Using Sparse Patch Annotation for Tumor Segmentation in Histopathological Images

Yiqing Liu $^{1,\dagger}$, Qiming He $^{1,\dagger}$, Hufei Duan $^{1}$, Huijuan Shi $^{2}$, Anjia Han $^{2,*}$ and Yonghong He $^{1,*}$

$^1$ Institute of Biopharmaceutical and Health Engineering, Tsinghua Shenzhen International Graduate School, Shenzhen 518055, China
$^2$ Department of Pathology, The First Affiliated Hospital, Sun Yat-sen University, Guangzhou 510080, China
* Correspondence: hananjia@mail.sysu.edu.cn (A.H.); heyh@sz.tsinghua.edu.cn (Y.H.)
† These authors contributed equally to this work.

Abstract: Tumor segmentation is a fundamental task in histopathological image analysis. Creating accurate pixel-wise annotations for such segmentation tasks in a fully-supervised training framework requires significant effort. To reduce the burden of manual annotation, we propose a novel weakly supervised segmentation framework based on sparse patch annotation, i.e., only small portions of patches in an image are labeled as ‘tumor’ or ‘normal’. The framework consists of a patch-wise segmentation model called PSeger, and an innovative semi-supervised algorithm. PSeger has two branches for patch classification and image classification, respectively. This two-branch structure enables the model to learn more general features and thus reduce the risk of overfitting when learning sparsely annotated data. We incorporate the idea of consistency learning and self-training into the semi-supervised training strategy to take advantage of the unlabeled images. Trained on the BCSS dataset with only 25% of the images labeled (five patches for each labeled image), our proposed method achieved competitive performance compared to the fully supervised pixel-wise segmentation models. Experiments demonstrate that the proposed solution has the potential to reduce the burden of labeling histopathological images.

Keywords: histology images; tumor segmentation; sparse annotation; weakly-supervised learning; semi-supervised learning

1. Introduction

Deep learning has made rapid development and remarkable progress in pathological image analysis in recent years [1–7]. The application of deep learning in pathological diagnosis and prognosis cannot be imagined without high-quality annotations. However, acquiring precise annotations is difficult since it requires knowledge of pathology and is time-consuming and labor-intensive, particularly for segmentation tasks that involve manually outlining the specific structures.

Unfortunately, experts with a wealth of pathological knowledge, the source of high quality and clean clinical tagging of key data, are often scarce and have limited energy to spend on data labeling. Therefore, deep-learning methods based on sparsely annotated labels are critical to reducing their workload of labeling and pushing the application of deep learning in the field of pathology. Tumor segmentation has been one of the most fundamental tasks in digital pathology for accurate diagnosis.

Since a whole slide image (WSI) usually has an extremely high resolution, e.g., 50,000 × 50,000 pixels, common practice is to crop it into smaller images and assign each of them a label for model training. There are two typical models, including an image-wise segmentation model [8–13] and pixel-wise segmentation model [14–18]. An image-wise segmentation model predicts whether the given image contains tumorous regions. A binary label (‘tumor’ or ‘normal’) is assigned to each image in the training set to train these models. However, the performance of an image-wise segmentation model is limited...
by the insufficiency of the labeling information. Since a mere binary label ‘tumor’ cannot reflect the location and proportion of the tumor, assigning the same label ‘tumor’ to different images as long as they contain the tumor may confuse the network training and lead to inaccurate segmentation results, which is unacceptable—in particular for small tumors.

In contrast, a pixel-wise segmentation model can produce more accurate segmentation results. However, pathologists must annotate the tumor regions as masks to train the model, which takes much more time and energy. More importantly, unlike other medical images, such as MRI and CT images, pathology images usually lack a clear distinction between the normal and tumor areas [19], which imposes additional difficulties for labeling.

To compensate for the shortcomings of the above two methods, we propose the concept of the patch-level label. Note that, in our proposed method, a patch refers to a grid cell of an image, which is different from the definition in other articles [11,18]. Suppose we divide the image with the size of $224 \times 224$ pixels into a $14 \times 14$ grid, then the patch size is $16 \times 16$ pixels. For each image in the training set, pathologists only need to annotate several (usually 5–10) patches as the label, significantly saving the annotation cost. The left of Figure 1 shows different types of labels.

Figure 1. Left: The illustration of different types of labels. (a) Pixel-level label, where the red area denotes tumor region and the black area denotes non-tumor region. (b) Image-level label, suggesting that the image contains tumor region. (c) Patch-level label (proposed), where the red patches and green patches are manual annotations indicating tumor and non-tumor regions, respectively. Right: A software we developed for sparse patch annotation.

We designed a patch-wise segmentation model called Pseger to accommodate this new label. It has two branches for image classification and patch classification, respectively. The image classification is an auxiliary task that helps improve the performance of the patch classification branches. Due to the superior performance the Transformer-based networks [20] have achieved in recent years, we select Swin Transformer [21], a representation of them as the backbone of the model. Moreover, this method can be easily extended to other backbones.

To take advantage of the unlabeled data, we trained our Pseger with an innovative semi-supervised algorithm. The algorithm is developed based on the characteristics of the patch-level label, integrating the ideas of consistent learning [22] and self-training [23]. The contributions of this paper are summarized as follows:

- We proposed the concept of sparse patch annotation for tumor segmentation, which can significantly reduce the annotation burden. To achieve this new way of labeling, we developed an annotation tool (Figure 1, right).
- In order to handle this new label, we created a patch-wise segmentation model called Pseger, which was equipped with an innovative semi-supervised algorithm to make full use of the unlabeled data.
- We comprehensively evaluated our proposed method on two datasets. The experimental results showed that when trained with only 25% labeled data (five patches
for each labeled image), our approach can yield a competitive result compared to the pixel-wise segmentation models trained using 100% labeled data. The ablation study showed the effectiveness of the semi-supervised algorithm.

2. Related Works

2.1. Weakly-Supervised Learning

Pixel-level labels require a considerable amount of time and effort, and the frequently occurring manual errors may give the network the wrong guidance. Weakly-supervised learning (WSL) has recently emerged as a paradigm to relieve the burden of dense pixel-wise annotations [24]. Many WSL techniques have been proposed, including global image-level labels [25,26], scribbles [19,27], points [28,29], bounding boxes [30,31], and global image statistics, such as the target-region size [32,33].

Although these weakly supervised methods have achieved good performance in natural and medical image segmentation, most weak annotations may not necessarily be best or most suited for tumor segmentation. As mentioned above, the image-level label cannot reflect the location and proportion of the tumor, which may result in inaccurate segmentation results. Other label types are more suitable for segmentation tasks where the instances have clear boundaries, such as glands and nuclei. Nevertheless, the boundary between the normal and the tumor area in pathology images is usually fuzzy and ambiguous. Unlike existing weak annotations, we propose patch-level annotation for patch-wise tumor segmentation.

