Computers in Biology and Medicine 154 (2023) 106585

Contents lists available at ScienceDirect

Computers in Biology and Medicine

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

Error-Correcting Mean-Teacher: Corrections instead of consistency-targets applied to semi-supervised medical image segmentation

Robert Mendel [a,*], David Rauber [a], Luis A. de Souza Jr. [c], João P. Papa [d], Christoph Palm [a,b]

[a] Regensburg Medical Image Computing (ReMIC), Ostbayerische Technische Hochschule Regensburg (OTH Regensburg), Regensburg, Germany
[b] Regensburg Center of Health Sciences and Technology (RCHST), OTH Regensburg, Regensburg, Germany
[c] Computer Science Department, Federal University of São Carlos, São Carlos, Brazil
[d] Department of Computing, São Paulo State University, Bauru, Brazil

ARTICLE INFO

Keywords:
Semi-supervised
Segmentation
Mean-Teacher
Pseudo-labels
Medical imaging

ABSTRACT

Semantic segmentation is an essential task in medical imaging research. Many powerful deep-learning-based approaches can be employed for this problem, but they are dependent on the availability of an expansive labeled dataset. In this work, we augment such supervised segmentation models to be suitable for learning from unlabeled data. Our semi-supervised approach, termed Error-Correcting Mean-Teacher, uses an exponential moving average model like the original Mean Teacher but introduces our new paradigm of error correction. The original segmentation network is augmented to handle this secondary correction task. Both tasks build upon the core feature extraction layers of the model. For the correction task, features detected in the input image are fused with features detected in the predicted segmentation and further processed with task-specific decoder layers. The combination of image and segmentation features allows the model to correct present mistakes in the given input pair. The correction task is trained jointly on the labeled data. On unlabeled data, the exponential moving average of the original network corrects the student's prediction. The combined outputs of the students' prediction with the teachers' correction form the basis for the semi-supervised update. We evaluate our method with the 2017 and 2018 Robotic Scene Segmentation data, the ISIC 2017 and the BraTS 2020 Challenges, a proprietary Endoscopic Submucosal Dissection dataset, Cityscapes, and Pascal VOC 2012. Additionally, we analyze the impact of the individual components and examine the behavior when the amount of labeled data varies, with experiments performed on two distinct segmentation architectures. Our method shows improvements in terms of the mean Intersection over Union over the supervised baseline and competing methods. Code is available at https://github.com/CloneRob/ECMT.

1. Introduction

Neural Networks enabled substantial advances in various domains like computer vision and natural language processing. The countless algorithmic advances were in part pushed by Krizhevsky et al. [1] with significant improvements in classification accuracy on the ImageNet dataset [2], the advances in automatic differentiation and GPU computing. Apart from the technological progress that enabled pursuing ideas infeasible before, an essential factor still is the availability of a large labeled dataset for the underlying task. For general purpose object detection or segmentation, such large datasets are available [1,3,4]. However, in some vision domains, the availability still poses a problem.

In medical imaging, data is commonly sparse, and its labeling is costly. The sparsity applies to an even greater extent to semantic segmentation problems. The manual annotation of such image data is time-consuming, for it cannot be outsourced and must be conducted by medical experts. Hence, medical image segmentation datasets have rarely approached the expansiveness and quality of more general-purpose datasets, and algorithms tackling these problems have to potentially deal with scarce or noisy annotations [5].

Medical image segmentation problems appear in many different forms. The tasks range from binary classification for detecting and identifying dermoscopic lesions from 2D images to delineating and assessing glioblastoma in MRI scans. There is a wide range of neural network architectures commonly applied in the medical domain, ranging from variations of the popular U-Net [6] to more general-purpose segmentation architectures [7].

* Corresponding author.
E-mail address: robert1.mendel@oth-regensburg.de (R. Mendel).

https://doi.org/10.1016/j.compbiomed.2023.106585
Received 29 June 2022; Received in revised form 25 December 2022; Accepted 22 January 2023
Available online 24 January 2023

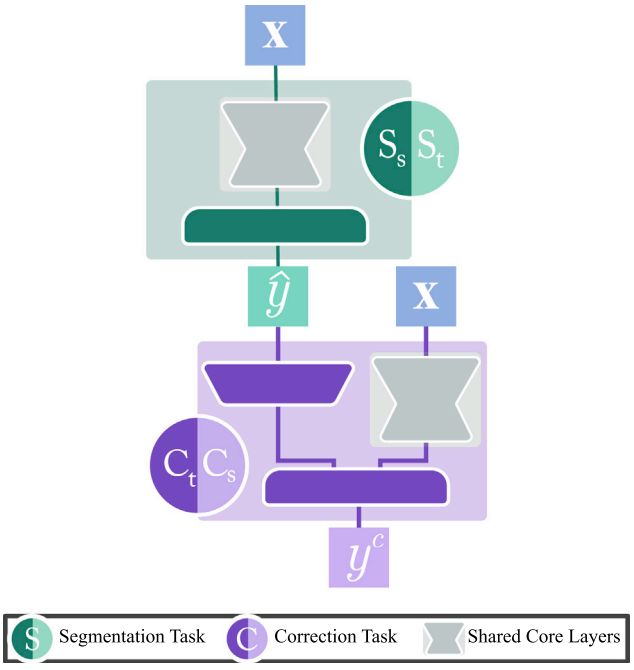

Fig. 1. A high-level overview of the multi-task architecture. In both tasks, a large majority of the parameters are shared. The correction task, C_s and C_t for student and teacher, respectively, contains independent output layers as well as an additional set of input layers. During the correction task, the model operates on features from both the original image and the predicted segmentation learned for the segmentation task.

For a semi-supervised segmentation approach to be widely applicable in the medical domain, it is essential to function independently of the network architecture or the data domain, be it multi-class 2D or binary 3D datasets. In the end, segmentation models are tools to improve diagnostics or therapeutic treatments [8,9]. Their prevalence will correlate with the quality and robustness of their outputs, and we believe that semi-supervised methods can optimize both of these metrics.

In this work, we propose to learn from unlabeled data with Error-Correcting Mean-Teacher (ECMT). In ECMT, the base segmentation network is extended to handle the task of spotting and correcting errors in a predicted segmentation. Semantically, our model can be categorized into three parts, shown in Fig. 1:

- A base feature extractor that is active in both tasks.
- The segmentation task, where a segmentation decoder is applied to the output of the feature extractor, producing a segmentation map with depth N for N labels.
- The correction task, where the model operates on both the image and the predicted segmentation.

For the correction task, the model is extended with an additional set of input and output layers. The input layers convolve the predicted segmentation with depth N and concatenate the representation with those returned by the shared feature extractor (Fig. 1). The final decoding layers then produce a segmentation map with depth $N + 1$, where the additional channel is used to indicate a matching between the predicted segmentation and the content of the input image. Therefore, the original N classes are used to correct perceived inaccuracies in the given segmentation, while class $N + 1$ signifies agreement.

Mean-Teacher (MT) [10] models have proven to provide accurate targets for semi-supervised tasks. However, those targets guiding the optimization problem are most commonly based on consistency regularization. Within ECMT, the Mean-Teacher, an exponential moving average of the weights of the primary model, builds pseudo-labels from both the outputs of the segmentation and correction tasks. In ECMT, if the teacher's output has high entropy, the student will not be optimized to match this distribution. Thus, ECMT represents a shift from the paradigm of consistency regularization towards error-correction. Depending on one's viewpoint, it is possible to interpret all consistency regularization or pseudo-labeling approaches as error correction. As long as the outputs between student and teacher do not perfectly match, following the teacher's prediction always represents a form of error correction. However, the proposed approach distinctly differs from other methods by including both image and prediction as inputs for the correction model enabling the nuanced distinction between accurate and inaccurate regions. Fig. 2 highlights an abstract view of these differences, showing the extension to map from labeled space to correction space, which is missing from consistency-based approaches.

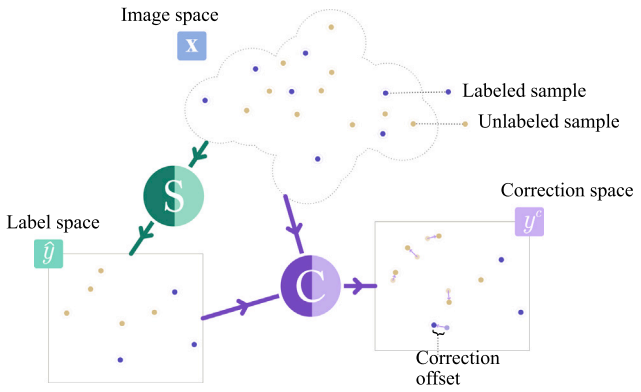

Fig. 2. Conceptual view of the data flow in ECMT. MT models only map from image to label space. In ECMT the corrector operates on a combined representation of image and label spaces and predicts the offset from the given inputs to the predicted *truth*.

By using both labeled and unlabeled data, our framework allows for efficient utilization of all data available, which is especially important in domains where data gathering is challenging.

In summary, our contributions are as follows:

- Shifting the Mean-Teacher from consistency regularization to error correction for semi-supervised segmentation.
- Proposing a multi-task architecture that leverages a shared feature extractor for the segmentation and correction tasks.
- Utilization of fine-grained error correction maps to optimize the secondary task on labeled data.
- Extensive evaluation on 2D and 3D medical datasets and comparisons between competing semi-supervised learning concepts.
- Analysis of the learned representations to reassure that ECMT does not just reinforce the supervised signal.

We have introduced the concept of error-correction [11] recently. Now, we propose a way to merge it with the Mean-Teacher model for the first time and additionally transfer it to the medical imaging domain. We provide granular evidence that the combination of segmentation and correction delivers a more accurate approximation of the unknown truth. Moreover, our analysis of the learned representations shows that our model can effectively incorporate this information. Further, we provide an ablation study of the main hyperparameters of the model and show how increasing the number of labeled samples develops the model's accuracy.

2. Related work

Besides approaches of supervised segmentation, within the area of semi-supervised segmentation the concept of Error-Correction is one of the essential parts of the proposed approach. Therefore, the works discussed in [11] are equally relevant in this setting. Additionally the techniques for consistency regularization are highlighted in the following.

