Pattern Recognition 155 (2024) 110621

Contents lists available at ScienceDirect

Pattern Recognition

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

# A multi-resolution self-supervised learning framework for semantic segmentation in histopathology

Hao Wang [a], Euijoon Ahn [b], Jinman Kim [a,*]

[a] Biomedical Data Analysis and Visualisation (BDAV) Lab, School of Computer Science, Faculty of Engineering, The University of Sydney, Sydney, NSW 2006, Australia

[b] College of Science and Engineering, James Cook University, Cairns, QLD 4870, Australia

ARTICLE INFO

*Keywords:*
Multi-resolution histopathology learning
Self-supervised learning
Semantic segmentation

ABSTRACT

Modern whole slide imaging technique together with supervised deep learning approaches have been advancing the field of histopathology, enabling accurate analysis of tissues. These approaches use whole slide images (WSIs) at various resolutions, utilising low-resolution WSIs to identify regions of interest in the tissue and high-resolution for detailed analysis of cellular structures. Due to the labour-intensive process of annotating gigapixels WSIs, accurate analysis of WSIs remains challenging for supervised approaches. Self-supervised learning (SSL) has emerged as an approach to build efficient and robust models using unlabelled data. It has been successfully used to pre-train models to learn meaningful image features which are then fine-tuned with downstream tasks for improved performance compared to training models from scratch. Yet, existing SSL methods optimised for WSI are unable to leverage the multi-resolutions and instead, work only in an individual resolution neglecting the hierarchical structure of multi-resolution inputs. This limitation prevents from the effective utilisation of complementary information between different resolutions, hampering discriminative WSI representation learning. In this paper we propose a Multi-resolution SSL Framework for WSI semantic segmentation (MSF-WSI) that effectively learns histopathological features. Our MSF-WSI learns complementary information from multiple WSI resolutions during the pre-training stage; this contrasts with existing works that only learn between the resolutions at the fine-tuning stage. Our pre-training initialises the model with a comprehensive understanding of multi-resolution features which can lead to improved performance in the subsequent tasks. To achieve this, we introduced a novel Context-Target Fusion Module (CTFM) and a masked jigsaw pretext task to facilitate the learning of multi-resolution features. Additionally, we designed Dense SimSiam Learning (DSL) strategy to maximise the similarities of image features from early model layers to enable discriminative learned representations. We evaluated our method using three public datasets on breast and liver cancer segmentation tasks. Our experiment results demonstrated that our MSF-WSI surpassed the accuracy of other state-of-the-art methods in downstream fine-tuning and semi-supervised settings.

## 1. Introduction

Whole slide images (WSI) are a high-resolution image produced by a complete microscope slide (also known as virtual microscopy). They supply various microscopic views including nuclear atypia, degree of gland formation, mitosis and inflammation under different image resolutions providing a thorough set of statistics about tissues and tumours. Pathologists use this information by analysing WSIs to assist with primary and secondary (consultation) diagnoses in histopathology [1]. In a standard WSI analysis, pathologists typically need to combine observations from multiple resolution of WSIs due to the variety of tumour growth patterns. As shown in Fig. 1, WSI patch with low resolution which we refer to as the *context images*, provides coarse-grained locations of tumours and the global architectural composition of tissue samples such as the presence of duct. Pathologists then use high-resolution images of each region of interest (ROI) of the tumour section, which we refer to as the *target images*, to analyse more specific information about cells such as the local cellular composition. However, such manual analysis of WSIs is immensely time-consuming and laborious, which requires careful expert examinations [2]. As a result, there has been sustained interest in recent years in building an automated computer-aided diagnosis (CAD) system for WSI analysis.

Automatic and accurate segmentation of WSI is a challenging problem for conventional machine learning methods due to the variations in

* Corresponding author.
*E-mail addresses:* hwan7885@uni.sydney.edu.au (H. Wang), euijoon.ahn@jcu.edu.au (E. Ahn), jinman.kim@sydney.edu.au (J. Kim).

https://doi.org/10.1016/j.patcog.2024.110621
Received 7 February 2024; Received in revised form 4 May 2024; Accepted 20 May 2024
Available online 23 May 2024

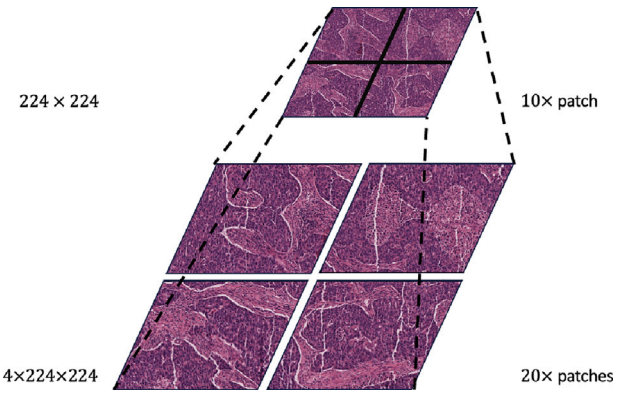


**Fig. 1.** An illustration of extracted patches from multi-resolution WSI. The 10× patches contain coarse-grained information with less details about the cell and the 20× patches contain fine-grained details with less information about surrounding tissues.

cell size, shape, fuzzy boundaries, different cell colours and increasing input image resolutions. Deep learning techniques, such as Convolutional Neural Networks (CNNs), have shown promising performance in improving WSI analysis. For example, Liu et al. [3] designed a patch-based method that splits high-resolution WSI into small-sized patches for fine-grained segmentation. It showed better performance than feeding entire low-resolution WSIs into the network. Similarly, Chan et al. [4] proposed to apply Grad-CAM on WSI patches to generate segmentation masks and then post-processed the results with Conditional Random Field to enhance the identification of the tissue contours. Recently, Zhang et al. [5] designed a dual-task approach where the tumour detection and the segmentation processes were conducted in parallel based on the assumption that knowledge learned from these two tasks are complementary. However, all these methods did not explicitly exploit the inherent multi-resolution features of WSIs, and therefore ignored the complementary information contained in different resolutions. As illustrated in Fig. 1, by using single resolution inputs, model can learn essential structural information, such as tumour locations, from 10× patches while discarding finer details like local cellular composition found in 20× patches, and vice versa. To address the information loss inherent from using a single resolution, researchers [6,7] proposed to use multiple networks as the encoder to process multi-resolution WSI inputs and then aggregate the corresponding feature maps in the decoder path. The usage of multi-resolution WSIs simulated the pathologist examination procedure and showed advantages on segmentation accuracy over single-resolution WSI inputs due to the integration of context and target features.