2.2. Multi-Task Learning

Multi-task learning is an emerging field in machine learning that seeks to improve the performance of multiple related tasks by leveraging useful information among them [34]. A deep-learning model for multi-task learning usually consists of a feature extractor shared by all the tasks and multiple branches for each task. In recent years, multi-task learning has been widely exploited in the field of pathological image analysis [18,35,36]. For example, Wang et al. [18] proposed a hybrid model for pixel-wise HCC segmentation of H&E-stained WSIs. The model had three subnetworks sharing the same encoder, corresponding to three associated tasks. Guo et al. [37] employed a classification model to filter images containing tumorous regions and subsequently refined the segmentation results by a pixel-wise segmentation model. Inspired by these seminal works, we adopted a two-branch model, one branch for image classification and another for patch segmentation, to learn more general features and thus reduce the risk of overfitting.

2.3. Semi-Supervised Learning

Semi-supervised learning (SSL) is a combination of both supervised and unsupervised learning methods, in which the network is trained with a small amount of labeled data and a large amount of unlabeled data. SSL methods can make full use of the information provided by unlabeled data, thereby improving the model performance. In recent years, SSL methods have been widely used in the computer vision field [38–43]. There are two common SSL strategies, including consistent learning [22] and self-training [23]. The general idea of consistent learning is that model prediction should keep constant under different perturbations to the input. This method allows for various perturbations to be designed depending on the characteristics of the data and the network. For instance, Xu et al. [40] proposed two novel data augmentation mechanisms and incorporated them into the consistency learning framework for prostate ultrasound segmentation.

Another strategy, self-training, can be broadly divided into four steps. First, train a teacher model using labeled data. Second, use a trained teacher model to generate pseudo labels for unlabeled images. Third, learn an equal-or-larger student model on labeled and unlabeled images. Finally, use the student as a teacher and repeat the above procedures several times. Wang et al. [41] proposed a few-shot learning framework by combining ideas of semi-supervised learning and self-training. They first adopted a teacher-student model
in the initial semi-supervised learning stage and obtained pseudo labels for unlabeled data. Then, they designed a self-training method to update pseudo labels and the segmentation model by alternating downsampling and cropping strategies.

3. Materials and Methods

Here, we propose a novel patch-wise segmentation model called PSeger. Equipped with an innovative semi-supervised algorithm, it can learn from the patch-level label and take advantage of the unlabeled data. Figure 2 gives an overview of the training procedure. It involves three steps: (1) basic training; (2) pseudo label generation; and (3) consistency learning. They are described in detail in the following. The information about the two datasets we used is also described later.

Step 1. Basic training: 
Step 2. Pseudo label generation: 
Step 3. Consistency learning: 

Figure 2. Overview of the framework for training PSeger. $DA(\cdot)$ indicates data augmentation module.

3.1. Basic Training

Since the idea of patch-level label is inspired by Vision Transformer (ViT) [20], we take it as the backbone of PSeger to illustrate the process of basic training. An overview of the model is depicted in Figure 3, which consists of an embedding projection module, a sequence of transformer encoder blocks, and two classifiers for image classification and patch classification, respectively. In the process of forward propagation, an input image $x \in \mathbb{R}^{H \times W \times N_C}$ ($H$, $W$, and $N_C$ represent the height, width, and number of channels of $x$, respectively) is first flattened into $M = HW/P^2$ non-overlapped patches with the size of $P \times P$ pixels. Then, a 2-D convolution operation is employed to obtain patch embeddings, supplemented with position encoding:

$$z_0 = \left[ x^1P_E; x^2P_E; \ldots; x^MP_E \right] + P_{E}^{pos},$$

where $z_0 \in \mathbb{R}^{M \times L}$ ($L$ represents the embedding length) is the input of the first transformer encoder block, $x^k \in \mathbb{R}^{P \times P \times C}$ is the $k$th patch, $P_E$ is the embedding projection, and $P_{E}^{pos}$ is the position encoding. Then, the embeddings are processed by the transformer encoder blocks. Each block includes a multi-head self-attention (MSA) [44] module and a multi-layer perceptron (MLP) module, both of which are operating as residual operators, and with a layer normalization (LN) [45]. The output of the $l$th transformer encoder block can be described as follows,

$$z'_l = MSA(LN(z_{l-1})) + z_{l-1}, \quad l = 1 \ldots L,$$

$$z_l = MLP(LN(z'_l)) + z'_l, \quad l = 1 \ldots L.$$
where \( z_L \) is the final output of the transformer encoder. Each element of the output \( z^k_L \in z_L \) contains contextual features due to the attention mechanism, which makes it possible to classify a patch based on the information of the related patches. We adopt an MLP head \( H_{\text{patch}} \) for patch classification. By these means, \( z_l \) processed by an LN is sent to \( H_{\text{patch}} \) before applying a softmax function to obtain predictions of each patch:

\[
\hat{y} = \text{Softmax}(H_{\text{patch}}(LN(z_L))),
\]

where \( \hat{y} \in \mathbb{R}^{M \times C} \) are the patch predictions, and \( C \) is the number of categories.

In addition to the patch classifier, we introduce an auxiliary image classifier \( H_{\text{image}} \) to the network, which determines whether an input image has a tumor or not. The main motivation for use of image classifier is to help the patch classifier achieve better performance, since in multi-task learning the network tends to find more representative features shared by different tasks [18]. Similar to the patch classifier, the image classifier receives the average of the \( L \)th transformer encoder output \( z^k_L \in \mathbb{R}^{M \times L} \) with an LN, and produces the classification result \( \hat{y}_{\text{img}} \in \mathbb{R}^C \) through a softmax function:

\[
\hat{y}_{\text{img}} = \text{Softmax}\left( H_{\text{image}} \left( LN\left( \frac{1}{M} \sum_{i=1}^{M} z^k_L \right) \right) \right).
\]

The loss function for the basic training is defined as:

\[
L_{\text{sup}} = L_{\text{patch}} + \alpha L_{\text{img}},
\]

where \( L_{\text{img}} \) and \( L_{\text{patch}} \) are the losses for image classification task and patch classification task, respectively. \( \alpha \) is a weighting factor for the two losses. Both \( L_{\text{img}} \) and \( L_{\text{patch}} \) are cross-entropy loss functions; however, \( L_{\text{patch}} \) only considers the annotated patches. Specifically, \( L_{\text{patch}} \) is defined as:

\[
L_{\text{patch}} = -\frac{1}{K} \sum_k \sum_c y^k \log \hat{y}^{(k,c)},
\]

where \( K \) is the number of the labeled patches in the sample \( x \), \( C \) is the number of classes, \( y^k \) is the binary indicator (0 or 1) if class label \( c \) is the correct classification for the \( k \)th patch. \( \hat{y}^{(k,c)} \) is the prediction of the \( k \)th patch at the \( c \)th class.