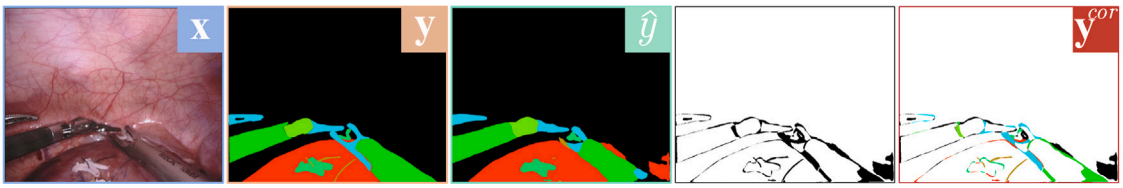

Fig. 3. Assigning a new label $N+1$ where the segmentation $\hat{\mathbf{y}}$ matches the ground-truth $\mathbf{y}$, while keeping the labels of $\mathbf{y}$ where the prediction diverges, results in the targets for the correction task $\mathbf{y}^{cor}$. (From left to right: input image, ground truth, predicted segmentation, difference between ground-truth and segmentation, and correction map).

2.1. Supervised semantic segmentation

Currently, segmentation models are mostly based on some form of spatial pyramid pooling [12–14] or show an encoder–decoder [6,15] structure or recently have started to include attention into the architecture [16,17]. The underlying design decisions for both of those architectural choices build on the fully convolution composition, proposed by Shelhamer et al. [18].

Applying pooling operations at various scales [14] results in output features that depict the image's content in diverse resolutions. The DeepLab family [13,19,20] introduced an Atrous Spatial Pyramid Pooling (ASPP) module. Here, dilated convolutions [21] track representations at different scales without changing the features' spatial resolution. In recent iterations of the DeepLab model [20], both the encoder–decoder and ASPP approaches were combined. The connection between the features detected in early and later stages of the model result in sharper object boundaries.

Transformer based approaches in 2D and 3D medical imaging often rely on an hybrid architecture [22,23] or take inspiration from the skip connections from established convolutional models [24].

A variety of the discussed segmentation models have been applied to the Robotic Scene Segmentation Challenge [7] or dermoscopic datasets like the ISIC 2017 Challenge [25]. Since their introduction U-Nets were among the most commonly used models in medical image segmentation. Especially in 3D imaging problems variants of the U-Net have achieved state of the art results on a large number of datasets [26]. But, challenges like [7] show that architectures that are more commonly used for general segmentation tasks are at least as competitive.

Apart from neural network based approaches, there is a large body of work based on, for example, graph based algorithms [27], nature-inspired methods [28,29], or hybrid applications that combine feature engineering with deep learning models [30].

2.2. Semi-supervised segmentation

For semi-supervised segmentation, labeled as well as unlabeled samples can be employed to optimize the model. Two research directions are starting to appear that operate on different principles.

Methods that feature an adversarial component, inspired by the wide-spread framework of Generative Adversarial Networks [31], incorporate a second discriminator network that judges a given image with varying granularity. Souly et al. [32] transferred the GAN framework to semi-supervised segmentation by choosing a segmentation architecture for the discriminator. Each pixel is classified as either generated or as one of the true classes in their approach.

Luc et al. [33] apply adversarial regularization in a purely supervised setting. The segmentation network is constrained to predict outputs a discriminator interprets as *real*. Unlike [32], Hung et al. [34] treat the segmentation network as the generator, and add a new discriminator. The Fully Convolutional Discriminator evaluates the ground-truth as well as the segmentation and judges at a pixel level if the input is *real* or *fake*, respectively. Similar to [33], an adversarial term is added to the supervised objective. On unlabeled data, depending on the discriminator accepting regions as *real*, these predictions are used to minimize a spatial weighted cross-entropy loss. Nie et al. [35] adapt this approach for the medical domain and show promising results on an MRI dataset. Wang et al. [36] combine a number of concepts from GAN literature to generate noisy pseudo-labels for semi-supervised segmentation.

Another direction that is explored for semi-supervised learning is consistency regularization. In most cases, the fundamental idea is to constrain the model to produce consistent outputs on labeled as well as unlabeled data, independent to any noise or augmentations that are added to the input. With Temporal Ensembling [37], an exponential moving average (EMA) for each prediction on the labeled and unlabeled samples is updated after every training step. These ensembled predictions are then used as targets for the consistency regularization. Variations of this concept have been applied to histopathology image analysis [38]. The Mean-Teacher (MT) framework [10] shifts the EMA from the predictions to the model itself. This model ensemble represents the role of the teacher providing the targets for the consistency regularization. In the medical domain, the MT framework has been used to improve the quality of brain-lesion segmentations [39] or augmented with label propagation to nuclei classification [40]. Augmented with a rich set of perturbations on the input data of the student and teacher models, Xiao et al. [41] effectively apply this framework from skin lesion segmentation to optic disc segmentation in fundus images. Yu et al. [42] add uncertainty-awareness to the MT framework. By repeated forward passes with independent dropout and noise, they filter the regions of the output that show large variances and suppress them in the consistency regularization. A variety of different forms of uncertainty-estimation have been explored for medical image segmentation, from models that build the estimate from an auto-encoded label representation [43] to multiple decoders [44] or by the inclusion of several learners [45]. Apart from MT models in consistency-regularization, Ouali et al. [46] augment the segmentation architecture with several auxiliary decoders to introduce perturbations to the output, and enforces consistency between the main and auxiliary predictions.

Guided Collaborative Training (GCT) [47] and Cross Pseudo Supervision (CPS) [48] employ two segmentation networks with different initializations, and has been extended to medical imaging data and to include shape awareness constraints [49]. In GCT, consistency is enforced between the two networks, while CPS takes a pseudo labeling approach, where the cross-entropy loss between one networks output and the pseudo labels produced by the other network is minimized.

The consistency-based methods share the architecture between student and teacher. Apart from the effects of the averaged model parameters, the augmentations control the outputs' similarity. When a target is predicted, the loss formulations encourage the student to strictly follow the teacher and thus potentially copy uncertainties as a label instead. The proposed loss formulation utilized in ECMT, with the added layers recontextualizing the input space, leads to less confident predictions being less influential and these uncertainties being the product of the combined teacher and student outputs. In ECMT, the proposed loss reduces the influence of less confident predictions and bases these decisions on the combination of the teacher's and the student's state.

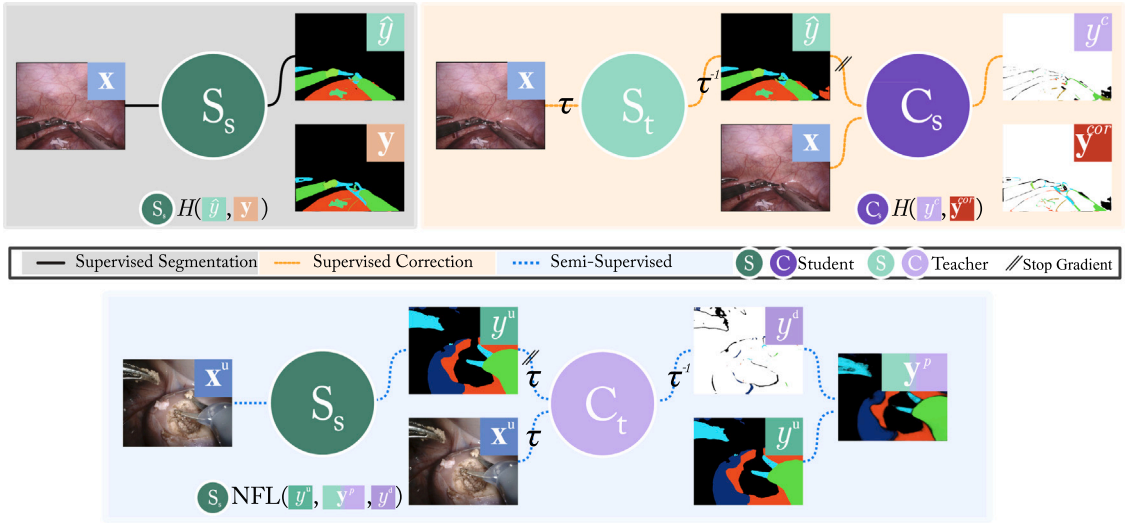

Fig. 4. Overview of Error-Correcting Mean-Teacher. Apart from the supervised segmentation, the method comprises two additional steps (represented by the backdrop color). The Error-Correcting network C is trained with image–segmentation $(\mathbf{x}, \hat{\mathbf{y}})$ pairs to spot inaccuracies in the input. For the semi-supervised step, the segmentation network learns to minimize the Negated Focal Loss (NFL) between the segmentation $\boldsymbol{y}^u$ of an unlabeled image and a pseudo-label. Note: for visual clarity, the outputs of both models are displayed as discrete segmentations.

3. Error-Correcting Mean-Teacher

The following sections will discuss the components of the proposed method. Besides introducing the notation, we highlight the technique of model averaging during training time [10], followed by the supervised and the semi-supervised steps of the framework.

During training, each iteration utilizes the labeled training data and correction maps, as shown in Fig. 3, as well as the pseudo-labels for unlabeled images. A single training iteration with ECMT consists of three parts:

- The supervised segmentation step, where the segmentation model is optimized.
- The supervised error-correcting step, where the correction network learns to find wrong predictions in an estimated segmentation.
- The semi-supervised step, where the correction network is used to improve the accuracy of an unlabeled sample's predicted segmentation, and the student is optimized to match the corrections, proportional to the teacher's confidence.

Detailed interactions, between student and teacher versions of the networks for the three steps are shown in Fig. 4.

In contrast to consistency based models, our use of pseudo-labels prevents the student from mirroring possible high entropy predictions and instead uses low information predictions to decrease the magnitude of the gradient. Thus, we hypothesize that the more pronounced decision boundary of the Cross-Entropy based Negated Focal Loss [11] propagates a more advantageous gradient through the network.

Notice that ECMT does not require weakly-labeled data such as image-level or bounding box annotations. The additional unlabeled data just has to belong to the same domain as the labeled training images.

3.1. Notation

The multi-task architecture consists of layers that are task specific and parts that are shared between the tasks. In the following we will term the layers active during the correction and segmentation tasks as correction network and segmentation network, respectively.

