance.

5.7. Limitations

While PacingPseudo achieves overall satisfactory results, there still remain limitations. Some segmentation results tend to be expanded outside the ground-truth boundaries. As seen in Fig. 9, on CHAOS T1&T2, while the segmentations of the full supervision distinguish different organs clearly, our results fail to provide accurate boundaries in the adjacent regions between different organs (e.g., the liver and right kidney). On ACDC and LVSC, while the thin myocardium is well delineated by the full supervision, our method generates much thicker segmentation regions. We conjecture these issues could be due to the inherent limitations of the insufficient supervision from the scribble annotations.

Table 5
Comparison of different augmentation strengths. The best results are underlined.

	Dataset	1/8	1/4	1/2	1
DSC	CHAOS T1	58.2$_{09}$	61.3$_{08}$	63.7$_{09}$	<u>68.0$_{07}$</u>
	CHAOS T2	62.5$_{08}$	66.5$_{07}$	70.1$_{07}$	<u>73.7$_{08}$</u>
	ACDC	82.8$_{04}$	82.7$_{05}$	<u>83.0$_{05}$</u>	82.9$_{04}$
	LVSC	54.1$_{22}$	56.5$_{23}$	58.4$_{24}$	<u>61.4$_{22}$</u>
HD95	CHAOS T1	17.6$_{05}$	16.4$_{04}$	15.3$_{03}$	<u>14.1$_{03}$</u>
	CHAOS T2	14.2$_{06}$	12.9$_{05}$	12.4$_{06}$	<u>12.2$_{06}$</u>
	ACDC	4.7$_{01}$	4.6$_{01}$	4.5$_{01}$	<u>4.3$_{01}$</u>
	LVSC	14.8$_{16}$	14.4$_{18}$	13.6$_{20}$	<u>11.9$_{15}$</u>

One possible solution to these issues is constraining unlabeled pixels' predictions based on low-level image information. Dorent et al. (2020) and Tang, Perazzi, et al. (2018) propose regularizers based on DCRFs. Kolesnikov and Lampert (2016) devise a loss function that compels a network to produce predictions analogous to DCRF-processed segmentations. These methods enable neural networks to learn low-level image information. PacingPseudo could potentially benefit from learning low-level image information and produce more precise boundaries.

6. Conclusion

We propose an effective non-iterative method for scribble-supervised segmentation in medical imaging. Our method attains a non-iterative training paradigm by means of a weight-sharing siamese architecture, wherein pseudo-masks reinforce network learning during training. Inspired by insights in consistency training, our designs to boost performance include the entropy regularization to obtain high-confidence pseudo-masks and the distorted augmentations to create discrepancy for consistency regularization. Besides, to complement scarce labeled pixels, we devise a memory bank that introduces extra cross-image ensemble features. Experimental results show that our method can bridge the gap between scribble-supervised and fully-supervised segmentation to some extent, and proves robust even to some undesirable scribbles. Thanks to its simplicity, the proposed method has the potential to be extended to 3D by using volumetric scribble-annotating approaches.

CRediT authorship contribution statement

Zefan Yang: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing – original draft, Visualization. **Di Lin:** Methodology, Validation, Supervision. **Dong Ni:** Resources, Funding acquisition. **Yi Wang:** Conceptualization, Methodology, Resources, Writing – review & editing, Supervision, Project administration, Funding acquisition.

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.

Data availability

We used publicly available datasets. We have cited these datasets in our manuscript.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grants 62071305, 61701312, 81971631 and 62171290, in part by the Guangdong Basic and Applied Basic Research Foundation under Grants 2022A1515011241, and in part by the Shenzhen Science and Technology Program (No. SGDX 20201103095613036).

References

Bernard, O., Lalande, A., Zotti, C., Cervenansky, F., Yang, X., Heng, P. A., et al. (2018). 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.

Berthelot, D., Carlini, N., Cubuk, E. D., Kurakin, A., Sohn, K., Zhang, H., et al. (2019). Remixmatch: Semi-supervised learning with distribution alignment and augmentation anchoring. arXiv preprint arXiv:1911.09785.

Berthelot, D., Carlini, N., Goodfellow, I., Papernot, N., Oliver, A., & Raffel, C. A. (2019). Mixmatch: A holistic approach to semi-supervised learning. *Advances in Neural Information Processing Systems*, *32*.

Bortsova, G., Dubost, F., Hogeweg, L., Katramados, I., & Bruijne, M. d. (2019). Semi-supervised medical image segmentation via learning consistency under transformations. In *International conference on medical image computing and computer-assisted intervention* (pp. 810–818). Springer.

Boykov, Y., & Kolmogorov, V. (2004). An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *26*(9), 1124–1137.

Bromley, J., Guyon, I., LeCun, Y., Säckinger, E., & Shah, R. (1993). Signature verification using a "siamese" time delay neural network. *Advances in Neural Information Processing Systems*, *6*.