**Figure 3.** Illustration of PSeger (using Vision Transformer as backbone).
3.2. Pseudo Label Generation

After the basic training process, the model with the best patch classification accuracy on the validation set is used to generate the pseudo labels for samples in the unlabeled data \( X_U \), as is depicted in Figure 4. The trained model receives as input an image \( x_i \in X_U \) and infers the image prediction \( ˆy_{i,img} \) and patch predictions \( ˆy_i \), which are subsequently transformed into the image probability \( p_{i,img} \) and patch probabilities \( p_i \) by the softmax function. The latter are then ranked by their dominant values. We move \( x_i \) from \( X_U \) to \( X_L \) along with its pseudo label if \( p_{i,img} \) and ranked \( p_i \) (denoted as \( r(p_i) \)) meet the following criteria:

1. \( \max(p_{i,img}) > \tau_1 \), where \( \tau_1 \) is the confidence threshold for the image prediction.
2. \( \max(r(p_i[K])) > \tau_2 \), where \( \tau_2 \) is the confidence threshold for the patch prediction.
3. \( \forall k \in [1,K], \argmax(r(p_i)[k]) = \argmax(p_{i,img}) \), which means the patch predictions should remain consistent with the image prediction.

We made some attempts with small-scale data in the early stage and found that the image prediction confidence scores were high (usually above 0.9); however, the patch prediction confidence scores were relatively low (usually below 0.7). Therefore, we empirically set \( \tau_1 = 0.8 \) and \( \tau_2 = 0.6 \).

![Figure 4. Illustration of the pseudo label generation process. Note that the ranked top \( K \) probabilities \( r(p_i) \) only displays the dominant values for each \( r(p_i)[k], k \in [1,K] \). For example, if \( r(p_i)[k] \) is 'tumor': 0.6, 'normal': 0.4, then the dominant value of \( r(p_i)[k] \) is 'tumor': 0.6. Thus, \( \max(r(p_i)[k]) = 0.6 \) and \( \argmax(r(p_i)[k]) = \text{‘tumor’} \).

3.3. Consistency Learning

When the step of pseudo label generation is finished, the model begins to retrain on the updated training set \( X_L \). The details are as follows. First, for an input image \( x \in X_L \), it is transformed into \( \text{aug}_x \) and \( \text{aug}_x' \) by twice independent data augmentation operation. Then, the student model and the teacher model take them as input and output two sets of patch predictions \( ˆy \) and \( ˆy' \), respectively. These two sets should remain consistent based on the smoothness assumption in semi-supervised learning [46]. Therefore, we apply the KL divergence consistency loss between \( ˆy \) and \( ˆy' \):

\[
L_{cons} = -\frac{1}{M} \sum_m \sum_c \frac{g^{(m,c)}}{\hat{g}^{(m,c)}} \log \frac{\hat{g}^{(m,c)}}{g^{(m,c)}}.
\]
where $M$ is the number of patches in the sample $x$; $C$ is the number of categories; $\hat{y}^{(m,c)}$ and $\hat{y}'^{(m,c)}$ are the predictions of the $m$th patches at the $c$th category. Thus, the total loss function can be written as,

$$L_{total} = L_{sup} + \lambda(E)L_{cons},$$  

(9)

where $L_{sup}$ is previously defined in Equation (6). $\lambda(E)$ is a function of training epoch index $E$, which helps control the balance between the supervised loss and the consistency loss. As is the case with other consistency learning methods [40,47], we use a Gaussian ramp-up function as $\lambda(E)$:

$$\lambda(E) = \begin{cases} 
\lambda_{max} \cdot \exp[-5(1 - \frac{E}{E_{max}})^2], & E < E_{max} \\
\lambda_{max}, & \text{otherwise}
\end{cases}$$  

(10)

where $E$ is the epoch index. When $E = E_{max}$, $\lambda$ reaches the maximum weight $\lambda_{max}$ for the consistency loss. We empirically set $\lambda_{max}$ to 1 and $E_{max}$ to 20 epochs. For the student model, the parameters $\theta$ are updated through back-propagation algorithm by minimizing $L_{total}$. For the teacher model, the parameter $\theta'$ are initially set to $\theta_0$ and updated by computing the exponential moving average of $\theta$:

$$\theta'_t = \alpha \theta'_{t-1} + (1 - \alpha) \theta_t,$$  

(11)

where $t$ represents the index of the global training steps. $\alpha$ helps control the speed at which the teacher model parameters $\theta'$ are updated, and we empirically set it to 0.99.

### 3.4. Datasets

We evaluated our proposed method on a public dataset BCSS [48] and an in-house dataset. BCSS dataset includes 151 hematoxylin and eosin-stained images corresponding to 151 histologically-confirmed breast cancer cases. The mean image size is 1.18 mm$^2$ (SD = 0.80 mm$^2$). We followed the train-test splitting rule (https://bcsegmentation.grand-challenge.org/Baseline/ (accessed on 1 June 2022)) that the images from these institutes were used as an unseen testing set to report accuracy: OL, LL, E2, EW, GM, and S3. (The abbreviations stand for tissue source sites (For more details, see https://docs.gdc.cancer.gov/Encyclopedia/pages/TCGA_Barcode/) (accessed on 1 June 2022)). Then, the remained 108 images were cropped into 27,207 smaller images (with the size of 224 × 224). We used 1018 of these smaller images for validation and the remained were for training.

The in-house dataset came from Department of Pathology, the First Affiliated Hospital of Sun Yat-sen University, China. This study was approved by the Ethics Committee of First Affiliated Hospital of Sun Yat-sen University, and data collection were performed in accordance with relevant guidelines and regulations. The dataset contains 28,187 images from 111 cases (WSIs). We used the images of 84 cases for training and validation, and the images from the remaining cases for test. For the training set, 292 images were from the non-tumor regions, labeled as ‘normal’.