We follow the notation defined in [11], where $\mathbf{x}$ represents an image, and $\mathbf{y} \in \{1, \dots, N\}^{h \times w}$ its corresponding label map. Further, $\hat{\boldsymbol{y}} \in \mathbb{R}^{N \times h \times w}$ is the *probabilistic* output of the segmentation network $S(\mathbf{x})$. The subscript s attached to the segmentation $S(\cdot)$ or correction network $C(\cdot, \cdot)$ denotes the use of the student network, while t symbolizes that the EMA teacher is utilized. The subscript i is used to identify an individual value, in the case of a label map, or vector at one of the $M = h \cdot w$ positions. Lastly, $\mathcal{D}$ and $\mathcal{D}_u$ indicate the labeled and unlabeled training data.

3.2. Exponential moving average

The success of the Mean-Teacher framework [10] for consistency regularization is based on the fact that an ensemble of the model parameters is a more effective teacher than the current parameters given by the state of the model. The weights of the teacher θ^t are the exponential moving average of the weights θ^s of the student model. While θ^s are updated with an optimizer of choice, the teacher's weights θ^t_j at training iteration j are updated as follows:

$$\theta^t_j = \beta \theta^t_{j-1} + (1 - \beta)\theta^s_j, \tag{1}$$

where β controls the temporal length of the ensemble. As the name suggests, the teacher is present in ECMT but it does not produce consistency-targets.

3.3. Supervised step

In ECMT, the primary model that is the basis for the student and teacher models is extended for an additional task (Fig. 1). The model consists of two tasks:

- one retains the functionality to produce a semantic segmentation of the input image into the classes defined in the dataset,
- and the other revises the predicted segmentation, given the features extracted from the input $\boldsymbol{x}$.

This enables the model to take just an image as well as the image–segmentation pair as input. With the pair as input, the model judges how well image and segmentation match on a pixel level and offers corrections for areas where ground-truth and predictions do not agree.

3.3.1. Segmentation task

The network utilized in the segmentation tasks consists of the layers shared in both tasks and a segmentation head (Fig. 1). The model is trained to minimize the cross-entropy H between $\hat{\mathbf{y}} = S_s(\mathbf{x})$ and the labels $\mathbf{y}$:

$$\mathcal{L}_s := \frac{1}{M}\sum_i^M H(\hat{y}_i, \mathrm{y}_i). \tag{2}$$

3.3.2. Error-Correction task

The correction network $C(\cdot,\cdot)$ transforms a given image–segmentation pair into a segmentation map of depth $N+1$, where N is the number of classes in the dataset. Regions labeled as the $(N+1)$th class indicate that the content of the input image was accurately captured by the segmentation.

An overall visualization of this architectural concept is shown in Fig. 1. In both tasks, large parts of the network parameters are shared. Only the final layers are duplicated and not shared between the tasks. For the correction task, the shared layers are extended by additional input and output layers. The input layer operates on the softmax probabilities of the predicted segmentation. Therefore the initial convolution layer accepts an input with N channels, i.e. the depth of output of the segmentation network. As shown in Fig. 1, the input of the final correction layers is the concatenation of features from the initial layer and of the shared core layers.

The final layers utilized for the correction task learn to predict if the image features match the segmentation's characteristics and offer corrections if they do not. Notice that only the initial and final layers of the correction network are *not* trained with information from the unlabeled data directly, and receive gradients only from the labeled data.

Therein lies the advantages of the proposed shared architecture. Only second-order effects would influence a correction network in a setting with two separate architectures for segmentation and correction. If just the segmentation network includes unlabeled data during training, the influence of the unlabeled data on a correction network is limited to the shifting quality of the segmentation, i.e. the correction network's input. With the shared architecture, as the core layers learn more accurate representations by incorporating unlabeled data, they directly influence the output of the correction network that builds on these representations.

Fine-grained correction maps With the Error-Correcting paradigm [11] for semi-supervised segmentation, an additional set of labels are required. To produce the needed targets for the correction task $\mathbf{y}^{cor}$, each prediction $\hat{\mathbf{y}} = \operatorname{argmax} \hat{\boldsymbol{y}}$ is compared with the ground-truth at pixel-level. If the content of ground-truth and segmentation differ, the labels of the former are inherited. All remaining regions are assigned to the $(N+1)$th class:

$$\mathrm{y}_i^{cor} = \begin{cases} N+1 & \text{if } \hat{\mathrm{y}}_i = \mathrm{y}_i \\ \mathrm{y}_i & \text{otherwise.} \end{cases} \tag{3}$$

The involved components as well as the resulting $\mathbf{y}^{cor}$ are shown in Fig. 3.

De-emphasizing class imbalances with a weighted loss Since the correction maps are generated during training and depend on the segmentation's underlying accuracy, the labels constantly shift and evolve during training. The distribution quickly shifts from mostly incorrect to mostly corrected predictions, so the frequency of the individual labels in $\mathbf{y}^{cor}$ are heavily biased towards the added $(N+1)$th class, as one can see in Fig. 3. Always predicting the $(N+1)$th class would result in high accuracies on average, without any learned understanding of the data.

Oversampling can circumvent unbalanced label distributions in a classification setting, and is applicable for segmentation in the same manner. In [11], weighting the loss-contributions of each class differently has been applied with a positive effect. Similarly, we here select a fixed weighting α, so that the contribution to the loss of the regular dataset classes stays untouched. The effects of the additional $(N+1)$th class on the loss, are penalized with a weight of α:

$$\alpha_i = \begin{cases} \alpha & \text{if } \mathrm{y}_i^{cor} = N+1 \\ 1.0 & \text{otherwise.} \end{cases} \tag{4}$$

This incentivizes the correction network to register the underrepresented regions. Setting the weighting for the N classes to 1.0 behaves like the standard cross-entropy loss, while 0.1 shifts the focus to the corrections. An α value of 0.5 can be an appropriate initial configuration for a hyperparameter search.

Correction objective The correction network is optimized with the cross-entropy loss,

$$\mathcal{L}_c := \frac{1}{M}\sum_i^M \alpha_i H(y_i^c, \mathrm{y}_i^{cor}), \tag{5}$$

and is computed between the outputs of the correction network:

$$\boldsymbol{y}^c = C_s(\mathbf{x}, \tau^{-1}(S_t(\tau(\mathbf{x})))), \tag{6}$$

and the generated correction maps $\mathbf{y}^{cor}$. In (6), τ is a function to perturbe the data.

Applying the MT framework to classification instead of segmentation problems, there are no concrete restrictions for the possible perturbations. A class being present in an image is invariant to, for example, rotation. In a segmentation setting, this is only partially true. Whereas with classification only the contents of the image are present in the loss calculation, the pixel-wise nature of segmentation requires the location of the objects as well. Thus, when MT is applied to semantic segmentation, perturbations that modify object locations have to be reverted in order to compare the two samples. In ECMT, the perturbations include noise in the form of dropout [50], as well as horizontal and vertical flipping, depending on the dataset. In this case, the noise-induced by dropout does not significantly alter object positions, and thus, does not have to be reverted. The flipping operations, on the other hand, need to be undone by τ^{-1} in (6). Both the correction and the segmentation networks are trained on labeled data. For the parameter update, the sum of the losses $\mathcal{L}_s$ and $\mathcal{L}_c$ is differentiated with respect to the parameters of the student model.

3.4 Semi-Supervised step

The concept behind ECMT involves the correction network of the teacher model judging the agreement between image and the predicted segmentation. For regions that do not seem to agree, the correction network tries to amend the apparent misclassification. Using the corrections as objective in the semi-supervised step requires additional processing to form pseudo-labels $\mathbf{y}^p$. On an unlabeled image $\mathbf{x}^u$, the continuous *probabilistic* outputs $\boldsymbol{y}^u = S_s(\mathbf{x}^u)$ and $\boldsymbol{y}^d = C_t(\mathbf{x}^u, \boldsymbol{y}^u)$ are both transformed to their discrete label representations $\mathbf{y}^u$ and $\mathbf{y}^d$ with the argmax operator over the depth of the output. All areas predicted as $N+1$ in $\mathbf{y}^d$ are replaced with the corresponding values of $\mathbf{y}^u$, while the remaining corrections are kept:

$$\mathrm{y}_i^p = \begin{cases} \mathrm{y}_i^u & \text{if } \mathrm{y}_i^d = N+1 \\ \mathrm{y}_i^d & \text{otherwise.} \end{cases} \tag{7}$$

3.4.1 Negated Focal Loss

The purpose of the Focal Loss [51] is to reduce the effect easy-to-classify examples have on the overall loss. Hence, the negated probability of the actual class acts as a weight term on the Cross-Entropy. Likewise, the Negated Focal Loss, proposed in [11], is in the shape of a weighted cross-entropy loss. When the Focal Loss is negated, the probabilities directly act as weighting factor:

$$NFL(\cdot,\cdot,y_i^d) = (\max y_i^d)^\gamma H(\cdot,\cdot), \tag{8}$$

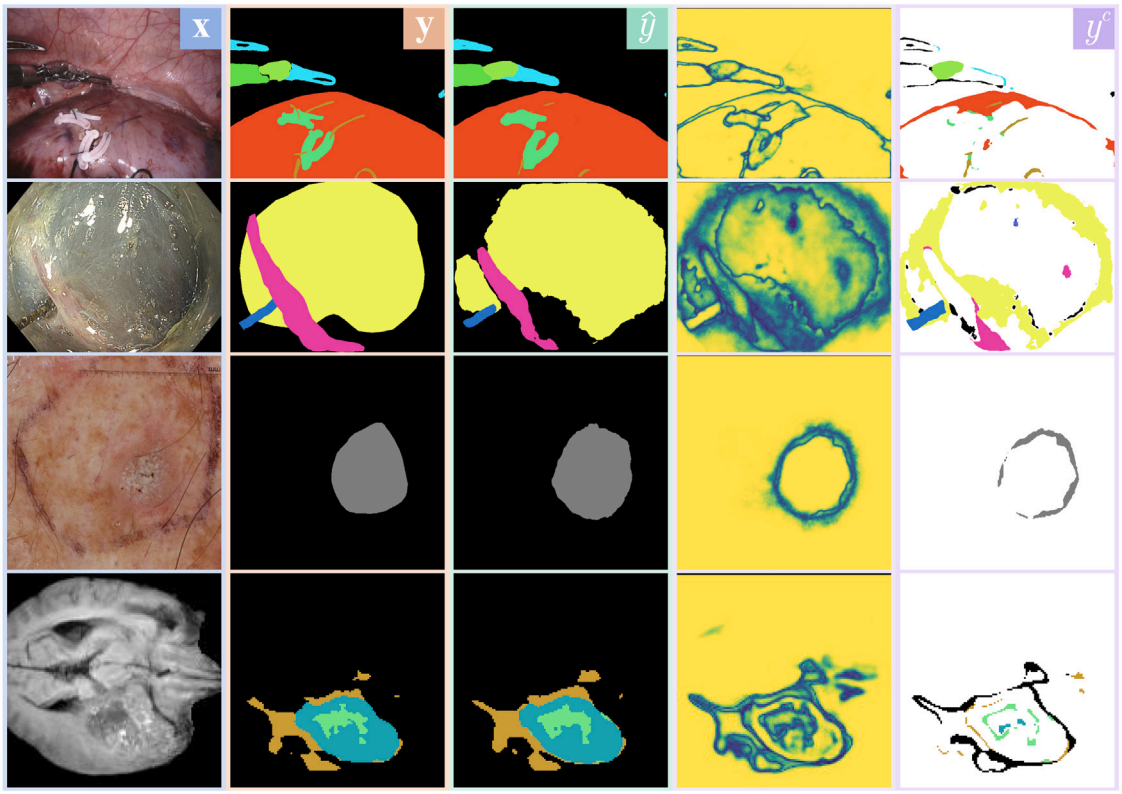

Fig. 5. Overview of some training images. From left to right: Input image, ground truth, segmentation, certainty, correction. Top to bottom: RSS, ESD, ISIC 2017, BraTS. Inspecting both the corrections and their certainty shows that even in some cases with inaccurate corrections, these areas still have a lower *certainty*. The images provide context for the following evaluation. In the first row, ECMT corrects the string class missing from the student segmentation. The model is generally less confident in the second row, which will influence the loss calculation. The correction branch proposed incorrect suggestions in the last row, but with low confidence.

where y_i^d is the $N+1$ dimensional softmax probability vector of the corrector's output at pixel position i, and the focusing parameter γ acts as a smoothing factor. With the Focal Loss, the output of the segmentation network is used as a weighting term. Here, $\boldsymbol{y}^d$, the output of the teacher model's correction task takes on this role. The intention behind weighting a loss calculated with pseudo labels proportional to the *certainty* of the teachers's corrections, is to lessen the contribution of high entropy predictions. Fig. 5 visualizes the *certainty* component used in *NFL*. The semi-supervised loss for a given unlabeled image $\mathbf{x}^u$ is defined as:

$$\mathcal{L}_u := \frac{1}{M}\sum_{i}^{M} NFL(y_i^u, \mathrm{y}_i^p, y_i^d). \tag{9}$$

3.5 Summary of the training process

In summary, one training iteration of ECMT consists of the following parts, shown in Fig. 4 and described in Algorithm 1: calculation of the supervised cross-entropy loss between the predicted segmentation $\hat{\mathbf{y}}$ of an image $\mathbf{x}$ and the ground-truth label $\mathbf{y}$ and differentiating the loss with respect to the parameters of the segmentation head and the shared parameters.

The exponential moving average model predicting a segmentation for the same labeled image followed by the assessment $\boldsymbol{y}^c$ of the image–segmentation pair by the student's correction network. The correction loss is differentiated with respect to the shared parameters and the two sets of additional layers used for the correction task.

On unlabeled data, the student model first segments an image $\mathbf{x}^u$. The teacher corrects the predicted segmentation, and its outputs contribute to the pseudo-label. The semi-supervised loss is differentiated with respect to the same parameters as in the supervised loss.

On none of the occasions, gradients flow through the teacher model. Following are two parameter updates: a single update for the accumulated gradients of the labeled, unlabeled, and correction losses $\mathcal{L}_s$, $\mathcal{L}_c$ and $\mathcal{L}_u$, and the calculation of the moving average.

3.6 Model architecture

The following section describes the architecture of the shared feature extractor and the task specific layers. Our design choices here are just implementation details and no hard requirements that are needed to use ECMT.

3.6.1 Segmentation network

For the 2D datasets, we chose a DeepLabV3+ [20] with a ResNet50 backbone [52,53] as the core model for ECMT. The architecture utilizes dilated instead of strided convolutions in the last residual block, such that the output features of the ResNet backbone are 1/16 of the resolution of the original input. The DeepLabV3+ decoder features Atrous Spatial Pyramid Pooling and applies the softmax function as the last layer.

On 3D data, we follow the architectural patterns described in [54]. The generated nnU-Net consists of 5 downsampling layers while expanding the channel dimension to a maximum of 320. Our implementation applies trilinear interpolation instead of transposed convolutions for the upsampling operations, batch-normalization and ReLU activations.

3.6.2 Correction layers

For 2D data, we chose a plain architecture for the correction networks' initial layers that mirror the initial layers commonly used with ResNets in the segmentation setting. This initial module consists of three 3×3 convolutions, each followed by a BatchNorm layer and

Algorithm 1 Training Pseudo-code. τ is omitted.

```
# .seg, .corr: respective forwad passes
# for the Segmentation and Correction Tasks
for (x, y), x_u in zip(D, D_u):
    # # # supervised segmentation
    y_h = student.seg(x)
    Ls = H(y_h, y).mean()

    # # # supervised correction
    y_h = teacher.seg(x).detach()
    correct = y_h.argmax(1) == y
    y_cor = y.clone()
    y_cor[correct] = N + 1
    y_c = student.corr(x, y_h)
    Lc = (alpha[y_cor]*H(y_c, y_cor)).mean()

    # # # semi-supervised
    y_u = student.seg(x_u)
    y_d = teacher.corr(x_u, y_u).detach()
    y_p = y_d.argmax(1)
    correct = y_p == N + 1
    y_p[correct] = y_u.argmax(1)[correct]
    Lu = NFL(y_u, y_p, y_d).mean()

    # # # parameter update
    (Ls + Lc + Lu).backward()
    optimizer.step()
    ema(student, teacher)
```

ReLU activation. The initial layer halves the spatial dimensions with a stride of two in the convolution and increases the depth to 64 channels. The second convolution preserves depth and spatial dimensions. The last convolution has 256 channels and is followed by an average pooling layer that further reduces the spatial dimension by half. The final block used in the correction task copies the structure of the last layers in the DeepLabV3+ architecture adjusted to accommodate the larger input volume and the additional output class. The interactions between primary and additional layers are shown in Fig. 1.

With nnU-Net [54], the initial layers have the same architecture as the first block of the model. The correction block again repeats the architecture of the last nnU-Net block only increasing the input dimension of the first 3D convolution layer to accommodate the depth of the incoming concatenated features. The correction layers in our nnU-Net operates on the same resolution as the input tensor.

4 Experiments and analysis

To assess the performance of our method, we conduct experiments on several datasets ranging from 2D and 3D challenge datasets and our proprietary dataset. On the medical datasets, we compare ECMT with own experiments applying TCSM [41] and CCT [46], while on the two non-medical datasets, we compare our model to the results posted in [48].

4.1 ISIC

The 2017 ISIC skin lesion segmentation challenge data set [25] includes 2000 annotated dermoscopic training images. We present the result of five independent training runs trained with 50 labeled and 1950 unlabeled images, each evaluated on the 600 images of the 2017 test set. All training and validation images are resized to 512×512 pixels.

4.2 Robotic Scene Segmentation dataset

The Robotic Scene Segmentation (RSS) dataset [7] is made up of 19 individual stereo sequences that, for the 2018 Endoscopic Vision Challenge, were divided into 15 training videos containing 149 labeled left frames and 149 unlabeled right frames and four test sets with 1000 frames in total. The sequences were recorded on a da Vinci X or Xi system from a single porcine training procedure each. In the 2017 challenge [55], the medical device of the da Vinci instrument had to be segmented. In 2018, it was extended with drop-in ultrasound probes, suturing needles, suturing thread, suction-irrigation devices, and surgical clips. Every non-biological object in the image has a corresponding label. The anatomical classes differentiate between a background class, that features all tissue that is not one of the following: kidney parenchyma, the kidney fascia, and perinephric fat termed 'covered kidney', and small intestine. In total, the 2018 dataset spans 12 object classes, which are not present in every sequence.

To increase the size of the dataset, we include all of the images from the 2017 Challenge as unlabeled data in our experiments, resulting in a total of 5335 images. The images are resized such that the shorter side has a length of 512 pixels and then randomly cropped to 624×496.

We evaluate our algorithm with five-fold cross-validation on a sequence level. Each fold consists of 1/5 of the 2018 sequences. We constrain the models to 10% of the labeled training set and include the remaining 90% as unlabeled samples. The amount of data in the respective validation fold remains unconstrained.

4.3 Endoscopic Submucosal Dissection

Endoscopic Submucosal Dissection (ESD) is an endoscopic technique for the en bloc resection of gastrointestinal lesions [56]. First, a submucosal fluid cushion is injected underneath the lesion. Then, an incision of the mucosa at the outer borders of the lesion is performed using a specialized ESD knife. With the opened submucosal space, further dissection of the submucosa is carried out. Finally, the lesion's complete en bloc resection is achieved when the circumferential mucosal incision is completed and the lesion is detached from the underlying muscle layer.

To detect the individual components in the ESD, we created a partially labeled dataset from 12 full-length procedures recorded at the University Hospital Augsburg. The partially labeled dataset consists of five individual classes, i.e. submucosa, blood vessel, knife, instrument shaft, and background. From the 12 videos, a total of 4120 frames were captured, of which 401 are labeled. The images, with an original resolution of 2700×2160 pixels, are resized and cropped equivalent to the process described for RSS. The evaluation consists of five-fold cross-validation on a sequence level similar to RSS. Contrary to the other datasets, we do not constrain the amount of labeled data as the size of dataset is already fairly limited, considering its diversity. Thus, we train the semi-supervised methods with all the images of the sequences contained in the current fold, and the supervised baseline with the labeled subset of the folds sequences.