Can, Y. B., Chaitanya, K., Mustafa, B., Koch, L. M., Konukoglu, E., & Baumgartner, C. F. (2018). Learning to segment medical images with scribble-supervision alone. In *Deep learning in medical image analysis and multimodal learning for clinical decision support* (pp. 236–244).

Chapelle, O., & Zien, A. (2005). Semi-supervised classification by low density separation. In *International workshop on artificial intelligence and statistics* (pp. 57–64). PMLR.

Chen, X., & He, K. (2021). Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 15750–15758).

Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020). A simple framework for contrastive learning of visual representations. In *International conference on machine learning* (pp. 1597–1607). PMLR.

Cubuk, E. D., Zoph, B., Shlens, J., & Le, Q. V. (2020). Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops* (pp. 702–703).

Dai, J., He, K., & Sun, J. (2015). Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In *Proceedings of the IEEE international conference on computer vision* (pp. 1635–1643).

Dolz, J., Desrosiers, C., & Ayed, I. B. (2021). Teach me to segment with mixed supervision: Confident students become masters. In *International conference on information processing in medical imaging* (pp. 517–529). Springer.

Dorent, R., Joutard, S., Shapey, J., Bisdas, S., Kitchen, N., Bradford, R., et al. (2020). Scribble-based domain adaptation via co-segmentation. In *Medical image computing and computer assisted intervention–MICCAI 2020: 23rd international conference, Lima, Peru, October 4–8, 2020, proceedings, part I 23* (pp. 479–489). Springer.

Grandvalet, Y., & Bengio, Y. (2004). Semi-supervised learning by entropy minimization. *Advances in Neural Information Processing Systems*, *17*.

Grill, J. B., Strub, F., Altché, F., Tallec, C., Richemond, P., Buchatskaya, E., et al. (2020). Bootstrap your own latent-a new approach to self-supervised learning. *Advances in Neural Information Processing Systems*, *33*, 21271–21284.

Han, Q., Wang, H., Hou, M., Weng, T., Pei, Y., Li, Z., et al. (2023). HWA-SegNet: Multi-channel skin lesion image segmentation network with hierarchical analysis and weight adjustment. *Computers in Biology and Medicine*, *152*, Article 106343.

He, K., Fan, H., Wu, Y., Xie, S., & Girshick, R. (2020). Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 9729–9738).

Isensee, F., Jaeger, P. F., Kohl, S. A., Petersen, J., & Maier-Hein, K. H. (2021). nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation. *Nature Methods*, *18*(2), 203–211.

Kavur, A. E., Gezer, N. S., Barış, M., Aslan, S., Conze, P.-H., Groza, V., et al. (2021). CAOS challenge-combined (CT-MR) healthy abdominal organ segmentation. *Medical Image Analysis*, *69*, Article 101950.

Kervadec, H., Dolz, J., Tang, M., Granger, E., Boykov, Y., & Ayed, I. B. (2019). Constrained-CNN losses for weakly supervised segmentation. *Medical Image Analysis*, *54*, 88–99.

Khoreva, A., Benenson, R., Hosang, J., Hein, M., & Schiele, B. (2017). Simple does it: Weakly supervised instance and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 876–885).

Kolesnikov, A., & Lampert, C. H. (2016). Seed, expand and constrain: Three principles for weakly-supervised image segmentation. In *European conference on computer vision* (pp. 695–711). Springer.

Krähenbühl, P., & Koltun, V. (2011). Efficient inference in fully connected crfs with gaussian edge potentials. *Advances in Neural Information Processing Systems*, *24*.

Laradji, I., Rodriguez, P., Manas, O., Lensink, K., Law, M., Kurzman, L., et al. (2021). A weakly supervised consistency-based learning method for covid-19 segmentation in ct images. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision* (pp. 2453–2462).

Lee, H., & Jeong, W. K. (2020). Scribble2label: Scribble-supervised cell segmentation via self-generating pseudo-labels with consistency. In *International conference on medical image computing and computer-assisted intervention* (pp. 14–23). Springer.

Li, X., Yu, L., Chen, H., Fu, C. W., Xing, L., & Heng, P. A. (2020). Transformation-consistent self-ensembling model for semisupervised medical image segmentation. *IEEE Transactions on Neural Networks and Learning Systems*, *32*(2), 523–534.

Lin, D., Dai, J., Jia, J., He, K., & Sun, J. (2016). Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3159–3167).

Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., et al. (2017). A survey on deep learning in medical image analysis. *Medical Image Analysis*, *42*, 60–88.

Luo, X., Hu, M., Liao, W., Zhai, S., Song, T., Wang, G., et al. (2022). Scribble-supervised medical image segmentation via dual-branch network and dynamically mixed pseudo labels supervision. In *Medical image computing and computer assisted intervention* (pp. 528–538).