A total of 24,971 images were from tumor regions but many of them did not contain any tumor cells. We selected 407 out of these images and labeled 10 patches for each images using our self-developed annotation tool. Among these labeled images, if one contains any tumor cells, then at least one patch will be labeled as ‘tumor’, and the image label will be ‘tumor’, as well. Details about the BCSS dataset and the in-house dataset are shown in Tables 1 and 2, respectively.
Table 1. Summary of the BCSS dataset.

| Cases/WSIs/ROIs | 151 |
|-----------------|-----|
| ROIs for training and validation | 106 |
| Images (224 × 224) for training | 26,189 |
| Images (224 × 224) for validation | 1018 |
| ROIs for test | 45 |
| Images (224 × 224) for test | 9444 |

Table 2. Summary of the in-house dataset.

| Cases/WSIs | 111 |
|------------|-----|
| Cases for training and validation | 84 |
| Patch-level-labeled Images (224 × 224) for training | 407 |
| Patch-level-labeled Images (224 × 224) for validation | 292 |
| Images (224 × 224) from non-tumor regions for training | 222 |
| Unlabeled images (224 × 224) for training | 24,564 |
| Cases for test | 27 |
| Patch-level-labeled Images (224 × 224) for test | 2702 |

4. Results

4.1. Experimental Setup

4.1.1. Training Settings

In the training step, we employed the AdamW optimizer [49] with a base learning rate of $5 \times 10^{-4}$. For the learning rate schedule, we adopted a linear warmup for five epochs (the warmup learning rate was $5 \times 10^{-7}$), followed by cosine annealing for 20 epochs. The batch size was 16, and the backbones used for Pseger were pre-trained on ImageNet. All experiments were done with a RTX 3090. There are five training strategies for Pseger:

- **Baseline**: train the model only on the labeled data.
- **Baseline+CL**: train the model only on the labeled data with consistency learning.
- **Baseline+CL with $X_u$**: train the model on both the labeled data and unlabeled data with consistency learning.
- **Baseline+ST with $X_u$**: first train the model on the labeled data, then use the trained model to infer the pseudo labels of the unlabeled data, and finally retrain the model on both the labeled data and pseudo-labeled data.
- **Baseline+ST+CL with $X_u$**: first train the model on the labeled data, then use the trained model to infer the pseudo labels of the unlabeled data, and finally retrain the model on both the labeled data and pseudo-labeled data with consistency learning.

4.1.2. Evaluation Metrics

In the experiment of comparison with segmentation models, we choose Intersection over Union (IoU) as the evaluation indicator, which is calculated as follows,

$$\text{IoU} = \frac{A \cap B}{A \cup B}, (12)$$

where A and B are the predicted tumor area and ground truth, respectively. The final IoU score is obtained by averaging the IoU for each RoI in the BCSS test set.

In the ablation study, since our in-house dataset has no pixel-wise annotations, we select patch-level and image-level Acc, AUC, and F1 as evaluation indicators. AUC (Area Under the Curve) score is simply the area under the Receiver Operating Characteristic (ROC) curve. Acc and F1 are calculated as follows,

$$\text{Acc} = \frac{TP + TN}{TP + TN + FP + FN}, (13)$$

$$\text{F1} = \frac{2TP}{2TP + FP + FN}, (14)$$
where TP, TN, FP, and FN are true positive, true negative, false positive, and false negative, respectively. The final scores of each evaluation indicator are calculated by averaging the score for each image in BCSS or the in-house test set.

4.2. Comparison with Segmentation Models

We compared our proposed method to a variety of segmentation models on the BCSS dataset (Figure 5). We trained PSeger with two strategies: Baseline and Baseline+ST+CL with \( X_u \). Five patches were labeled for each image in the labeled training set, and ratios of labeled training data were from 1\% to 25\%. In comparison, we chose two architectures of segmentation models, DeepLabv3+ [50] and Unet++ [51], and equipped them with six backbones: ResNet18, ResNet34, ResNet50 [52], EfficientNet-B1, EfficientNet-B3 [53], and RegNetX-1.6GF [54], respectively.

Therefore, 12 segmentation models were trained and tested on the BCSS dataset. These segmentation models and the training and test steps were implemented base on Segmenta-

![Figure 5. Comparison between our proposed method and pixel-wise segmentation models on the BCSS dataset.](image)

Left: IoU values of PSeger trained on different ratios of labeled training data by two training strategies (Baseline, Baseline+ST+CL with \( X_u \)). Right: IoU values of different segmentation models trained on the full training set. The values of the black dotted lines in the left and right are both 80.31, representing the IoU that PSeger (trained by Baseline+ST+CL with \( X_u \) on the training set with 25\% labeled data) and the third-best segmentation model (DeepLabv3plus+EfficientNet-b1: IoU = 80.31 ± 0.95\%) out of 12 segmentation models.

4.3. Visualization of Segmentation Results

To further compare our proposed method with the pixel-wise segmentation method, we selected one of the best performing PSegers (trained by Baseline+ST+CL with \( X_u \) with 25\% images in the training set labeled, IoU = 80.65\%) and compared it with the best performing model in segmentation models (Unetplusplus+EfficientNet-b3, IoU = 81.74\%), as is shown in Figure 6.

In general, the performance of PSeger is comparable to that of Unetplusplus+EfficientNet-b3. The largest prediction differences aroused in case 1 and case 4. In case 1, PSeger performed worse because of more false detection on non-tumorous area; in case 4, Unetplusplus+EfficientNet-b3 performed poorly because of more false positive regions and much more missed detection on tumorous area.

In addition, Figures 7 and 8 display some segmentation results on our in-house dataset. Red and green overlays are tumor regions and non-tumor regions judged by PSeger, respectively, while regions not covered by any overlay are background areas. It can be seen from Figure 8 that our method can accurately segment the invasive tumor and distinguish some non-tumor structures easily confused with tumors.
Figure 6. Comparison between PSeger and Unetplusplus+EfficientNet-b3. **Left:** IoU values (orange bars) of PSeger on 45 tested ROIs and their differences (sky-blue bars) with those of Unetplusplus+EfficientNet-b3. The bar pairs are sorted in descending order of the values of the blue bars. **Right:** Images of four representative cases. From top to bottom, rows are case 1–4, also framed by black dotted rectangles in the bar graph on the left. From left to right, columns are input images, segmentation results by PSeger, segmentation results by Unetplusplus+EfficientNet-b3, and ground truths. Green overlays are annotated or predicted tumor regions, black overlays are ignored regions, and others are non-tumor regions.

Figure 7. Segmentation results on a whole slide image.
4.4. Ablation Study

4.4.1. The Effect of the Amount of Labeling

As an important factor affecting model performance, the amount of labeling is reflected in two aspects: the ratio of annotated training samples to all training samples (denoted as $X_l\%$), and the number of the labeled patches in each sample (denoted as $K$). We conducted experiments on the BCSS dataset to examine the effect of $X_l\%$ and $K$ on the model performance. Figure 9 shows the patch-level AUC values and the image-level AUC values of Baseline and Baseline+ST+CL with $X_u$ under different $X_l\%$ and $K$, respectively, and the results are given as the mean of three experiments performed in duplicate.

Overall, the two AUC values have increased with increased $X_l\%$ and $K$. However, the increase has slowed down with higher $X_l\%$ and $K$. More importantly, Baseline+ST+CL with $X_u$ always outperforms Baseline on image-level AUC, while the former has better patch-level AUC than the latter only when $X_l = 1\%$ or $K = 3$.