While using multi-resolution WSIs has shown good results, supervised methods are impacted from the issue of generalisability, e.g., model performance can greatly vary when the training/testing datasets are collected in different settings. To address the above issues, self-supervised learning (SSL), as a label-free algorithm, has received increasing attention. Recent advancements of SSL [8,9], have successfully shown that CNNs are capable of learning meaningful image features without the need of manual labels, and the learned representations are robust in various image analysis tasks. Many recent approaches [10,11] have also validated the effectiveness of SSL on medical image applications that SSL-pre-training is helpful to improve performance when the model is later fine-tuned with smaller set of labelled data. This also applies to WSIs that researchers [12,13] have shown promising outcomes by integrating SSL-pre-trained feature extractors to mitigate data scarcity and domain shift (i.e., transferring from natural images to WSIs) problems in histopathology.

Despite the successful adoption of SSL, these approaches were not designed to learn image features from multi-resolution WSIs. Often, they focused on single resolution features, thereby disregarding the valuable complementary information within multi-resolution WSIs. For instance, low-resolution WSIs assist in locating ROIs, while high-resolution WSIs help extract specific cellular structures. Therefore, lacking multi-resolution learning during pre-training can hinder model convergence in the subsequent fine-tuning stage. To solve this problem, we propose a new Multi-resolution Self-supervised representation learning Framework for WSI semantic segmentation (MSF-WSI) that simultaneously learns both context and target features. We design a novel *Context-Target Fusion Module (CTFM)* to enable the learning of multi-resolution features in the SSL pre-training. CTFM proposes a new pretext task named *masked jigsaw*, wherein target features are randomly masked and shuffled before being concatenated with context features. The assumption is that different augmented views of the same pair (context and target patches) have maximum similarities. In contrast to existing WSI-specific SSL methods, CTFM formulates a new learning task that requires holistic understanding of multi-resolution features. In this task, context (low-resolution) features help the model understand the semantics behind missing and rearranged target (high-resolution) features. This enables the model to leverage the complementary information from different resolution patches, thus enhancing segmentation performance. Furthermore, we propose the *Dense SimSiam Learning (DSL)* approach that improves the extraction of meaningful image features across intermediate layers of the model. Instead of merely contrasting features from the final model layer, as prevalent in other SSL methods, DSL maximises the feature similarity from early layers. In the initial layers, the network detects low-level features, whereas in the deeper layers, it learns higher-level features that encompass broader patterns and semantic information. Our proposed DSL imitates the examination procedure of the pathologist in their reading of WSIs and in doing so, our model is able to learn low-level WSI features (i,e., edges, texture and colours) from the early layers. We evaluated our framework by comparing it against several State-Of-The-Art (SOTA) supervised and SSL approaches using three public histopathology segmentation datasets. Our proposed MSF-WSI framework has demonstrated superior performances compared to other approaches under the fine-tuning and semi-supervised settings.

## 2. Related work

### 2.1. Recent supervised WSI segmentation works

Since the appearance of neural networks and digital pathology, CAD system for WSI segmentation has shown promising results in modern clinical practice [14–16]. For instance, Chen et al. [17] proposed a contour-aware Fully Convolution Network (FCN), which classified object appearance (e.g., textures, colours, etc.) and contour information, to segment glands and nuclei. This contour-aware design was then adopted by Van Eycke et al. [18] and combined with CNN architecture U-Net [19] and ResNet [20] for segmentation of glandular epithelium in colon cancer. Graham et al. [21] proposed a HoVer-Net to generate horizontal and vertical distance maps based on the length of cells to their mass centres. By learning how to generate these maps, the model can leverage a shape prior to assist the prediction of nuclear segmentation mask. Recent works have attempted to further improve algorithm performance by integrating multiple tasks [5]. For example, Graham et al. [22] designed a framework for the segmentation and the classification of nuclei, glands, lumina, and different tissue regions, using data from several independent sources. By solving different tasks using the shared feature maps, these models learned and benefited from complementary information among tasks.

Due to the diverse appearance of target objects (e.g., shape, size and texture), it is essential to extract features from multiple scales of images for accurate segmentation. Prior works that introduced multi-scale feature maps are not applicable for WSI due to the gigapixel dimension. Instead, combining WSI features under different resolutions can alternatively provide both coarse-grained and fine-grained information for