Luo, X., Liao, W., Xiao, J., Chen, J., Song, T., Zhang, X., et al. (2022). WORD: A large scale dataset, benchmark and clinical applicable study for abdominal organ segmentation from CT image. *Medical Image Analysis*, Article 102642.

Milletari, F., Navab, N., & Ahmadi, S.-A. (2016). V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 Fourth international conference on 3D vision* (pp. 565–571). IEEE.

Papandreou, G., Chen, L.-C., Murphy, K. P., & Yuille, A. L. (2015). Weakly-and semi-supervised learning of a deep convolutional network for semantic image segmentation. In *Proceedings of the IEEE international conference on computer vision* (pp. 1742–1750).

Patel, G., & Dolz, J. (2022). Weakly supervised segmentation with cross-modality equivariant constraints. *Medical Image Analysis*, *77*, Article 102374.

Qi, A., Zhao, D., Yu, F., Heidari, A. A., Wu, Z., Cai, Z., et al. (2022). Directional mutation and crossover boosted ant colony optimization with application to COVID-19 X-ray image segmentation. *Computers in Biology and Medicine*, *148*, Article 105810.

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* (pp. 234–241). Springer.

Roth, H. R., Yang, D., Xu, Z., Wang, X., & Xu, D. (2021). Going to extremes: weakly supervised medical image segmentation. *Machine Learning and Knowledge Extraction*, *3*(2), 507–524.

Shen, D., Wu, G., & Suk, H.-I. (2017). Deep learning in medical image analysis. *Annual Review of Biomedical Engineering*, *19*, 221.

Snell, J., Swersky, K., & Zemel, R. (2017). Prototypical networks for few-shot learning. *Advances in Neural Information Processing Systems*, *30*.

Sohn, K., Berthelot, D., Carlini, N., Zhang, Z., Zhang, H., Raffel, C. A., et al. (2020). Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *Advances in Neural Information Processing Systems*, *33*, 596–608.

Suinesiaputra, A., Cowan, B. R., Al-Agamy, A. O., Elattar, M. A., Ayache, N., Fahmy, A. S., et al. (2014). A collaborative resource to build consensus for automated left ventricular segmentation of cardiac MR images. *Medical Image Analysis*, *18*(1), 50–62.

Tajbakhsh, N., Jeyaseelan, L., Li, Q., Chiang, J. N., Wu, Z., & Ding, X. (2020). Embracing imperfect datasets: A review of deep learning solutions for medical image segmentation. *Medical Image Analysis*, *63*, Article 101693.

Tang, M., Djelouah, A., Perazzi, F., Boykov, Y., & Schroers, C. (2018). Normalized cut loss for weakly-supervised CNN segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1818–1827).

Tang, M., Perazzi, F., Djelouah, A., Ben Ayed, I., Schroers, C., & Boykov, Y. (2018). On regularized losses for weakly-supervised cnn segmentation. In *Proceedings of the European conference on computer vision* (pp. 507–522).

Uslu, F., & Bharath, A. A. (2023). TMS-net: A segmentation network coupled with a run-time quality control method for robust cardiac image segmentation. *Computers in Biology and Medicine*, *152*, Article 106422.

Valvano, G., Leo, A., & Tsaftaris, S. A. (2021). Learning to segment from scribbles using multi-scale adversarial attention gates. *IEEE Transactions on Medical Imaging*, *40*(8), 1990–2001.

Wang, Y., Dou, H., Hu, X., Zhu, L., Yang, X., Xu, M., et al. (2019). Deep attentive features for prostate segmentation in 3D transrectal ultrasound. *IEEE Transactions on Medical Imaging*, *38*(12), 2768–2778.

Xie, Q., Dai, Z., Hovy, E., Luong, T., & Le, Q. (2020). Unsupervised data augmentation for consistency training. *Advances in Neural Information Processing Systems*, *33*, 6256–6268.

Xie, Q., Luong, M. T., Hovy, E., & Le, Q. V. (2020). Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 10687–10698).

Yu, L., Wang, S., Li, X., Fu, C.-W., & Heng, P. A. (2019). Uncertainty-aware self-ensembling model for semi-supervised 3D left atrium segmentation. In *International conference on medical image computing and computer-assisted intervention* (pp. 605–613). Springer.

Zhang, K., & Zhuang, X. (2022). CycleMix: A holistic strategy for medical image segmentation from scribble supervision. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 11656–11665).

Zhao, Z., Yang, L., Zheng, H., Guldner, I. H., Zhang, S., & Chen, D. Z. (2018). Deep learning based instance segmentation in 3D biomedical images using weak annotation. In *International conference on medical image computing and computer-assisted intervention* (pp. 352–360). Springer.

Zou, Y., Zhang, Z., Zhang, H., Li, C. L., Bian, X., Huang, J. B., et al. (2020). Pseudoseg: Designing pseudo labels for semantic segmentation. arXiv preprint arXiv:2010.09713.