![Figure 8](image)

**Figure 8.** Segmentation results of some ROIs. (a,b) Examples of invasive tumor. (c) An example of lobules (a normal structure in breast tissue). (d) An example of lobules surrounded by the invasive tumor. Lobules in (c,d) are outlined by green dashed polygons.

![Figure 9](image)

**Figure 9.** The effect of the amount of labeling.
4.4.2. Training with Different Strategies

To assess the contributions of self-training and consistency learning separately, we performed experiments on the BCSS dataset and the in-house dataset with five different training strategies mentioned before. Each experiment was repeated five times independently and the results are summarized in Tables 3 and 4, where bold and underlined values represent the best and second-best results on a metric, respectively.

Table 3. Model performance on the BCSS dataset with different training strategies.

| Training Strategy | AUC   | Acc   | F1    | AUC_{img} | Acc_{img} | F1_{img} |
|-------------------|-------|-------|-------|-----------|-----------|----------|
| Baseline          | 88.62| 84.28 | 78.63 | 93.25     | 85.91     | 87.21    |
| Baseline+CL       | 88.41| 83.71 | 77.62 | 93.23     | 86.06     | 87.41    |
| Baseline+CL with X_u | 88.67| 84.02 | 78.29 | 93.09     | 86.17     | 87.55    |
| Baseline+ST with X_u | 91.98| 85.58 | 80.05 | 94.05     | 85.48     | 86.39    |
| Baseline+ST+CL with X_u | 92.04| 85.72 | 80.40 | 94.31     | 85.89     | 86.85    |

From Table 3, the strategy of Baseline+ST+CL with X_u helps PSeger achieve the best performance on four of the six indicators (AUC = 92.04%, Acc = 85.72%, F1 = 80.4%, AUC_{img} = 94.31%), significantly higher than the value that the strategy of Baseline has achieved (AUC = 88.62%, Acc = 84.28%, F1 = 78.63%, AUC_{img} = 93.25%). The strategy of Baseline+ST with X_u achieves the second-best performance (AUC = 91.98%, Acc = 85.58%, F1 = 80.05%, AUC_{img} = 94.05%), which is roughly similar to that of Baseline+ST+CL with X_u. Additionally, the performance of Baseline+CL is inferior to that of Baseline. Furthermore, when X_u is involved in the training procedure, the model (Baseline+CL with X_u) performs better than Baseline and has reached the highest in the two indicators of Acc_{img} (86.17%) and F1_{img} (87.55%).

From Table 4, while the performance of PSeger trained by Baseline+ST+CL with X_u on the in-house dataset is still better than that trained by Baseline, combining the two semi-supervised strategies (consistency learning and self-training) does not achieve better performance than either.

4.4.3. Backbone Selections

In this experiment, we used all labeled data in the BCSS training set to train the models with different backbones, including DenseNet121 [56], EfficientNet-B0, EfficientNet-B1 [53], HRNet-w18 [57], ResNet18, ResNet34, ResNet50 [52], ResNeXt-101 (32 × 8d) [58], ViT-base [20], and Swin-base [21] and tested their performance on the BCSS test set (Table 5). The experiment was repeated five times. From the results, the model using Swin-base as backbone achieves the best performance, significantly better than other models.

Nevertheless, the CNN-based models still achieve decent outcomes. It is somewhat surprising that the model using ViT-base as the backbone is not as good as the models using the CNN architecture in the patch-level evaluation indexes; however, it can surpass most CNN architecture models in the image-level evaluation indexes (second only to ResNeXt-101 (32 × 8d)).
Table 5. Model performance on the BCSS dataset using different backbones.

| Backbone           | AUC      | ACC      | F1       | AUC (Image) | ACC (Image) | F1 (Image) |
|--------------------|----------|----------|----------|-------------|-------------|------------|
| DenseNet121        | 94.33 ± 0.06 | 87.47 ± 0.09 | 83.04 ± 0.13 | 95.68 ± 0.12 | 89.67 ± 0.22 | 91.22 ± 0.20 |
| EfficientNet-B0    | 94.76 ± 0.12 | 87.57 ± 0.19 | 83.00 ± 0.37 | 95.66 ± 0.08 | 89.57 ± 0.24 | 91.14 ± 0.18 |
| EfficientNet-B1    | 94.57 ± 0.04 | 87.30 ± 0.04 | 82.71 ± 0.17 | 95.80 ± 0.10 | 89.60 ± 0.07 | 91.15 ± 0.02 |
| HRNet-w18          | 94.31 ± 0.09 | 87.21 ± 0.14 | 82.47 ± 0.27 | 95.99 ± 0.10 | 89.99 ± 0.18 | 91.39 ± 0.14 |
| ResNet18           | 94.03 ± 0.09 | 87.04 ± 0.12 | 82.26 ± 0.21 | 95.35 ± 0.19 | 88.96 ± 0.26 | 90.56 ± 0.22 |
| ResNet34           | 94.35 ± 0.05 | 87.37 ± 0.09 | 82.85 ± 0.16 | 95.72 ± 0.16 | 89.62 ± 0.28 | 91.17 ± 0.21 |
| ResNet50           | 94.11 ± 0.12 | 87.33 ± 0.13 | 82.76 ± 0.29 | 95.94 ± 0.18 | 90.01 ± 0.38 | 91.48 ± 0.30 |
| ResNeXt-101 (32 × 8d) | 94.64 ± 0.09 | 87.58 ± 0.09 | 83.11 ± 0.13 | 96.25 ± 0.07 | 90.34 ± 0.14 | 91.64 ± 0.09 |
| ViT-base           | 94.47 ± 0.07 | 87.39 ± 0.06 | 82.94 ± 0.08 | 96.16 ± 0.09 | 90.20 ± 0.21 | 91.66 ± 0.17 |
| Swin-base          | 95.41 ± 0.05 | 88.40 ± 0.08 | 84.29 ± 0.12 | 96.64 ± 0.10 | 91.47 ± 0.04 | 92.70 ± 0.05 |

5. Discussion

In the ablation study, we first investigated the effect of the amount of labeling on model performance (Figure 9). On the image-level AUC, the model trained by Baseline+ST+CL with $X_u$ was always better than that trained by Baseline under otherwise equal conditions. However, on the patch-level AUC, that was not always true, particularly when $K > 3$ and $X_u% > 1\%$. This meant that the proposed semi-supervised method can effectively improve the image classification performance; however, it enhanced the patch classification performance only when the amount of annotation was small. When the annotation amount increased, the semi-supervised learning method was not as good as the fully-supervised learning method. Further study is therefore needed to optimize semi-supervised training.