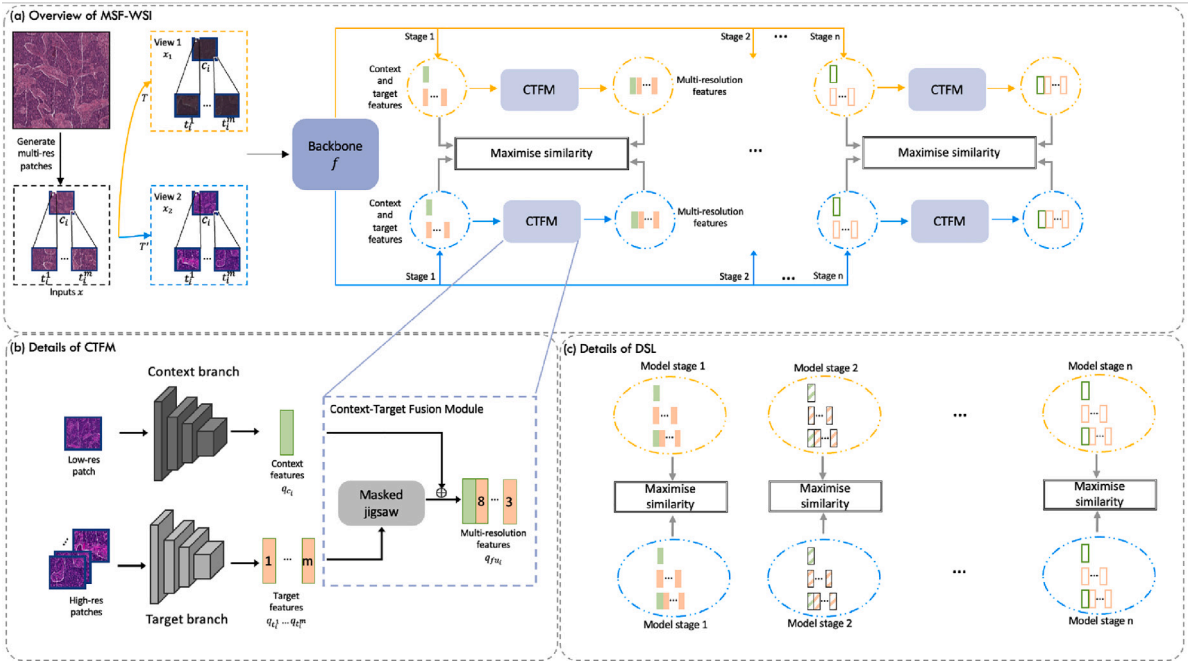


**Fig. 2.** (a) Overview of MSF-WSI. Initial multi-resolution patches $x$ are transformed by $T$ and $T'$, resulting in two views $x_1$ and $x_2$. These views are input into the backbone network $f$ to extract context and target features, which are further refined by the Context-Target Fusion Module (CTFM) to produce multi-resolution features. Contrastive learning is then applied to maximise similarity within these three feature types. Additionally, Dense SimSiam Learning (DSL) is introduced to maximise the similarities between features at intermediate model layers. (b) Structure of CTFM. Using two distinct but identical structure networks (with non-shared weights), we process two-resolution images separately to generate context and target features. Subsequently, CTFM introduces random masking and shuffling of target features, followed by concatenation with context features to produce multi-resolution features. (c) Structure of DSL. For each of the $n$ model stages, we apply SimSiam contrastive learning to context, target, and multi-resolution features $n$ times. In this figure, a mini-batch size of 1 is used for clarity.

segmentation [6]. For example, Nir et al. [23] extracted image features from different WSI resolutions and integrated them later with support vector machine. van Rijthoven et al. [24] proposed an encoder–decoder network architecture where different resolution patches were processed separately in two CNNs and were then integrated over the decoder path. Similarly, Schmitz et al. [7] developed a family of multi-encoder modules which merged model paths with different WSI resolutions in a spatial relationship-preserving fashion. The performances of these studies, however, were dependent on the availability of training data and cannot generalise to various tumour types.

### 2.2. Recent SSL works for WSI segmentation

Self-supervised learning can obtain meaningful representations by delving intrinsic data characteristics without the involvement of labels such that fine-tuning these representations could yield better performance with faster convergence. This is usually achieved by defining different pretext tasks, such as context prediction [25], solving jigsaw puzzles [26], image colourisation [27] and rotation prediction [28]. Recently, the research community has focused on a variant of SSL, termed contrastive learning, which models the similarity and dissimilarity of images [8,29,30] from different augmented views via data transformations. Additionally, researchers [9,31] delved into a more efficient contrastive style which compared the similarity of image views only without measuring their dissimilarity. Despite of promising results reported in these SOTA methods, direct application of them to WSI segmentation can be compromised due to the differences in image statistics, scale, and task-relevant features between natural images and histopathology images.

One common approach to use SSL in WSI analysis is to simply exchange the ImageNet pre-trained extractor with SSL pre-trained models using algorithms such as contrastive predictive coding [32], momentum contrast [8] and SimCLR [12]. For example, Koohbanani et al. [13] designed a multi-task self-supervised approach that involved formulating both domain-agnostic (e.g., image rotation prediction) and domain-specific (e.g., hematoxylin channel prediction) auxiliary tasks to learn histopathology-related features. Azizi et al. [33] proposed a multi-instance contrastive learning strategy that involved constructing positive pairs using crops from two different images of the same patient case. This approach helped the model learn features that were invariant to both the viewpoint and the tissue conditions. Similarly, Li et al. [34] applied SimCLR to each of WSI resolution separately during the pre-training and the learned resolution-specific features were then aggregated for the subsequent multiple instance learning. Recently, Ciga et al. [12] demonstrated the effectiveness of SSL by building a large diverse pre-training dataset that included samples from various histopathology datasets. Moreover, Wang et al. [35] designed a hybrid model using Transformer and CNN to extract local–global universal feature representations (i.e., cell-level structures and tissue-level contexts) by pre-training it on a massive dataset containing 15 million unlabelled WSI patches. However, a limitation of the aforementioned approaches is that the majority of SSL-based histopathology methods overlooked the intrinsic multi-resolution features present in WSIs, with the exception of DSMIL [34]. DSMIL recognised the potential of leveraging low- and high-resolution information to accommodate the diverse tumour appearances. However, a drawback persists in the absence of effective usage of multi-resolution features. During the pre-training of DSMIL, the context and target features were learned independently. This isolated learning process could lead to sub-optimal fine-tuning results, due to the loss of valuable complementary insights from the different resolutions.

## 3. Method

### 3.1. Overview

An illustration of our framework is depicted in Fig. 2. The process commences with the generation of WSI patches at two resolutions: low (context patches) and high (target patches). Then, we apply an

identical set of random data augmentations, transforming each patch into two different views. These transformed patches are inputted into the backbone to extract resolution-specific image features. We then refine context and target features using our CTFM. This module introduces a masked jigsaw pretext task which concatenates context features with masked shuffled target features to form multi-resolution features. Furthermore, we introduce the DSL mechanism which involves contrastive learning that starts from early layers of the model aiming to help the learning of WSI segmentation representations. Overall, our framework combines *context features*, *target features*, and *multi-resolution features* as inputs to DSL, thereby fostering robust WSI segmentation representation learning.

### 3.2. Multi-resolution SSL pipeline

The overview of our SSL pre-training pipeline is depicted in Fig. 2.a.

#### 3.2.1. Resolution-specific learning

The backbone $f$ comprises two identical models, named context branch and target branch, with unshared weights to process multi-resolution inputs. To learn meaningful resolution-specific WSI representations, two tasks were defined. (i) Context features learning: we fed low-resolution patches into the context branch and followed the standard process of SimSiam [31] as explained in Appendix A; (ii) Target features learning: we processed high-resolution patches through the target branch, following the same procedure as the context feature learning. These two tasks helped each branch of the model to extract context and target information contained in the corresponding resolutions respectively. We defined the loss functions as follows:

$$\mathcal{L}_c = \frac{1}{2}\mathcal{D}(p_c^1, stopgrad(z_c^2)) + \frac{1}{2}\mathcal{D}(p_c^2, stopgrad(z_c^1)) \tag{1}$$

$$\mathcal{L}_t = \frac{1}{2}\mathcal{D}(p_t^1, stopgrad(z_t^2)) + \frac{1}{2}\mathcal{D}(p_t^2, stopgrad(z_t^1)) \tag{2}$$

where $p_c$ and $z_c$ are outputs from context branch, and $p_t$ and $z_t$ are outputs from target branch.