4.4 BraTS

Apart from the 2D data, we also include a 3D medical imaging dataset, the brain tumor segmentation (BraTS) challenge 2020 [57–59] in our evaluation. The 2020 dataset consists of 369 training and 125 validation cases. For each scan, four modalities are available i.e., T1, T1c, T2, and Flair.

During training, we randomly crop the stacked modality slices resulting in an input tensor to a shape of $4 \times 128 \times 128 \times 128$ per sample. Since the labels for the challenge validation set are not publicly available, we incorporate the validation data as unlabeled samples. We evaluate the models with five-fold cross-validation. For each run, the labeled training data is limited to 20% of each fold. The remaining training samples of the fold and all of the slices of the challenge validation set are used as unlabeled samples.

Table 1
Results on the ISIC 2017 test set. Only TCSM and ECMT slightly improve over the supervised baseline with 2.5% of the labels.

Run	Sup 2.5%	CCT [46]	TCSM [41]	ECMT	Sup 100%
#1	75.14	74.72	73.58	**76.61**	82.28
#2	72.22	71.64	74.56	**75.15**	82.33
#3	**77.34**	75.04	77.10	77.19	81.97
#4	76.26	76.55	**76.85**	75.96	82.79
#5	78.32	78.35	**78.57**	77.33	82.34
Mean	75.86	75.26	76.13	**76.45**	82.34

4.5 Setup

For all experiments, the segmentation network is trained with Stochastic Gradient Descent [60,61] with a learning rate of 0.01, momentum 0.9, weight decay of $1e-4$, and polynomial learning rate decay $lr = lr_{initial} \cdot (1 - \frac{\text{iter}}{\text{maxiter}})^{0.9}$ applied after each iteration.

In the segmentation and semi-supervised steps, we apply label-smoothing of 0.1 to the labels and generated pseudo-labels. The hyperparameter α in the correction loss is set to 0.2 on the 2D and 0.5 on the 3D datasets.

We chose a batch size of four for both the labeled and unlabeled data and trained for 20 400 iterations. The iteration count derives from redefining the length of one epoch as 256 iterations on all datasets, independent of the actual amount of labeled data, and then training the algorithms for 80 epochs.

The weights of the DeepLab are initialized to the publicly available pre-trained COCO [4] model contained in the PyTorch [62] repository. The remaining layers are initialized as described in [52]. The nnU-Net is trained from scratch, i.e. without pretrained weights.

4.5.1 Augmentations

Apart from dataset-specific aspect ratios for the random cropping, all training images are flipped randomly along both axes. Aside from BraTS, color jittering and gaussian noise, implemented in the albumentations [63] library, are randomly added to all images during training. Each RGB image is normalized with the mean and standard-deviation intended for the pretrained weights.

The inputs and outputs of the teacher's correction task are flipped horizontally and vertically, as represented by the transformation τ. In combination with noise, jittering, and dropout layers in the network, this step further ensures that student- and teacher-model receive a different view of the same data point.

4.5.2 Evaluation approach

The quality of the models is assessed with the commonly used mean Intersection-over-Union (mIoU),

$$\frac{1}{N}\sum_{n}^{N}\frac{TP_n}{TP_n + FP_n + FN_n}, \tag{10}$$

where TP_n, FP_n and FN_n are the true positives, false positives and false negatives concerning label $n \in N$. For the Robotic Scene Segmentation results, we specifically follow the protocol outlined in [7].

With the definition of one epoch as 256 iterations, we validate all models on the last ten training epochs and present the averages over the ten runs. The main results show just the performance of the student's segmentation performance. This measure is intended to further reduce the variance in our evaluation. No early stopping or related techniques are applied.

4.6 Results

In the following sections, we compare ECMT with limited labeled data with our own implementation of Transformation Consistent Self-Ensembling (TCSM) [41], the public implementation of Cross Consistency Training (CCT) [46] as well as the purely supervised model.

Table 2
Overview of the performances on the Robotic Scene Segmentation dataset. With 10% of the labeled data, all semi-supervised methods improve over the baseline. CCT beats TCSM in fold 2,3 and 4. ECMT is a substantial improvement over both CCT and TCSM on all but one fold. None of the methods surpasses the supervised model with 100% of the labeled data.

Fold	Sup 10%	CCT [46]	TCSM [41]	ECMT	Sup 100%
#1	47.95	50.17	50.29	**54.88**	55.05
#2	52.97	55.41	53.27	**63.18**	64.16
#3	54.44	**58.78**	58.44	58.53	65.13
#4	55.09	57.90	57.63	**61.59**	65.39
#5	62.91	65.25	65.78	**66.56**	73.00
Mean	54.67	57.50	57.08	**60.95**	64.55

Table 3
Results on our ESD dataset. All semi-supervised methods improve upon the supervised baseline. TCSM and ECMT are close in performance and benefit the most from the additional unlabeled samples.

Fold	CCT [46]	TCSM [41]	ECMT	Sup 100%
#1	59.94	62.43	**63.63**	58.64
#2	59.17	61.75	**62.20**	57.57
#3	56.45	**58.88**	58.62	56.46
#4	43.84	43.06	**44.88**	43.50
#5	68.82	72.08	**72.19**	67.05
Mean	57.65	59.64	**60.30**	56.64

4.6.1 ISIC

On the dermoscopy dataset [25], all methods achieve similar values for the mIoU measure. Table 1 highlights that CCT [46] slightly deteriorates in comparison to the baseline, while TCSM [41] and ECMT improve upon it. Li et al. [41] included the results on the test set, trained with 300 as labeled images and the remainder as unlabeled data, achieving a mIoU of 78.1. We replicate their experiment with our TCSM implementation and increased the mIoU to 79.6 averaged over five runs. With the same amount of labeled data, ECMT reaches a mIoU of 80.63 on the test set.

4.6.2 Robotic Scene segmentation

On the Robotic Scene Segmentation [7,55] dataset, all semi-supervised models surpass the supervised baseline, presented in Table 2. TCSM and CCT achieved similar results with an averaged mIoU of 57.08 and 57.50, respectively. ECMT surpasses both methods on all but one validation fold, averaging a mIoU of 60.95 with only 10% of the labels used. A fully supervised model obtains a mIoU of 64.55.

4.6.3 ESD

On our ESD dataset, ECMT again provides the best results (Table 3). Similar to the RSS, there is a large variance between the folds since the validation sequences do not overlap. TCSM achieves the best mIoU on fold 3, and the differences in the average performance are not as pronounced as in the Robotic Scene Segmentation dataset, with a mIoU of 59.64 to ECMT's 60.30.

4.6.4 BraTS

On the BraTS [57–59] dataset, the common evaluation metric is the dice coefficient. Instead of reporting the average performance per fold for each training class, we present the results similar to [54]. The training classes of edema, non-enhancing tumor and enhancing tumor are recombined to the *whole*, *core* and *enhancing* classes, respectively.

The dice scores in Table 4 present the averages for the aggregated classes over five folds. On the BraTS dataset, ECMT with 20% of the labeled data achieved a substantial lead over the other semi-supervised methods and approached the fully supervised model with a mean dice-score of 86.22 to 87.38. Increasing the labeled data to 100% and only

Table 4
Supervised and semi-supervised results on the BraTS 2020 dataset. Whole, Enhancing, and Core values are the average dice scores for the respective class over the five validation folds. ECMT considerably improves upon the baseline and competing methods.

Class	Sup 20%	CCT [46]	TCSM [41]	ECMT	Sup 100%
Whole	89.81	88.63	90.19	**91.24**	90.99
Core	80.34	79.32	81.10	**83.63**	85.95
Enh.	82.45	81.16	82.18	**83.79**	85.20
Mean	84.20	83.03	84.49	**86.22**	87.38

Table 5
Mean IoU of the 5 × 2 cv runs.

Dataset	TCSM [41]	ECMT	
RSS	52.57 ± 3.89	**55.23** ± 4.48	$p < 0.01$
ESD	57.99 ± 4.59	**59.34** ± 4.92	$p < 0.05$
ISIC	85.13 ± 1.30	**85.33** ± 1.94	$p > 0.05$

utilizing the cases from the BraTS validation set as unlabeled data, leads to a mean dice-score of 87.70. Isensee et al. [54] report a mean dice score of 87.07 on their cross-validation experiments.

4.7 Statistical significance

In most experiments, ranking the algorithms by mIoU scores ECMT occupies the top position, followed by TCSM and CCT. But this is not consistent over every validation fold, and if ECMT is in the lead, the extent varies from dataset to dataset. To determine whether our method leads to statistically significant improvements, we conduct the 5 × 2 cv test [64], between ECMT and the closest competitor, TCSM (see Table 5). The training data is partitioned into two folds, with the alteration that the ratios of labeled and unlabeled data from the main results are kept. In numbers, on ISIC, each fold contains 1000 images, where 25 are used as labeled, and the remaining 975 images are used as unlabeled training images. The two-fold cross-validation is repeated five times. On RSS and ESD, ECMT achieved a statistical significant improvement with $P < 0.01$ and $P < 0.05$ respectively. On BraTS, TCSM failed to converge with 36 labeled slides. Only on the ISIC dataset, both approaches perform equally well, with no statistically significant differences.

4.8 Cityscapes and Pascal VOC

Although the focus of this paper is medical imaging, our method is not strictly tailored to this type of data and can be applied to other semantic segmentation datasets — like Cityscapes [65] and Pascal VOC 2012 [66]. On both datasets, we conduct experiments with labeled ratios of 6.25%, 12.5%, 25%, and 50%. On Cityscapes, we define the length of an epoch to 495 iterations and train for 90 epochs. On Pascal VOC with the sbd labels, our training lasts for 90 epochs, where one epoch is defined as 440 iterations. The crop size for the Cityscapes training images is 800 × 800 pixels, and 512 × 512 pixels on Pascal VOC. The models are trained with batch size 8 on Cityscapes and 16 on Pascal VOC. The chosen number of iterations results in a similar length of training to CPS [48] and we use the same scaling augmentations.

Table 6 presents a comparison between CCT, CPS and ECMT on the validation sets. Both CPS and ECMT lead to a sizable improvement over CCT. CPS and ECMT perform similarly, with ECMT leading in settings with the fewest labeled training data.