Next, we performed experiments on different training strategies (Tables 3 and 4). Both consistency learning and self-training benefited the model, and self-training improved the model performance more significantly. Additionally, combining the consistency learning strategy with the self-training strategy has the potential to fully utilize the pseudo-annotated data and further improve model performance. However, it depends on the dataset and requires appropriate parameter settings to achieve the expected result.

Finally, the experiment of training with different backbones (Table 5) proves that our proposed method is suitable for transformer-based models and models with CNN architecture. By comparing the performance of different models, we found that Swin Transformer was better than CNN models on both image-level metrics and patch-level metrics. In comparison, Vision Transformer was only better than most CNNs on image-level metrics and inferior to many CNNs on patch-level metrics. This may because the patch classification accuracy depends on the ability to capture localized features and the sensitivity to context-driven features. Although Vision Transformer is more sensitive to contextual features than CNN models, its local feature extraction ability is poorer, which affects the final patch classification accuracy.

Our proposed method can be improved in several ways:

- **Hierarchical patch-level label.** Here, we only considered the annotation form at a single scale, which did not take advantage of the information at different magnifications of the pathological images. Therefore, the annotation can be extended to multiple scales, allowing the model to learn from hierarchical information.

- **Automatic patch selection for labeling.** Choosing which patches to label is subjective and will affect the learning effect of the model. Hence, an active learning mechanism [59] can be introduced to automatically find the most informative patches to label, improving learning efficiency.

- **Hybrid CNN-transformer architecture.** In terms of local feature extraction and global feature capture, CNN and transformer have respective advantages, as analyzed before. Therefore, a hybrid CNN-transformer architecture, like in [60,61], might combine the benefits of the two better to achieve greater performance.

- **More advanced semi-supervised algorithm.** Our semi-supervised algorithm still has problems, such as being sensitive to hyperparameters. In the future, ideas from some...
advanced semi-supervised algorithms in recent years, such as Mixmatch [62], can be introduced into the training algorithm. At the same time, some constraints can be added to prevent the model from overfitting, such as the consistency of prediction results between the patch classification branch and the image classification branch.

6. Conclusions
In this work, we proposed a novel form of annotation, sparse patch annotation, and developed an annotation tool to achieve this new way of labeling. We created a patch-wise segmentation model called Pseger to handle this new label, which was equipped with an innovative semi-supervised algorithm to fully utilize the unlabeled data. We compared the proposed method to various pixel-wise segmentation models (Figure 5). It was shown that, when trained with only 25% labeled data (five patches for each labeled image), our model achieved comparable segmentation results with the semantic segmentation models trained on fully pixel-level labeled data.

Our proposed method enables pathologists to focus their time and energy on labeling the representative parts of the image rather than carefully delineating complex boundaries, significantly reducing the annotation burden.

Author Contributions: Conceptualization, Y.L.; methodology, Y.L. and Q.H.; software, H.D.; validation, Q.H., H.D. and H.S.; formal analysis, Q.H.; investigation, Y.L.; resources, A.H.; data curation, H.S.; writing—original draft preparation, Y.L.; writing—review and editing, Q.H. and H.D.; visualization, Q.H.; supervision, A.H. and Y.H.; funding acquisition, Y.H. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by National Science Foundation of China (61875102), Science and Technology Research Program of Shenzhen City (JCYJ20180508152528735), Oversea cooperation foundation, Graduate School at Shenzhen, Tsinghua University (HW2018007), and Tsinghua University Spring Breeze Fund (2020Z99CFZ023).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Our annotation tool is available at: https://github.com/FHDD/PSeger-LabelMe (accessed on 1 June 2022). The public dataset used in this study can be accessed at the following link: https://bcsegmentation.grand-challenge.org/ (accessed on 1 June 2022). The private dataset is available upon reasonable request to the corresponding authors.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations
The following abbreviations are used in this manuscript:

AUC Area Under the Curve
BCSS Breast Cancer Semantic Segmentation
CT Computed Tomography
CL Consistency Loss
HCC Hepatocellular Carcinoma
H&E Hematoxylin and Eosin
IoU Intersection over Union
LN Layer Normalization
MLP Multi-layer Perceptron
MRI Magnetic Resonance Imaging
MSA Multi-head Self-attention
ROC Receiver Operating Characteristic
ROI Region of Interests
SD  Standard Deviation
SSL  Semi-supervised learning
ST   Self-training
ViT  Vision Transformer
WSI  Whole Slide Image
WSL  Weakly-supervised learning

References
1. Campanella, G.; Hanna, M.G.; Geneslaw, L.; Mirafiori, A.; Silva, V.W.K.; Busam, K.J.; Brogi, E.; Reuter, V.E.; Klimstra, D.S.; Fuchs, T.J. Clinical-grade computational pathology using weakly supervised deep learning on whole slide images. *Nat. Med.* 2019, 25, 1301–1309. [CrossRef] [PubMed]

2. Lu, M.Y.; Williamson, D.F.; Chen, T.Y.; Chen, R.J.; Barbieri, M.; Mahmood, F. Data-efficient and weakly supervised computational pathology on whole-slide images. *Nat. Biomed. Eng.* 2021, 5, 555–570. [CrossRef] [PubMed]

3. Coudray, N.; Ocampo, P.S.; Sakellaropoulou, T.; Narula, N.; Snuderl, M.; Fenyo, D.; Moreira, A.L.; Razavian, N.; Tsirigos, A. Classification and mutation prediction from non–small cell lung cancer histopathology images using deep learning. *Nat. Med.* 2018, 24, 1559–1567. [CrossRef] [PubMed]

4. Courtiol, P.; Maussion, C.; Moarri, M.; Pronier, E.; Pilcer, S.; Sefta, M.; Manceron, P.; Toldo, S.; Zaslavskiy, M.; Le Stang, N.; et al. Deep learning-based classification of mesothelioma improves prediction of patient outcome. *Nat. Med.* 2019, 25, 1519–1525. [CrossRef] [PubMed]

5. Kather, J.N.; Pearson, A.T.; Halama, N.; Jäger, D.; Krause, J.; Loosен, S.H.; Marx, A.; Poer, P.; Tacke, F.; Neumann, U.P.; et al. Deep learning can predict microsatellite instability directly from histology in gastrointestinal cancer. *Nat. Med.* 2019, 25, 1054–1056. [CrossRef] [PubMed]

6. Lu, M.Y.; Chen, T.Y.; Williamson, D.F.; Zhao, M.; Shady, M.; Lipkova, J.; Mahmood, F. AI-based pathology predicts origins for cancers of unknown primary. *Nature* 2021, 594, 106–110. [CrossRef] [PubMed]

7. Naik, N.; Madani, A.; Esteva, A.; Keskar, N.S.; Press, M.F.; Ruderman, D.; Agus, D.B.; Socher, R. Deep learning-enabled breast cancer hormonal receptor status determination from base-level H&E stains. *Nat. Commun.* 2020, 11, 5727. [PubMed]

8. Wang, D.; Khosla, A.; Gargeya, R.; Irshad, H.; Beck, A.H. Deep learning for identifying metastatic breast cancer. *arXiv* 2016, arXiv:1606.05718.