#### 3.2.2. Context-target fusion module

To learn the relationships between WSI resolutions during pre-training, we introduce the *Context-Target Fusion Module (CTFM)* to generate multi-resolution features by the fusion of context and target features (see Fig. 2(b)). This is accomplished through the integration of a third auxiliary task termed the *masked jigsaw*, involving the random occlusion of shuffled image patches. The pseudo-code of CTFM is provided in Appendix B. Given that a single context image $c_i$ corresponds to multiple target images $\{t_i^1, \ldots, t_i^m\}$ due to resolution differences (e.g., a 10× context patch corresponds to 16 of 40× target patches), we extracted context and target features ($q_{c_i}$ and $\{q_{t_i^1}, \ldots, q_{t_i^m}\}$) from these images via the backbone network. These features were subsequently refined by CTFM, where target features were both shuffled and randomly masked with a predefined ratio $r$. The resulting multi-resolution feature $q_{fu_i}$ was obtained through concatenating context features and masked target features (length of $m * r$). The objective of our proposed *masked jigsaw* task is to optimise the similarities between multi-resolution features from different views, as denoted by the equation:

$$\mathcal{L}_{fu} = \frac{1}{2}\mathcal{D}(p_{fu}^1, stopgrad(z_{fu}^2)) + \frac{1}{2}\mathcal{D}(p_{fu}^2, stopgrad(z_{fu}^1)) \tag{3}$$

Here, $p_{fu}$ and $z_{fu}$ represent predicted outputs and projected embeddings derived from $q_{fu}$, respectively. The underlying assumption is that randomly sampled target features from the same set should retain similar semantic information. Additionally, the concatenation of context features with masked shuffled target features introduces a challenging task that requires a holistic understanding of interrelationships among multi-resolution patches.

### 3.3. Dense SimSiam learning

We propose the *Dense SimSiam Learning (DSL)* strategy to enable SSL training at early model stages. The procedure of DSL is illustrated in Fig. 2(c). We used ResNet-18 [20] as the branch backbone and defined the model stages as $S = \{s^1, s^2, s^3, s^4\}$ based on the dimensions of output feature maps, i.e., 56, 28, 14, 7. The corresponding stage features were flatten and denoted as $K$, and we applied independent projectors and predictors for each of stage features resulting the sets of projectors and predictors as $G$ and $H$, respectively. We denote the projected embedding set as $Z$ and the predicted vector set as $P$. Parameters in $K, G, H, Z, P$ have the same format that superscripts come from $S$ indicate values at different stages. Thus, we defined the loss function for the $i_{th}$ model stage as

$$\mathcal{L}^{s^i} = \frac{1}{2}\mathcal{D}(p_1^{s^i}, stopgrad(z_2^{s^i})) + \frac{1}{2}\mathcal{D}(p_2^{s^i}, stopgrad(z_1^{s^i})) \tag{4}$$

and

$$\mathcal{L}_{DSL} = \sum_{i=1}^{4} w_i \cdot \mathcal{L}^{s^i} \tag{5}$$

where $w$ is the loss weight, and $i$ denotes the stage index.

In summary, we conducted DSL for context, target and multi-resolution features. Different sets of projectors and predictors were used, i.e., there were $4 \times 3 = 12$ projectors and $4 \times 3 = 12$ predictors for the proposed DSL. We defined the final loss function as:

$$\mathcal{L} = \mathcal{L}_{DSL,c} + \mathcal{L}_{DSL,t} + \mathcal{L}_{DSL,fu} \tag{6}$$

### 3.4. Fine-tuning and inference

We adopted the previous work HookNet [24], a semantic segmentation network for histopathology, as our baseline method and demonstrated the effectiveness of our pre-training algorithm in improving the model segmentation performance. HookNet is a dual-branch encoder–decoder network using multi-resolution patches as inputs. The information from different resolutions was combined via a "hook" mechanism, where feature maps in the decoder part from the context branch were cropped and concatenated with the bottleneck feature maps in the target branch, as shown in Appendix C. After pre-training the encoder part by the proposed MSF-WSI, we can simply initialise and fine-tune the encoder for semantic segmentation task.

## 4. Experiments and results

### 4.1. Datasets

We used three public WSI semantic segmentation datasets including Camelyon16 [36], Pathology Artificial Intelligence Platform 2019 challenge (PAIP2019) dataset [37], and Breast Cancer Semantic Segmentation (BCSS) dataset [38]. We validated the effectiveness of our framework on breast semantic segmentation (BCSS dataset) and liver tumour segmentation (PAIP2019) datasets by conducting internal testing (pre-train and fine-tune on the same dataset) and external testing (pre-train and fine-tune on different datasets).

#### 4.1.1. Camelyon16 dataset

The Camelyon16 dataset is a large-scale histopathology dataset containing 399 WSIs of sentinel lymph node collected from two independent medical centres. Each of WSI contains pixel-level annotations about tumour areas provided by the pathologists.

For the data pre-processing, we firstly tiled each WSI using sliding window size of 1024 × 1024 with step size of 512. Based on these tiles, the context patches were generated by directly resizing them into 224 × 224 and target patches were generated by applying a window size of 256 × 256 with a step size of 256 on the 1024 × 1024 patches. We then resized back into 224 × 224. Thus, each context patch (10×

magnification) had $4 \times 4 = 16$ corresponding target patches (40× magnification). We followed the official dataset split of training set containing 270 WSIs (160 normal and 110 tumour) and testing set of 129 WSIs. During the pre-training, we used all training set image data without annotations.

#### 4.1.2. PAIP2019 dataset

The PAIP 2019 dataset [37] contains 50 WSIs of liver cancer from 50 patients. Two types of annotation are provided: viable regions of cancer cells for continuous tumour areas, as well as whole cancer regions for boundary between the non-tumorous hepatic lobules and the viable tumour. Additionally, we also generated annotations for "tissue area" which indicates healthy tissue pixels by threshold of (R, G, B) $\leq (235, 210, 235)$. This is consistent with the work in [7] and allows sampling of healthy tissue patches that can be used as meaningful negative examples. During the evaluation, all 3 classes were considered including tissue area, whole tumour area and viable tumour area.

For the data pre-processing, we generated context (5× magnification) and target patches (20× magnification) consistent with the settings used in Camelyon16. We randomly selected 10 out of 50 WSIs as the testing set for the 5-fold cross-validation (CV). For internal testing, we pre-trained models on the training set of each fold to prevent potential information leakage.