4.9 Ablation and discussion

We conduct a series of self-contained experiments to show the influence individual components of our proposed method have.

Table 6
MIoU on the Cityscapes and PASCAL VOC 2012 validation sets. The results for methods marked † with are taken from [48]. For ECMT, we report the average mIoU of the last ten epochs.

Dataset	Method	6.25%	12.5%	25%	50%
Cityscapes	**CCT** [46]†	66.35	72.46	75.68	76.78
	CPS [48]†	69.79	74.39	**76.85**	**78.64**
	ECMT	**71.19**	**75.18**	76.73	77.84
Pascal	**CCT** [46]†	65.22	70.87	73.43	74.49
	CPS [48]†	68.21	73.20	74.24	**75.91**
	ECMT	**71.48**	**73.37**	**74.95**	75.53

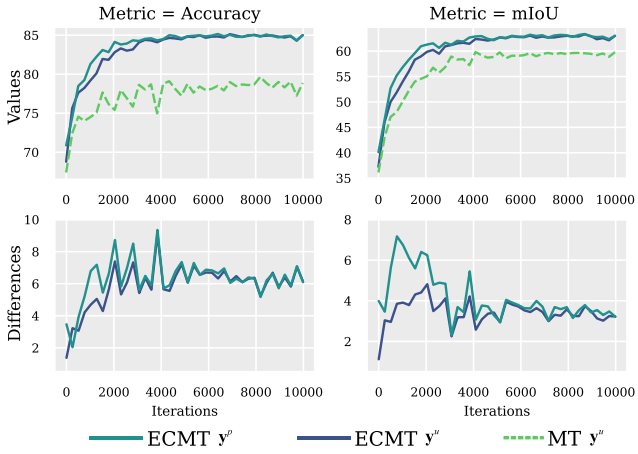

Fig. 6. Comparing the quality of the pseudo labels between ECMT and a standard Mean-Teacher model. The first row shows the absolute values and the second row the difference between ECMT's values and MT. Especially in the early iterations the pseudo-labels $\mathbf{y}^p$ in ECMT provide better approximations for the true labels than just the segmentation in ECMT and MT. In later stages, the output of the segmentation task $\mathbf{y}^u$ approaches the quality of the combination of segmentation and correction $\mathbf{y}^p$.

4.9.1 Relation between correction and truth

We perform an additional experiment on the Robotic Scene Segmentation dataset to gain a better understanding why the correction objective outperforms the consistency targets. We partition the data to use 5 of the 15 sequences in the semi-supervised step and evaluate how the accuracy of the predictions develops throughout 10 000 training iterations.

We train both ECMT and a simple Mean-Teacher consistency regularization approach with this setup. With ECMT, we calculate pixel accuracy and mIoU for the output of the segmentation network $S_t(\mathbf{x}^u)$ and the combination of the two tasks, the pseudo labels $\mathbf{y}^p$ (Fig. 6).

The differences in mIoU and accuracies imply that the outputs in ECMT provide considerably better approximations than in a consistency regularization setting. After ∼ 1500 iterations, the pseudo-labels predicted with ECMT are closer to the ground-truth than the best result of the consistency regularization model. More accurate representations produced early during training lead to better pseudo-labels, which improves the representations.

In the early stages of training, pseudo labels are more accurate than just the teacher's segmentation. The correction network operates on features from the input image and its segmentation. It processes more information than during the segmentation task, and the experiment suggests that this can lead to more accurate predictions. The gap vanishes throughout training as the segmentation network converges.

4.9.2 Increasing the amount of labeled data

For the main experiments (Section 4.6) on the medical datasets, the amount of labeled data is limited to a fraction of the entire set. It is, however, also valuable to study the effects that increasing amounts of labeled data provide for ECMT. Fig. 7 displays a progressive performance increase when more labels are made available. With 50%

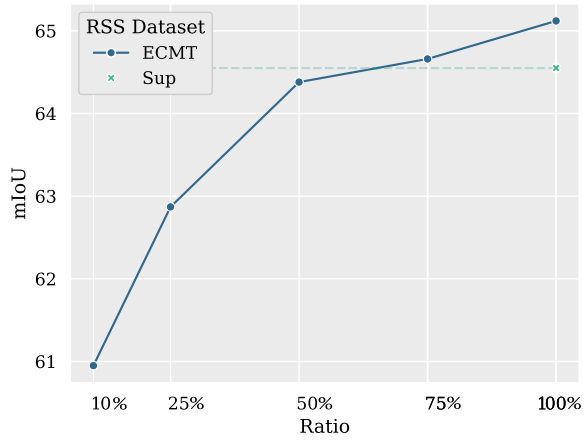

Fig. 7. Evolution of average mIoU over the five folds with increasing amounts of labeled data on the Robotic Scene Segmentation Dataset. Our algorithm profits from more labeled data, and also surpasses the fully labeled supervised baseline.

Table 7
Ablation of weighting scheme used to train the correction task. The method is fairly robust in the choice of α and only falls below a competing approach in a single case. Decreasing α, and thus, encouraging the correction network to focus on mistakes has significant positive effects on the evaluation metric.

	α			
Dataset	1.0	0.5	0.2	0.1
ISIC	76.38	76.71	76.45	**77.79**
RSS	58.19	60.15	60.95	**62.07**
ESD	59.60	59.49	60.30	**60.51**
BraTS	83.59	**86.22**	84.91	84.22

Table 8
Comparison of TCSM with label smoothing (LS) and the negated focal loss with ECMT on the ESD and Robotic Scene Segmentation datasets.

	RSS	ESD
TCSM	57.08	59.64
TCSM-LS	57.15	58.64
TCSM-NFL	56.35	59.81
ECMT	**60.95**	**60.30**

of the labeled data, ECMT borders on the fully supervised model and continues to increase to an average mIoU of 65.12 over the five folds. In the last case, the unlabeled data only consists of challenge images from 2017.

4.9.3 Measuring the similarities between the learned representations

Evaluating the performance on the respective validation sets, the results in the prior section indicate that ECMT was more effective at learning from unlabeled data. But just comparing the segmentation quality only gives limited insight into the learning dynamics of the models. A different approach is the investigation of the representations themselves, learned by the neural networks. Kornblith et al. [67] have proposed Centered Kernel Alignment (CKA) as a similarity measure for representations. With a linear kernel, the CKA between the centered matrices $X, Y \in \mathbb{R}^{n \times p}$ of p activations for n samples is calculated as following:

$$\mathrm{CKA}(X, Y) = \frac{\|Y^T X\|_F^2}{\|X^T X\|_F \|Y^T Y\|_F}. \tag{11}$$

CKA can reliably discern a correspondence between the same layers in networks trained with different initializations.

We use linear CKA to study the similarities between the representations learned after the ASPP module on the 2D datasets. We compare the representations of the bootstrapped validation data learned by the semi-supervised models with their supervised counterparts trained with partially and fully labeled data.

Of the three 2D datasets, only on ISIC the unlabeled set is strictly a subset of the whole labeled dataset. On RSS and ESD the data, the semi-supervised models have access to, is a superset of the labeled data.

With these distinct settings, we form two assumptions.

- An effective semi-supervised method should produce similar representations to a supervised model when trained on the same data.
- If the semi-supervised model has access to additional data that exhibits some domain shift, we would expect to see less similar representations if the added data influences the model.

Fig. 8 shows the similarities for the three datasets. On ISIC both TCSM and ECMT are generally similar to their supervised counterparts and between each other, suggesting that the inclusion of the correction task does not lead to notable representational shifts.

On ESD, the difference is the largest, with ECMT seemingly having learned far different representations. ECMT did achieve the best mIoU on the ESD dataset, ruling out degrading model performance as the cause of the discrepancy. On RSS, the differences are less pronounced, but ECMT's representational shift is still visible. Following the stated assumptions, the representations being more similar on ISIC and less so on the other two datasets could indicate ECMT's effectiveness at learning from unlabeled data.

4.9.4 Effect of α in the correction loss

The prior experiment and Fig. 6 show that, on the Robotic Scene Segmentation dataset, the accuracies of the correction task generally hover around 88%. Thus, classifying the whole input as *correct* by the correction network could reach an error rate of $\sim$ 10%, without any learning taking place. But in turn, a corrector that accepts every input equally would not produce pseudo-labels and thus degrade the quality of the segmentation. The prevention of this phenomenon is the intended objective of the weighting scheme, discussed in Section 3.3.2 and Eq. (4). This hyperparameter choice impacts the overall performance of ECMT. Accordingly, we report the results on all datasets with $\alpha \in \{1.0, 0.5, 0.2, 0.1\}$ in this section. On the 2D datasets, reducing α steadily raises the mIoU, apart from a single outlier on the ESD dataset. However, on the 3D BraTS dataset, ECMT appears sensitive to the parameter choice, with both low and high values of α decaying the average dice score. Setting α to 0.5 leads to the best results in this case. Although this attests that ECMT requires some hyperparameter tuning when applied to new problems, a general default setting of 0.5 can be a very competitive initial value, with further gains being possible (see Table 7).

4.9.5 Label smoothing and Negated Focal Loss

The Negated Focal Loss incorporates the *certainty* in the prediction to reduce the influence of individual high entropy outputs. Apart from the correction task, the *NFL* loss (Eq. (8)) is the second distinction between ECMT and the consistency-based approaches featured in this work. To highlight that ECMT's improvements are not exclusively attained by a differing loss, we incorporate Cross-Entropy with label-smoothing (LS) and the *NFL* with TCSM and repeat the experiments on the Robotic Scene Segmentation and ESD datasets. Table 8 presents the average mIoU over five folds. Depending on the dataset, either label-smoothing or *NFL* can improve over the baseline results. However, in all cases, these improvements are not consistent and trail ECMT. These results indicate that just including the certainty is not the source of ECMT's performance advantage. The addition of the correction task forces the network to learn more accurate representations, and is a critical component and reinforces the results from Section 4.9.1. The proposed correction mechanism of ECMT is essential for consistent performance in the semi-supervised settings.