9. Qaiser, T.; Tsang, Y.W.; Taniyama, D.; Sakamoto, N.; Nakane, K.; Epstein, D.; Raijput, N. Fast and accurate tumor segmentation of histology images using persistent homology and deep convolutional features. *Med. Image Anal.* 2019, 55, 1–14. [CrossRef] [PubMed]

10. Ni, H.; Liu, H.; Wang, K.; Wang, X.; Zhou, X.; Qian, Y. WSI-Net: Branch-based and hierarchy-aware network for segmentation and classification of breast histopathological whole-slide images. In *International Workshop on Machine Learning in Medical Imaging*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 36–44.

11. Hou, L.; Samaras, D.; Kure, T.M.; Gao, Y.; Davis, J.E.; Saltz, J.H. Patch-based convolutional neural network for whole slide tissue image classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2424–2433.

12. Liu, Y.; Gadeapplii, K.; Norouzi, M.; Dahl, G.E.; Kohlerberge, T.; Boyko, A.; Venugopalan, S.; Timofeev, A.; Nelson, P.Q.; Corrado, G.S.; et al. Detecting cancer metastases on gigapixel pathology images. *arXiv* 2017, arXiv:1703.02442.

13. Mi, W.; Li, J.; Guo, Y.; Ren, X.; Liang, Z.; Zhang, T.; Zou, H. Deep learning-based multi-class classification of breast digital pathology images. *Cancer Manag. Res.* 2021, 13, 4605. [CrossRef] [PubMed]

14. Li, Z.; Tao, R.; Wu, Q.; Li, B. Da-refinenet: A dual input whole slide image segmentation algorithm based on attention. *arXiv* 2019, arXiv:1907.06358.

15. Dong, N.; Kampffmeyer, M.; Liang, X.; Wang, Z.; Dai, W.; Xing, E. Reinforced auto-zoom net: Towards accurate and fast breast cancer segmentation in whole-slide images. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 317–325.

16. Van Rijthoven, M.; Balkenhol, M.; Silina, K.; Van Der Laak, J.; Ciompi, F. HookNet: Multi-resolution convolutional neural networks for semantic segmentation in histopathology whole-slide images. *Med. Image Anal.* 2021, 68, 101890. [CrossRef] [PubMed]

17. Chan, L.; Hosseini, M.S.; Rowllel, C.; Plataniotiess, K.N.; Damaskinos, S. Histosegnet: Semantic segmentation of histological tissue type in whole slide images. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 10662–10671.

18. Wang, X.; Fang, Y.; Yang, S.; Zhu, D.; Wang, M.; Zhang, J.; Tong, K.; Han, X. A hybrid network for automatic hepatocellular carcinoma segmentation in H&E-stained whole slide images. *Med. Image Anal.* 2021, 68, 101914. [PubMed]

19. Cho, S.; Jang, H.; Tan, J.W.; Jeong, W.K. DeepScribble: Interactive Pathology Image Segmentation Using Deep Neural Networks with Scribbles. In Proceedings of the 2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI), Nice, France, 13–16 April 2021; pp. 761–765.

20. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv* 2020, arXiv:2010.11929.
21. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 10012–10022.

22. Tarvainen, A.; Valpola, H. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 1195–1204.

23. Yaliniz, I.Z.; Jegou, H.; Chen, K.; Paluri, M.; Mahajan, D. Billion-scale semi-supervised learning for image classification. arXiv 2019, arXiv:1905.05546.

24. Belharbi, S.; Ben Ayed, I.; McCaffrey, L.; Granger, E. Deep active learning for joint classification & segmentation with weak annotator. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 3–8 January 2021; pp. 3338–3347.

25. Pinckaers, H.; Bulten, W.; van der Laak, J.; Lijtens, G. Detection of prostate cancer in whole-slide images through end-to-end training with image-level labels. IEEE Trans. Med. Imaging 2021, 40, 1817–1826. [CrossRef]

26. Zhou, C.; Jin, Y.; Chen, Y.; Huang, S.; Huang, R.; Wang, Y.; Zhao, Y.; Chen, Y.; Guo, L.; Liao, J. Histopathology classification and localization of colorectal cancer using global labels by weakly supervised deep learning. Comput. Med. Imaging Graph. 2021, 88, 101861. [CrossRef]

27. Lin, D.; Dai, J.; Jia, J.; He, K.; Sun, J. Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 3159–3167.

28. Bearman, A.; Russakovsky, O.; Ferrari, V.; Fei-Fei, L. What is the point: Semantic segmentation with point supervision. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 549–565.

29. Qu, H.; Wu, P.; Huang, Q.; Yi, J.; Yan, Z.; Li, K.; Riedlinger, G.M.; De, S.; Zhang, S.; Metaxas, D.N. Weakly supervised deep nuclei segmentation using partial points annotation in histopathology images. IEEE Trans. Med. Imaging 2020, 39, 3655–3666. [CrossRef]

30. Mahani, G.K.; Li, R.; Evangelou, N.; Sotiropolous, S.; Morgan, P.S.; French, A.P.; Chen, X. Bounding Box Based Weakly Supervised Deep Convolutional Neural Network for Medical Image Segmentation Using an Uncertainty Guided and Spatially Constrained Loss. In Proceedings of the 2022 IEEE 19th International Symposium on Biomedical Imaging (ISBI), Kolkata, India, 28–31 March 2022; pp. 1–5.

31. Liang, Y.; Yin, Z.; Liu, H.; Zeng, H.; Wang, J.; Liu, J.; Che, N. Weakly Supervised Deep Nuclei Segmentation with Sparsely Annotated Bounding Boxes for DNA Image Cytometry. IEEE ACM Trans. Comput. Biol. Bioinform. 2022, early access. [CrossRef]

32. Jia, Z.; Huang, X.; Eric, I.; Chang, C.; Xu, Y. Constrained deep weak supervision for histopathology image segmentation. IEEE Trans. Med. Imaging 2017, 36, 2376–2388. [CrossRef] [PubMed]

33. Kervadec, H.; Dolz, J.; Tang, M.; Granger, E.; Boykov, Y.; Ayed, I.B. Constrained-CNN losses for weakly supervised segmentation. Med. Image Anal. 2019, 54, 88–99. [CrossRef] [PubMed]

34. Zhang, Y.; Yang, Q. An overview of multi-task learning. Natl. Sci. Rev. 2018, 5, 30–43. [CrossRef]

35. Graham, S.; Vu, Q.D.; Jahanifar, M.; Minhas, F.; Sneed, D.; Rajpoot, N. One Model is All You Need: Multi-Task Learning Enables Simultaneous Histology Image Segmentation and Classification. arXiv 2022, arXiv:2203.00077.