#### 4.1.3. BCSS dataset

The BCSS dataset [38] is a subset of TCGA dataset [39] where data were collected from multiple institutes. The BCSS dataset consists of 151 H&E stained WSIs coming from 151 independent breast cancer cases. Total of 5 classes were annotated including Tumour (TUM), Stroma (STR), Lymphocytic infiltrate (LYM), Necrosis (NEC) and Other (OTR). During the evaluation, all 5 classes were considered including TUM, STR, LYM, NEC and OTR.

For the data pre-processing, we generated context (10× magnification) and target patches (40× magnification) consistent with the settings used in the Camelyon16 dataset. To facilitate 5-fold CV, we augmented the official data split by randomly selecting additional 4 folds, allowing us to use distinct testing set. We used the institute-exclusive strategy following the settings described in [38]. Here, none of training data comes from institutes in the testing set, and vice versa. The number of institutes in each testing fold remained consistent with the official testing set.

### 4.2. Model configurations

We used ResNet-18 [20] as the encoder of each branch and applied the default data transformation settings in SimSiam [31] for the pre-training. The resolution difference between the context and target images was set to a ratio of $1 : 4$, and the random masking ratio was set to $1 : 1$ for the CTFM configuration. As for the DSL, we used a three-layer MLP as the projector, where the hidden dimension and output dimension were set to be equal to the input dimension. Each predictor was a two-layer MLP, and the input and output dimensions were identical, but the hidden dimension was a quarter of the input dimension. The weights of each stage were set to $\{0.1, 0.4, 0.7, 1.0\}$. Other related hyperparameters are shown in Appendix D.

For HookNet [24], we used the official code implementation[1] but changed the encoder to ResNet-18 for fair comparisons with other SOTAs. Other parameter configurations were kept the same as the original paper, where $\lambda$ was set to 1 to ignore the context loss. Other related hyperparameters are also shown in Appendix D.

### 4.3. Evaluation

We evaluated model performance by Dice's coefficient (DSC), Intersection over Union (IoU), and pixel accuracy (Acc).

We benchmarked with both single-resolution methods, such as the U-Net model [19] for general medical imaging segmentation, and the WSI-focused Cerberus [22]. Additionally, we compared multi-resolution WSI methods, including HookNet [24] and msY-Net [7]. Furthermore, we evaluated various SSL algorithms, including general SSL strategy SimSiam [31], and WSI-specific approaches like Slf-Hist [12], DSMIL [34], and SSL_CR_Histo [40]. Implementation details are provided below:

- U-Net [19]: a encoder–decoder model architecture which is a supervised method for biomedical image segmentation. We modified this architecture by changing the encoder to ResNet-18. Only target patches were used to generate fine-grained segmentation masks.
- Cerberus [22]: a ResNet-34 weights which was pre-trained by a supervised multi-task learning including segmentation and classification of nuclei, glands, lumina and different tissue regions. We used U-Net and altered the encoder to ResNet-34 and initialised with Cerberus. Only target patches were used for fine-tuning.
- HookNet [24]: a model architecture using multi-resolution WSIs for histopathology semantic segmentation. We trained this model from the scratch in a supervised manner.
- msY-Net [7]: a model architecture using multi-resolution WSIs for histopathology image segmentation. We used their source code[2] and trained this model from scratch using labelled data.
- SimSiam [31]: a popular SSL algorithm for natural images. We applied this algorithm to pre-train context and target branch separately with corresponding resolution patches. Afterwards, we initialised the HookNet encoder with the pre-trained weights and fine-tuned with multi-resolution inputs.
- Slf-Hist [12]: a ResNet-18 weights which was pre-trained by a SSL method proposed for WSI analysis tasks. We used their pre-trained weights to initialise corresponding encoders of HookNet. They used a hybrid dataset built with total of 57 datasets consisting of around 4 million patches.
- DSMIL [34]: a ResNet-18 weights pre-trained by a SSL method proposed for WSI classification. We initialised the HookNet with this weight and fine-tuned it with multi-resolution inputs.
- SSL_CR_Histo [40]: a ResNet-18 weights pre-trained by a SSL method proposed for WSI classification and tumour cellularity quantification. We used this initialisation for HookNet and fine-tuned it with multi-resolution patches.

Two settings were considered for model performance evaluation. (a) Fine-tuning: After initialising with pre-trained weights, models were trained with labels by the full training set and validated by the full validation set. (b) Semi-supervised: After initialising with pre-trained weights, models were trained with labels by a fraction (50%, 10%and 1%) of the training set and validated by the full validation set.

### 4.4. Main results

#### 4.4.1. Fine-tuning results

The results of fine-tuning are presented in Table 1. In the PAIP2019 dataset, our MSF-WSL pre-trained on PAIP2019 had the best performance, yielding the highest mean DSC of 0.9236, a mean IoU of 0.8603, and a mean Acc of 0.9492. Our MSF-WSI, even when pre-trained on Camelyon16, achieved the second-best performance, obtaining a mean DSC of 0.9138, a mean IoU of 0.8454, and a mean Acc of 0.9436. The msY-Net approach had the third-best performance, achieving a mean DSC of 0.9106, a mean IoU of 0.8435, and a mean Acc of 0.9407. Tables 2 and 3 present the detailed per-class performances. Our MSF-WSI pre-trained on PAIP2019 outperformed others, achieving the highest scores for Whole Tumour (mean DSC of 0.86 and mean

[1] https://github.com/DIAGNijmegen/pathology-hooknet

[2] https://github.com/ipmi-icns-uke/multiscale/

**Table 1**
The results of fine-tuning experiments on PAIP2019 and BCSS with 5-fold CV in the format mean(standard deviation). Based on the mean values, the best results are in bold, the second best results are underlined, and the third best results are in italic.