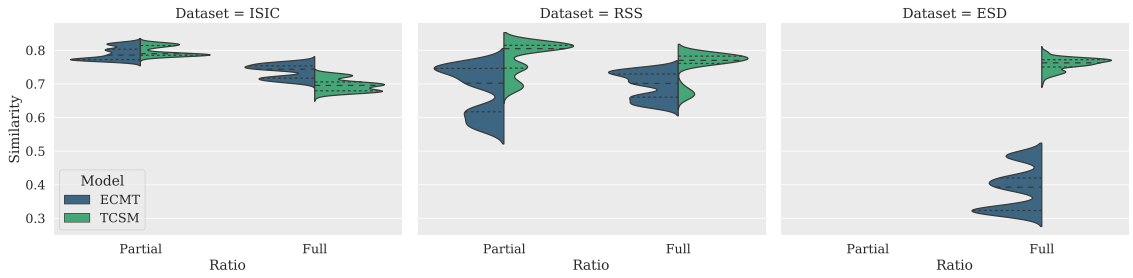

Fig. 8. Comparison of the similarities between the representations of the semi-supervised and supervised models calculated with Centered Kernel Alignment [67]. The representational shift on RSS and ESD, where the semi-supervised models have access to images unavailable to the supervised models, implies that ECMT incorporated information from the unlabeled samples more effectively.

5 Conclusion

Error-Correcting Mean-Teacher offers an alternative for semi-supervised segmentation utilizing an EMA teacher while combining it with the Error-Correction paradigm [11]. It can easily be added to existing supervised models if additional unlabeled data is present. The pseudo-labels produced by fusing the outputs of the correction network with the preceding segmentation have shown to be suitable approximations for the ground-truth labels. Since the correction network operates on the features shared with the primary segmentation network, it can profit from an improving segmentation model.

A possible direction for future work is architectural advancements of the layers merging image and segmentation features. The inductive biases of convolutions lend themselves advantageously for imaging tasks, but the correction problem could be interpreted as a sequence problem with long-range to global dependencies. A larger receptive field or attention could be better suited to correlate the incoming features as, especially with attention, the strict neighborhood constraints are broken. Further, the choice of concatenating the incoming features is an implementation detail and should be evaluated in future work.

We have shown that our approach consistently outperforms a supervised baseline as well as competing semi-supervised methods. The datasets vary between 2D and 3D data, from binary segmentation to more complex problems with 12 individual classes. We have shown that the method is robust to variations in the data domain by including the images from the previous challenges in the experiments in the case of Robotic Scene Segmentation. By experimenting with two widely different architectures, namely nnU-Net and DeepLabV3+, we have shown that ECMT is adaptable and should apply to newer model architectures and paradigms.

Declaration of competing interest

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

Acknowledgments

The authors thank Prof. Dr. Helmut Messmann and Dr. Alanna Ebigbo from the University Hospital in Augsburg, Germany, for curating and providing the ESD dataset. The first author is funded by the Bavarian Research Institute for Digital Transformation (bidt). Open Access funded by Ostbayerische Technische Hochschule Regensburg.

References

[1] A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks, in: F. Pereira, C.J.C. Burges, L. Bottou, K.Q. Weinberger (Eds.), Advances in Neural Information Processing Systems 25, Curran Associates, Inc., 2012, pp. 1097–1105.

[2] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, ImageNet: A large-scale hierarchical image database, in: CVPR09, 2009.

[3] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, B. Schiele, The cityscapes dataset for semantic urban scene understanding, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 3213–3223.

[4] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C.L. Zitnick, Microsoft COCO: Common objects in context, in: D. Fleet, T. Pajdla, B. Schiele, T. Tuytelaars (Eds.), Computer Vision – ECCV 2014, Springer International Publishing, Cham, 2014, pp. 740–755.

[5] N. Tajbakhsh, L. Jeyaseelan, Q. Li, J.N. Chiang, Z. Wu, X. Ding, Embracing imperfect datasets: A review of deep learning solutions for medical image segmentation, Med. Image Anal. 63 (2020) 101693, http://dx.doi.org/10.1016/j.media.2020.101693.

[6] O. Ronneberger, P. Fischer, T. Brox, U-Net: Convolutional networks for biomedical image segmentation, in: N. Navab, J. Hornegger, W.M. Wells, A.F. Frangi (Eds.), Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, Springer International Publishing, Cham, 2015, pp. 234–241.

[7] M. Allan, S. Kondo, S. Bodenstedt, S. Leger, R. Kadkhodamohammadi, I. Luengo, F. Fuentes, E. Flouty, A. Mohammed, M. Pedersen, et al., 2018 Robotic scene segmentation challenge, 2020, arXiv preprint arXiv:2001.11190.

[8] A. Ebigbo, R. Mendel, M.W. Scheppach, A. Probst, N. Shahidi, F. Prinz, C. Fleischmann, C. Römmele, S.K. Gölder, G. Braun, D. Rauber, T. Rückert, L.A. de Souza Jr., J.P. Papa, M. Byrne, C. Palm, H. Messmann, Vessel and tissue recognition during third-space endoscopy using a deep learning algorithm, Gut 71 (2022) 2388–2390, http://dx.doi.org/10.1136/gutjnl-2021-326470.

[9] A. Ebigbo, R. Mendel, A. Probst, M. Meinikheim, M.F. Byrne, H. Messmann, C. Palm, Multimodal imaging for detection and segmentation of Barrett's esophagus-related neoplasia using artificial intelligence, Endoscopy 54 (10) (2022) http://dx.doi.org/10.1055/a-1704-7885.

[10] A. Tarvainen, H. Valpola, Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results, in: I. Guyon, U.V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Eds.), Advances in Neural Information Processing Systems 30, Curran Associates, Inc., 2017, pp. 1195–1204.

[11] R. Mendel, L.A. de Souza, D. Rauber, J.P. Papa, C. Palm, Semi-supervised segmentation based on error-correcting supervision, in: A. Vedaldi, H. Bischof, T. Brox, J.-M. Frahm (Eds.), Computer Vision – ECCV 2020, Springer International Publishing, Cham, 2020, pp. 141–157.

[12] K. He, X. Zhang, S. Ren, J. Sun, Spatial pyramid pooling in deep convolutional networks for visual recognition, IEEE Trans. Pattern Anal. Mach. Intell. 37 (9) (2015) 1904–1916.

[13] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A.L. Yuille, DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs, IEEE Trans. Pattern Anal. Mach. Intell. 40 (4) (2018) 834–848.

[14] H. Zhao, J. Shi, X. Qi, X. Wang, J. Jia, Pyramid scene parsing network, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[15] V. Badrinarayanan, A. Kendall, R. Cipolla, SegNet: A deep convolutional encoder-decoder architecture for image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 39 (12) (2017) 2481–2495.

[16] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, in: I. Guyon, U.V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Eds.), Advances in Neural Information Processing Systems, Vol. 30, Curran Associates, Inc., 2017, URL https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[17] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, N. Houlsby, An image is worth 16x16 words: Transformers for image recognition at scale, 2020, CoRR abs/2010.11929, arXiv:2010.11929, URL https://arxiv.org/abs/2010.11929.

[18] E. Shelhamer, J. Long, T. Darrell, Fully convolutional networks for semantic segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 39 (4) (2017) 640–651, arXiv:1411.4038.

[19] L.-C. Chen, G. Papandreou, F. Schroff, H. Adam, Rethinking atrous convolution for semantic image segmentation, 2017, arXiv:1706.05587.

[20] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, H. Adam, Encoder-decoder with atrous separable convolution for semantic image segmentation, in: The European Conference on Computer Vision (ECCV), 2018.

[21] F. Yu, V. Koltun, Multi-scale context aggregation by dilated convolutions, 2015, arXiv:1511.07122.

[22] Y. Zhang, R. Higashita, H. Fu, Y. Xu, Y. Zhang, H. Liu, J. Zhang, J. Liu, A multi-branch hybrid transformer network for corneal endothelial cell segmentation, in: M. de Bruijne, P.C. Cattin, S. Cotin, N. Padoy, S. Speidel, Y. Zheng, C. Essert (Eds.), Medical Image Computing and Computer Assisted Intervention – MICCAI 2021, Springer International Publishing, Cham, 2021, pp. 99–108.

[23] J. Wang, L. Wei, L. Wang, Q. Zhou, L. Zhu, J. Qin, Boundary-aware transformers for skin lesion segmentation, in: M. de Bruijne, P.C. Cattin, S. Cotin, N. Padoy, S. Speidel, Y. Zheng, C. Essert (Eds.), Medical Image Computing and Computer Assisted Intervention – MICCAI 2021, Springer International Publishing, Cham, 2021, pp. 206–216.

[24] H. Peiris, M. Hayat, Z. Chen, G. Egan, M. Harandi, A robust volumetric transformer for accurate 3D tumor segmentation, in: L. Wang, Q. Dou, P.T. Fletcher, S. Speidel, S. Li (Eds.), Medical Image Computing and Computer Assisted Intervention – MICCAI 2022, Springer Nature Switzerland, Cham, 2022, pp. 162–172.

[25] N.C.F. Codella, D. Gutman, M.E. Celebi, B. Helba, M.A. Marchetti, S.W. Dusza, A. Kalloo, K. Liopyris, N. Mishra, H. Kittler, A. Halpern, Skin lesion analysis toward melanoma detection: A challenge at the 2017 International symposium on biomedical imaging (ISBI), hosted by the international skin imaging collaboration (ISIC), in: 2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018), 2018, pp. 168–172, http://dx.doi.org/10.1109/ISBI.2018.8363547.

[26] F. Isensee, P.F. Jaeger, S.A.A. Kohl, J. Petersen, K.H. Maier-Hein, nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation, Nature Methods 18 (2) (2020) 203–211, http://dx.doi.org/10.1038/s41592-020-01008-z.

[27] W. Wang, A. Tu, F.B. and, Improved minimum spanning tree based image segmentation with guided matting, KSII Trans. Internet Inf. Syst. 16 (1) (2022) 211–230, http://dx.doi.org/10.3837/tiis.2022.01.012.

[28] H. Su, D. Zhao, H. Elmannai, A.A. Heidari, S. Bourouis, Z. Wu, Z. Cai, W. Gui, M. Chen, Multilevel threshold image segmentation for COVID-19 chest radiography: A framework using horizontal and vertical multiverse optimization, Comput. Biol. Med. 146 (2022) 105618, http://dx.doi.org/10.1016/j.compbiomed.2022.105618, URL https://www.sciencedirect.com/science/article/pii/S0010482522004103.