36. Cheng, J.; Liu, J.; Kuang, H.; Wang, J. A Fully Automated Multimodal MRI-Based Multi-task Learning for Glioma Segmentation and IDH Genotyping. IEEE Trans. Med. Imaging 2022, 41, 1520–1532. [CrossRef]

37. Guo, Z.; Liu, H.; Ni, H.; Wang, X.; Xu, M.; Guo, W.; Wang, K.; Jiang, T.; Qian, Y. A fast and refined cancer regions segmentation framework in whole-slide breast pathological images. Sci. Rep. 2019, 9, 882. [CrossRef]

38. Shi, F.; Chen, B.; Cao, Q.; Wei, Y.; Zhou, Q.; Zhang, R.; Zhou, Y.; Yang, W.; Wang, X.; Fan, R.; et al. Semi-Supervised Deep Transfer Learning for Benign-Malignant Diagnosis of Pulmonary Nodules in Chest CT Images. IEEE Trans. Med. Imaging 2021, 41, 771–781. [CrossRef]

39. Nguyen, H.H.; Saarakkala, S.; Blaschko, M.B.; Tiulpin, A. Semixup: In-and out-of-manifold regularization for deep semi-supervised knee osteoarthritis severity grading from plain radiographs. IEEE Trans. Med. Imaging 2020, 39, 4346–4356. [CrossRef]

40. Xu, X.; Sanford, T.; Turkbey, B.; Xu, S.; Wood, B.J.; Yan, P. Shadow-consistent Semi-supervised Learning for Prostate Ultrasound Segmentation Methods. IEEE Trans. Med. Imaging 2021, 41, 1331–1345. [CrossRef]

41. Wang, W.; Xia, Q.; Hu, Z.; Yan, Z.; Li, Z.; Wu, Y.; Huang, N.; Gao, Y.; Metaxas, D.; Zhang, S. Few-shot learning by a Cascaded framework with shape-constrained Pseudo label assessment for whole Heart segmentation. IEEE Trans. Med. Imaging 2021, 40, 2629–2641. [CrossRef]

42. Zhang, Y.; Li, M.; Ji, Z.; Fan, W.; Yuan, S.; Liu, Q.; Chen, Q. Twin self-supervision based semi-supervised learning (TS-SSL): Retinal anomaly classification in SD-OCT images. Neurocomputing 2021, 462, 491–505. [CrossRef]

43. Li, D.; Yang, J.; Kreis, K.; Torralba, A.; Fidler, S. Semantic segmentation with generative models: Semi-supervised learning and strong out-of-domain generalization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Virtual, 19–25 June 2021; pp. 8300–8311.

44. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is all you need. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 6000–6010.
45. Touvron, H.; Cord, M.; Douze, M.; Massa, F.; Sablayrolles, A.; Jégou, H. Training data-efficient image transformers & distillation through attention. In Proceedings of the International Conference on Machine Learning, Vienna, Austria, 18–24 July 2021; pp. 10347–10357.

46. Laine, S.; Aila, T. Temporal ensembling for semi-supervised learning. arXiv 2016, arXiv:1610.02242.

47. Li, X.; Yu, L.; Chen, H.; Fu, C.W.; Xing, L.; Heng, P.A. Transformation-consistent self-ensembling model for semisupervised medical image segmentation. IEEE Trans. Neural Netw. Learn. Syst. 2020, 32, 523–534. [CrossRef] [PubMed]

48. Amgad, M.; Elfandy, H.; Hussein, H.; Atteya, L.A.; Elsebaie, M.A.; Sakr, R.A.; Salem, H.S.; Ismail, A.F.; Saad, A.M.; et al. Structured crowdsourcing enables convolutional segmentation of histology images. Bioinformatics 2019, 35, 3461–3467. [CrossRef] [PubMed]

49. Loshchilov, I.; Hutter, F. Decoupled weight decay regularization. arXiv 2017, arXiv:1711.05101.

50. Chen, L.C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-decoder with atrous separable convolution for semantic image segmentation. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 801–818.

51. Zhou, Z.; Siddiquee, M.M.R.; Tajbakhsh, N.; Liang, J. Unet++: A nested u-net architecture for medical image segmentation. In Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support; Springer: Berlin/Heidelberg, Germany, 2018; pp. 3–11.

52. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.

53. Tan, M.; Le, Q. Efficientnet: Rethinking model scaling for convolutional neural networks. In Proceedings of the International Conference on Machine Learning, Long Beach, CA, USA, 10–15 June 2019; pp. 6105–6114.

54. Radosavovic, I.; Kosaraju, R.P.; Girshick, R.; He, K.; Dollár, P. Designing network design spaces. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 10428–10436.

55. Yakubovskiy, P. Segmentation Models Pytorch. 2020. Available online: https://github.com/qubvel/segmentation_models.pytorch (accessed on 1 June 2022).

56. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708.

57. Wang, J.; Sun, K.; Cheng, T.; Jiang, B.; Deng, C.; Zhao, Y.; Liu, D.; Mu, Y.; Tan, M.; Wang, X.; et al. Deep high-resolution representation learning for visual recognition. IEEE Trans. Pattern Anal. Mach. Intell. 2020, 43, 3349–3364. [CrossRef] [PubMed]

58. Xie, S.; Girshick, R.; Dollár, P.; Tu, Z.; He, K. Aggregated residual transformations for deep neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1492–1500.

59. Yang, L.; Zhang, Y.; Chen, J.; Zhang, S.; Chen, D.Z. Suggestive annotation: A deep active learning framework for biomedical image segmentation. In International Conference on Medical Image Computing And Computer-Assisted Intervention; Springer: Berlin/Heidelberg, Germany, 2017; pp. 399–407.

60. Xie, Y.; Zhang, J.; Shen, C.; Xia, Y. Cotr: Efficiently bridging cnn and transformer for 3d medical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Strasbourg, France, 27 September–1 October 2021; Springer: Berlin/Heidelberg, Germany, 2021; pp. 171–180.

61. Dalmaz, O.; Yurt, M.; Çukur, T. ResViT: Residual vision transformers for multi-modal medical image synthesis. arXiv 2021, arXiv:2106.16031.

62. Berthelot, D.; Carlini, N.; Goodfellow, I.; Papernot, N.; Oliver, A.; Raffel, C.A. Mixmatch: A holistic approach to semi-supervised learning. In Proceedings of the 33rd Conference on Neural Information Processing Systems (NeurIPS), Vancouver, BC, Canada, 8–14 December 2019; pp. 1–11.