| Method | Pre-training strategy | Pre-training dataset | PAIP2019 | | | BCSS | | |
|---|---|---|---|---|---|---|---|---|
| | | | DSC | IoU | Acc | DSC | IoU | Acc |
| ***Single Resolution Methods*** | | | | | | | | |
| U-Net [19] | Supervised | ImageNet | 0.8940(0.0321) | 0.8194(0.0347) | 0.9290(0.0213) | 0.7492(0.0198) | 0.6097(0.0234) | 0.8997(0.0079) |
| Cerberus [22] | Supervised | 9 histopathology datasets | 0.9001(0.0300) | 0.8281(0.0344) | 0.9337(0.0199) | *0.7850(0.0129)* | *0.6560(0.0158)* | *0.9140(0.0052)* |
| ***Multi-resolution Methods*** | | | | | | | | |
| msY-Net [7] | Random | N/A | *0.9106(0.0320)* | *0.8435(0.0411)* | *0.9407(0.0214)* | 0.7620(0.0122) | 0.6246(0.0151) | 0.9048(0.0049) |
| HookNet [24] | Random | N/A | 0.8962(0.0304) | 0.8231(0.0336) | 0.9311(0.0203) | 0.7458(0.0164) | 0.5956(0.0211) | 0.8984(0.0065) |
| Slf-Hist [12] | SSL | 57 histopathology datasets | 0.9089(0.0241) | 0.8401(0.0301) | 0.9395(0.0161) | 0.7782(0.0159) | 0.6453(0.0192) | 0.9113(0.0064) |
| DSMIL [34] | SSL | Camelyon16 | 0.9003(0.0320) | 0.8303(0.0332) | 0.9337(0.0214) | 0.7640(0.0151) | 0.6274(0.0175) | 0.9056(0.0061) |
| SSL_CR_Histo [40] | SSL | Camelyon16 | 0.8962(0.0326) | 0.8232(0.0342) | 0.9310(0.0218) | 0.7602(0.0197) | 0.6222(0.0237) | 0.9041(0.0079) |
| HookNet + MSF-WSI (ours) | SSL | Camelyon16 | 0.9138(0.0219) | 0.8454(0.0259) | 0.9436(0.0152) | 0.7851(0.0091) | 0.6562(0.0109) | 0.9141(0.0036) |
| SimSiam [31] | SSL | PAIP2019/BCSS | 0.9097(0.0191) | 0.8385(0.0284) | *0.9407(0.0122)* | 0.7704(0.0145) | 0.6345(0.0186) | 0.9082(0.0058) |
| HookNet + MSF-WSI (ours) | SSL | PAIP2019/BCSS | **0.9236(0.0087)** | **0.8603(0.0136)** | **0.9492(0.0059)** | **0.7949(0.0118)** | **0.6672(0.0143)** | **0.9180(0.0047)** |

**Table 2**
The per-class results of fine-tuning experiments on PAIP2019 using with 5-fold CV in the format mean (standard deviation). The best results are in bold. Based on the mean values, the best results are in bold, the second best results are underlined, and the third best results are in italic.

| Method | DSC | | | Acc | | |
|---|---|---|---|---|---|---|
| | Tissue | Whole | Viable | Tissue | Whole | Viable |
| U-Net [19] | 0.6335(0.0241) | 0.8124(0.0254) | 0.9247(0.0298) | 0.9798(0.0023) | 0.8942(0.0311) | 0.9147(0.0318) |
| Cerberus [22] | 0.6581(0.0375) | 0.8263(0.0182) | 0.9286(0.0312) | 0.9805(0.0021) | 0.9000(0.0296) | 0.9214(0.0278) |
| msY-Net [7] | **0.7089(0.0325)** | *0.8473(0.0258)* | 0.9308(0.0337) | **0.9852(0.0027)** | *0.9090(0.0360)* | 0.9234(0.0344) |
| HookNet [24] | 0.6599(0.0292) | 0.8189(0.0227) | 0.9261(0.0292) | 0.9813(0.0024) | 0.8961(0.0304) | 0.9174(0.0298) |
| Slf-Hist [12] | 0.6632(0.0418) | 0.8396(0.0133) | 0.9344(0.0281) | 0.9823(0.0020) | 0.9082(0.0234) | 0.9279(0.0234) |
| DSMIL [34] | 0.6611(0.0392) | 0.8253(0.0264) | 0.9297(0.0295) | 0.9815(0.0024) | 0.8997(0.0307) | 0.9200(0.0319) |
| SSL_CR_Histo [40] | 0.6417(0.0397) | 0.8167(0.0250) | 0.9252(0.0289) | 0.9812(0.0024) | 0.8965(0.0320) | 0.9153(0.0312) |
| HookNet + MSF-WSI (ours) | *0.6902(0.0384)* | 0.8476(0.0107) | 0.9392(0.0268) | *0.9825(0.0026)* | 0.9089(0.0222) | 0.9333(0.0205) |
| SimSiam [31] | 0.6822(0.0492) | 0.8455(0.0171) | *0.9349(0.0178)* | 0.9802(0.0054) | 0.9172(0.0101) | *0.9330(0.0096)* |
| HookNet + MSF-WSI (ours) | 0.6971(0.0400) | **0.8600(0.0101)** | **0.9462(0.0174)** | 0.9829(0.0029) | **0.9230(0.0073)** | **0.9416(0.0083)** |

**Table 3**
The per-class results of fine-tuning experiments on BCSS using with 5-fold CV in the format mean(standard deviation). Based on the mean values, the best results are in bold, the second best results are underlined, and the third best results are in italic.

| | DSC | | | | | Acc | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | TUM | STR | LYM | NEC | OTR | TUM | STR | LYM | NEC | OTR |
| U-Net [19] | 0.7690(0.0397) | 0.6606(0.0180) | 0.6626(0.0227) | 0.5520(0.0713) | 0.4030(0.0343) | 0.8498(0.0095) | 0.8028(0.0094) | 0.9227(0.0144) | 0.9597(0.0127) | 0.9641(0.0036) |
| Cerberus [22] | **0.8200(0.0250)** | **0.7163(0.0177)** | 0.7116(0.0224) | **0.7404(0.0498)** | 0.4631(0.0421) | 0.8786(0.0086) | *0.8287(0.0108)* | 0.9397(0.0088) | *0.9700(0.0056)* | 0.9669(0.0038) |
| msY-Net [7] | 0.7831(0.0323) | 0.6735(0.0251) | 0.6690(0.0219) | 0.3927(0.0731) | 0.4295(0.0587) | 0.8607(0.0083) | 0.8128(0.0099) | 0.9314(0.0077) | 0.9557(0.0092) | 0.9635(0.0041) |
| HookNet [24] | 0.7605(0.0360) | 0.6507(0.0070) | 0.5903(0.1096) | 0.5243(0.1068) | 0.4217(0.0577) | 0.8411(0.0088) | 0.7895(0.0103) | 0.9335(0.0125) | 0.9555(0.0099) | 0.9636(0.0038) |
| Slf-Hist [12] | 0.8093(0.0188) | 0.6968(0.0263) | 0.6767(0.0365) | 0.6787(0.0716) | 0.4428(0.0741) | 0.8683(0.0090) | 0.8200(0.0101) | 0.9352(0.0079) | 0.9691(0.0112) | 0.9640(0.0026) |
| DSMIL [34] | 0.7901(0.0339) | 0.6705(0.0199) | 0.6747(0.0347) | 0.5534(0.1704) | 0.4266(0.0628) | 0.8581(0.0073) | 0.8111(0.0076) | 0.9353(0.0092) | 0.9584(0.0139) | 0.9650(0.0044) |
| SSL_CR_Histo [40] | 0.7844(0.0375) | 0.6625(0.0242) | 0.6597(0.0448) | 0.5882(0.0973) | 0.4035(0.0352) | 0.8516(0.0140) | 0.8121(0.0094) | 0.9275(0.0143) | 0.9642(0.0106) | 0.9651(0.0053) |
| HookNet + MSF-WSI (ours) | *0.8147(0.0210)* | 0.7114(0.0240) | **0.7188(0.0167)** | 0.7231(0.0529) | **0.4697(0.0556)** | *0.8752(0.0058)* | 0.8328(0.0111) | **0.9399(0.0087)** | 0.9719(0.0067) | **0.9673(0.0037)** |
| SimSiam [31] | 0.7932(0.0325) | 0.6863(0.0226) | 0.6832(0.0269) | 0.5925(0.0747) | 0.4281(0.0711) | 0.8613(0.0087) | 0.8186(0.0078) | 0.9310(0.0135) | 0.9659(0.0079) | 0.9649(0.0048) |
| HookNet + MSF-WSI (ours) | 0.8158(0.0294) | *0.7084(0.0246)* | *0.6860(0.0075)* | *0.6795(0.0657)* | *0.4568(0.0650)* | **0.8794(0.0054)** | **0.8349(0.0104)** | *0.9369(0.0091)* | **0.9725(0.0082)** | *0.9661(0.0037)* |