[29] A. Qi, D. Zhao, F. Yu, A.A. Heidari, Z. Wu, Z. Cai, F. Alenezi, R.F. Mansour, H. Chen, M. Chen, Directional mutation and crossover boosted ant colony optimization with application to COVID-19 X-ray image segmentation, Comput. Biol. Med. 148 (2022) 105810, http://dx.doi.org/10.1016/j.compbiomed.2022.105810, URL https://www.sciencedirect.com/science/article/pii/S0010482522005716.

[30] K. Hu, L. Zhao, S. Feng, S. Zhang, Q. Zhou, X. Gao, Y. Guo, Colorectal polyp region extraction using saliency detection network with neutrosophic enhancement, Comput. Biol. Med. 147 (2022) 105760, http://dx.doi.org/10.1016/j.compbiomed.2022.105760, URL https://www.sciencedirect.com/science/article/pii/S0010482522005340.

[31] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, in: Z. Ghahramani, M. Welling, C. Cortes, N.D. Lawrence, K.Q. Weinberger (Eds.), Advances in Neural Information Processing Systems 27, Curran Associates, Inc., 2014, pp. 2672–2680.

[32] N. Souly, C. Spampinato, M. Shah, Semi supervised semantic segmentation using generative adversarial network, in: The IEEE International Conference on Computer Vision (ICCV), 2017.

[33] P. Luc, C. Couprie, S. Chintala, J. Verbeek, Semantic segmentation using adversarial networks, 2016, arXiv:1611.08408.

[34] W.-C. Hung, Y.-H. Tsai, Y.-T. Liou, Y.-Y. Lin, M.-H. Yang, Adversarial learning for semi-supervised semantic segmentation, in: Proceedings of the British Machine Vision Conference (BMVC), 2018.

[35] D. Nie, Y. Gao, L. Wang, D. Shen, ASDNet: Attention based semi-supervised deep networks for medical image segmentation, in: A.F. Frangi, J.A. Schnabel, C. Davatzikos, C. Alberola-López, G. Fichtinger (Eds.), Medical Image Computing and Computer Assisted Intervention – MICCAI 2018, Springer International Publishing, Cham, 2018, pp. 370–378.

[36] L. Wang, D. Guo, G. Wang, S. Zhang, Annotation-efficient learning for medical image segmentation based on Noisy Pseudo Labels and adversarial learning, IEEE Trans. Med. Imaging (2020) 1, http://dx.doi.org/10.1109/TMI.2020.3047807.

[37] S. Laine, T. Aila, Temporal ensembling for semi-supervised learning, 2016, arXiv preprint arXiv:1610.02242.

[38] X. Shi, H. Su, F. Xing, Y. Liang, G. Qu, L. Yang, Graph temporal ensembling based semi-supervised convolutional neural network with noisy labels for histopathology image analysis, Med. Image Anal. 60 (2020) 101624, http://dx.doi.org/10.1016/j.media.2019.101624, URL https://www.sciencedirect.com/science/article/pii/S1361841519301604.

[39] W. Cui, Y. Liu, Y. Li, M. Guo, Y. Li, X. Li, T. Wang, X. Zeng, C. Ye, Semi-supervised brain lesion segmentation with an adapted mean teacher model, in: A.C.S. Chung, J.C. Gee, P.A. Yushkevich, S. Bao (Eds.), Information Processing in Medical Imaging, Springer International Publishing, Cham, 2019, pp. 554–565.

[40] H. Su, X. Shi, J. Cai, L. Yang, Local and global consistency regularized mean teacher for semi-supervised nuclei classification, in: D. Shen, T. Liu, T.M. Peters, L.H. Staib, C. Essert, S. Zhou, P.-T. Yap, A. Khan (Eds.), Medical Image Computing and Computer Assisted Intervention – MICCAI 2019, Springer International Publishing, Cham, 2019, pp. 559–567.

[41] X. Li, L. Yu, H. Chen, C.-W. Fu, L. Xing, P.-A. Heng, Transformation-consistent self-ensembling model for semisupervised medical image segmentation, IEEE Trans. Neural Netw. Learn. Syst. 32 (2) (2021) 523–534, http://dx.doi.org/10.1109/TNNLS.2020.2995319.

[42] L. Yu, S. Wang, X. Li, C.-W. Fu, P.-A. Heng, Uncertainty-aware self-ensembling model for semi-supervised 3D left atrium segmentation, in: D. Shen, T. Liu, T.M. Peters, L.H. Staib, C. Essert, S. Zhou, P.-T. Yap, A. Khan (Eds.), Medical Image Computing and Computer Assisted Intervention – MICCAI 2019, Springer International Publishing, Cham, 2019, pp. 605–613.

[43] S. Adiga Vasudeva, J. Dolz, H. Lombaert, Leveraging labeling representations in uncertainty-based semi-supervised segmentation, in: L. Wang, Q. Dou, P.T. Fletcher, S. Speidel, S. Li (Eds.), Medical Image Computing and Computer Assisted Intervention – MICCAI 2022, Springer Nature Switzerland, Cham, 2022, pp. 265–275.

[44] Y. Wu, Z. Ge, D. Zhang, M. Xu, L. Zhang, Y. Xia, J. Cai, Mutual consistency learning for semi-supervised medical image segmentation, Med. Image Anal. 81 (2022) 102530, http://dx.doi.org/10.1016/j.media.2022.102530, URL https://www.sciencedirect.com/science/article/pii/S1361841522001773.

[45] P. Wang, J. Peng, M. Pedersoli, Y. Zhou, C. Zhang, C. Desrosiers, Self-paced and self-consistent co-training for semi-supervised image segmentation, Med. Image Anal. 73 (2021) 102146, http://dx.doi.org/10.1016/j.media.2021.102146, URL https://www.sciencedirect.com/science/article/pii/S1361841521001924.

[46] Y. Ouali, C. Hudelot, M. Tami, Semi-supervised semantic segmentation with cross-consistency training, in: The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[47] Z. Ke, D. Qiu, K. Li, Q. Yan, R.W.H. Lau, Guided collaborative training for pixel-wise semi-supervised learning, in: Computer Vision – ECCV 2020, Springer International Publishing, Cham, 2020, pp. 429–445.

[48] X. Chen, Y. Yuan, G. Zeng, J. Wang, Semi-supervised semantic segmentation with cross pseudo supervision, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021, pp. 2613–2622.

[49] J. Liu, C. Desrosiers, Y. Zhou, Semi-supervised medical image segmentation using cross-model pseudo-supervision with shape awareness and local context constraints, in: L. Wang, Q. Dou, P.T. Fletcher, S. Speidel, S. Li (Eds.), Medical Image Computing and Computer Assisted Intervention – MICCAI 2022, Springer Nature Switzerland, Cham, 2022, pp. 140–150.

[50] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: A simple way to prevent neural networks from overfitting, J. Mach. Learn. Res. 15 (56) (2014) 1929–1958, URL http://jmlr.org/papers/v15/srivastava14a.html.

[51] T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollar, Focal loss for dense object detection, in: Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017.

[52] K. He, X. Zhang, S. Ren, J. Sun, Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification, in: Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2015.

[53] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[54] F. Isensee, P.F. Jaeger, P.M. Full, P. Vollmuth, K.H. Maier-Hein, nnU-Net for brain tumor segmentation, 2020, arXiv preprint arXiv:2011.00848.

[55] M. Allan, A. Shvets, T. Kurmann, Z. Zhang, R. Duggal, Y.-H. Su, N. Rieke, I. Laina, N. Kalavakonda, S. Bodenstedt, et al., 2017 Robotic instrument segmentation challenge, 2019, arXiv preprint arXiv:1902.06426.

[56] G. Tziatzios, A. Ebigbo, S.K. Gölder, A. Probst, H. Messmann, Methods that assist traction during endoscopic submucosal dissection of superficial gastrointestinal cancers: a systematic literature review, Clin. Endosc. 53 (3) (2020) 286.

[57] B.H. Menze, et al., The multimodal brain tumor image segmentation benchmark (BRATS), IEEE Trans. Med. Imaging 34 (10) (2015) 1993–2024, http://dx.doi.org/10.1109/TMI.2014.2377694.

[58] S. Bakas, et al., Advancing the cancer genome atlas glioma MRI collections with expert segmentation labels and radiomic features, Sci. Data 4 (1) (2017) 1–13.

[59] S. Bakas, et al., Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the BRATS challenge, 2019, arXiv preprint arXiv:1811.02629.

[60] J. Kiefer, J. Wolfowitz, Stochastic estimation of the maximum of a regression function, Ann. Math. Stat. 23 (3) (1952) 462–466.

[61] H. Robbins, S. Monro, A stochastic approximation method, Ann. Math. Stat. (3) 400–407.

[62] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, PyTorch: An imperative style, high-performance deep learning library, in: Advances in Neural Information Processing Systems 32, Curran Associates, Inc., 2019, pp. 8024–8035.

[63] A. Buslaev, et al., Albumentations: Fast and flexible image augmentations, Information 11 (2) (2020) http://dx.doi.org/10.3390/info11020125, URL https://www.mdpi.com/2078-2489/11/2/125.

[64] T.G. Dietterich, Approximate statistical tests for comparing supervised classification learning algorithms, Neural Comput. 10 (7) (1998) 1895–1923, http://dx.doi.org/10.1162/089976698300017197.

[65] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, B. Schiele, The cityscapes dataset for semantic urban scene understanding, in: Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[66] M. Everingham, S.M.A. Eslami, L. Van Gool, C.K.I. Williams, J. Winn, A. Zisserman, The pascal visual object classes challenge: A retrospective, Int. J. Comput. Vis. 111 (1) (2015) 98–136.

[67] S. Kornblith, M. Norouzi, H. Lee, G. Hinton, Similarity of neural network representations revisited, in: K. Chaudhuri, R. Salakhutdinov (Eds.), Proceedings of the 36th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, 97, PMLR, 2019, pp. 3519–3529, URL https://proceedings.mlr.press/v97/kornblith19a.html.