Acc of 0.923) and Viable Tumour (mean DSC of 0.9462 and mean Acc of 0.9416). The msY-Net approach attained the highest score for the Tissue class, with a mean DSC of 0.7089 and a mean Acc of 0.9852.

For the BCSS dataset, our MSF-WSI pre-trained on BCSS achieved better performance, yielding the highest mean DSC of 0.7949, a mean IoU of 0.6672, and a mean Acc of 0.918. Following closely, our MSF-WSI pre-trained on Camelyon16 ranked as the second-best method, achieving a mean DSC of 0.7851, a mean IoU of 0.6562, and a mean Acc score of 0.9141. The Cerberus approach had the third-best results, with a mean DSC of 0.785, a mean IoU of 0.656, and a mean Acc of 0.914. Similar results were achieved when examining per-class performance. Specifically, Cerberus exhibited the most promising DSC performance on TUM (0.82), STR (0.7163), and NEC (0.7404), whereas our MSF-WSI pre-trained on Camelyon16 demonstrated superiority in terms of DSC for LYM (0.7188) and OTR (0.4697). Notably, with respect to the Accuracy metric, our MSF-WSI pre-trained on BCSS excelled, achieving the highest results for TUM, STR, and NEC with scores of 0.8794, 0.8349, and 0.9725, respectively. Meanwhile, the consistent performance of our MSF-WSI pre-trained on Camelyon16 was evident in LYM and OTR, obtaining leading positions with scores of 0.9399 and 0.9673, respectively.

#### 4.4.2. Semi-supervised results

We extended our evaluation to show the model's performance under a semi-supervised setting. Tables 4 and 5 present the experiment results for both the datasets, including DSC, IoU, and Acc scores.

For the PAIP2019 dataset, our MSF-WSI model pre-trained on PAIP2019 exhibited superior performance across all metrics and settings. SimSiam approach was the second best in all scenarios, barring the DSC score when employing 1% of the training data, where it achieved the third-best outcome. Our MSF-WSI model pre-trained on Camelyon16 had the third-best performance in terms of DSC, IoU, and Acc under the 50% and 10% settings. In contrast, the Slf-Hist method had the second-best DSC score and the third-best IoU and Acc scores when training data constituted only 1% of the dataset.

**Table 4**
The results of semi-supervised experiments on PAIP2019 with 5-fold CV in the format mean(standard deviation). Based on the mean values, the best results are in bold, the second best results are underlined, and the third best results are in italic.

| Method | DSC | | | IoU | | | Acc | | |
|---|---|---|---|---|---|---|---|---|---|
| | 50% | 10% | 1% | 50% | 10% | 1% | 50% | 10% | 1% |
| U-Net [19] | 0.8868(0.0309) | 0.8607(0.0196) | 0.7868(0.0306) | 0.8072(0.0329) | 0.7604(0.0261) | 0.6577(0.0398) | 0.9247(0.0206) | 0.9071(0.0131) | 0.8578(0.0204) |
| Cerberus [22] | 0.8959(0.0290) | 0.8789(0.0237) | 0.8501(0.0244) | 0.8214(0.0313) | 0.7932(0.0235) | *0.7488(0.0256)* | 0.9309(0.0193) | 0.9195(0.0159) | 0.9003(0.0159) |
| msY-Net [7] | 0.9019(0.0331) | 0.8598(0.0340) | 0.8129(0.0191) | 0.8306(0.0343) | 0.7663(0.0340) | 0.6840(0.0235) | 0.9346(0.0221) | 0.9010(0.0329) | 0.8753(0.0127) |
| HookNet [24] | 0.8906(0.0312) | 0.8608(0.0219) | 0.7970(0.0218) | 0.8119(0.0344) | 0.7691(0.0197) | 0.6657(0.0286) | 0.9272(0.0208) | 0.9072(0.0146) | 0.8647(0.0145) |
| Slf-Hist [12] | 0.8929(0.0311) | 0.8765(0.0325) | 0.8538(0.0139) | 0.8175(0.0321) | 0.7911(0.0342) | *0.7488(0.0200)* | 0.9288(0.0207) | 0.9177(0.0217) | *0.9027(0.0093)* |
| DSMIL [34] | 0.8933(0.0300) | 0.8728(0.0288) | 0.8425(0.0214) | 0.8179(0.0317) | 0.7837(0.0316) | 0.7329(0.0296) | 0.9290(0.0199) | 0.9152(0.0192) | 0.8950(0.0143) |
| SSL_CR_Histo [40] | 0.8894(0.0287) | 0.8684(0.0277) | 0.8294(0.0175) | 0.8104(0.0316) | 0.7764(0.0297) | 0.7164(0.0194) | 0.9264(0.0191) | 0.9122(0.0184) | 0.8863(0.0117) |
| HookNet + MSF-WSI (ours) + Canelyon16 | *0.9049(0.0209)* | *0.8828(0.0270)* | 0.8458(0.0219) | *0.8326(0.0260)* | *0.8003(0.0261)* | 0.7430(0.0214) | *0.9368(0.0140)* | *0.9219(0.0178)* | 0.8974(0.0146) |
| SimSiam [31] | 0.9116(0.0165) | 0.8860(0.0269) | *0.8536(0.0247)* | 0.8426(0.0211) | 0.8041(0.0304) | 0.7541(0.0283) | 0.9413(0.0110) | 0.9241(0.0179) | 0.9030(0.0165) |
| HookNet + MSF-WSI (ours) + PAIP2019 | **0.9151(0.0072)** | **0.9009(0.0118)** | **0.8613(0.0281)** | **0.8451(0.0120)** | **0.8225(0.0174)** | **0.7674(0.0272)** | **0.9436(0.0049)** | **0.9340(0.0078)** | **0.9076(0.0188)** |

**Table 5**
The results of semi-supervised experiments on BCSS with 5-fold CV in the format mean(standard deviation). Based on the mean values, the best results are in bold, the second best results are underlined, and the third best results are in italic.

| Method | DSC | | | IoU | | | Acc | | |
|---|---|---|---|---|---|---|---|---|---|
| | 50% | 10% | 1% | 50% | 10% | 1% | 50% | 10% | 1% |
| U-Net [19] | 0.7211(0.0198) | 0.6327(0.0149) | 0.5434(0.0430) | 0.5757(0.0219) | 0.4761(0.0162) | 0.3898(0.0436) | 0.8885(0.0080) | 0.8531(0.0059) | 0.8182(0.0163) |
| Cerberus [22] | **0.7827(0.0129)** | **0.7650(0.0147)** | **0.7123(0.0215)** | **0.6522(0.0149)** | **0.6293(0.0183)** | **0.5649(0.0227)** | 0.9131(0.0052) | **0.9060(0.0059)** | **0.8850(0.0086)** |
| msY-Net [7] | 0.7274(0.0191) | 0.6683(0.0150) | 0.5559(0.0455) | 0.5829(0.0227) | 0.5143(0.0166) | 0.3969(0.0432) | 0.8910(0.0076) | 0.8673(0.0060) | 0.8224(0.0181) |
| HookNet [24] | 0.7053(0.0145) | 0.6494(0.0133) | 0.5363(0.0601) | 0.5573(0.0171) | 0.4925(0.0135) | 0.3825(0.0553) | 0.8822(0.0058) | 0.8598(0.0053) | 0.8152(0.0242) |
| Slf-Hist [12] | 0.7553(0.0188) | 0.7465(0.0134) | 0.6507(0.0244) | 0.6157(0.0225) | 0.6052(0.0185) | 0.4957(0.0248) | 0.9022(0.0075) | 0.8986(0.0054) | 0.8603(0.0098) |
| DSMIL [34] | 0.7554(0.0125) | 0.7153(0.0280) | 0.6079(0.0528) | 0.6156(0.0149) | 0.5672(0.0314) | 0.4509(0.0541) | 0.9022(0.0050) | 0.8862(0.0112) | 0.8432(0.0211) |
| SSL_CR_Histo [40] | 0.7406(0.0186) | 0.6765(0.0155) | 0.4111(0.0428) | 0.5981(0.0221) | 0.5234(0.0174) | 0.2740(0.0340) | 0.8963(0.0075) | 0.8707(0.0062) | 0.7740(0.0259) |
| HookNet + MSF-WSI (ours) + Canelyon16 | 0.7769(0.0094) | *0.7493(0.0102)* | *0.6554(0.0170)* | *0.6440(0.0110)* | *0.6076(0.0120)* | *0.4989(0.0183)* | *0.9108(0.0038)* | *0.8998(0.0041)* | *0.8623(0.0068)* |
| SimSiam [31] | *0.7623(0.0114)* | 0.7326(0.0214) | 0.6365(0.0312) | 0.6245(0.0146) | 0.5873(0.0253) | 0.4778(0.0339) | 0.9049(0.0046) | 0.8931(0.0086) | 0.8550(0.0119) |
| HookNet + MSF-WSI (ours) + BCSS | **0.7827(0.0121)** | 0.7572(0.0095) | 0.6876(0.0262) | 0.6470(0.0114) | 0.6174(0.0127) | 0.5354(0.0275) | **0.9136(0.0039)** | 0.9031(0.0037) | 0.8751(0.0105) |

**Table 6**
Ablation experiments on our CTFM and DSL. The best result is bold.

| Method | DSC |
|---|---|
| Ours (w/o CTFM and DSL, random-init) | 0.7471 |
| Ours (w/o CTFM and DSL, ImageNet-init) | 0.7768 |
| Ours (w/o CTFM and DSL, simsiam-init) | 0.7633 |
| Ours (w/o DSL, only jigsaw) | 0.7753 |
| Ours (w/o DSL, only masking) | 0.7773 |
| Ours (w/o DSL, only CTFM) | 0.7881 |
| Ours (w/o CTFM, only DSL) | 0.7896 |
| Ours (with CTFM and DSL) | **0.8072** |

**Table 7**
Ablation study on SSL pre-training strategy. The best result is bold.

| Method | DSC |
|---|---|
| SimCLR [29] | 0.7614 |
| MoCo v2 [8] | 0.7642 |
| SimSiam [31] | **0.7671** |
| BYOL [9] | 0.7661 |

Within the context of the BCSS dataset, Cerberus consistently achieved the top position across all scenarios, achieving the highest DSC, IoU, and Acc scores, except for the Acc score when utilising 50% of the training data, where it had the second-best result. Our MSF-WSI model, pre-trained on the BCSS dataset, obtained the second-best results in most cases. Notably, our MSF-WSI model achieved the best DSC and Acc scores when using 50% of the training data. The method that had the third-best performance across all metrics and settings was our MSF-WSI model pre-trained on Camelyon16.

### 4.5. Ablation studies

We conducted all ablation studies on fold 1 of the BCSS dataset and used the same model configurations as with the fine-tuning experiments.

#### 4.5.1. Effectiveness of CTFM and DSL

Table 6 presents the result measuring the efficacy of the proposed model components. We initiated our evaluation by benchmarking the randomly-initialised model which yielded a DSC score of 0.7471. Notably, we observed that employing weights pre-trained solely through the SimSiam approach led to modest enhancements in model performance, resulting in an improvement of approximately 0.02, although this improvement was less pronounced compared to the utilisation of ImageNet-pre-trained weights (DSC score of 0.7768).

Furthermore, we investigated the impact of our CTFM, which achieved a DSC score of 0.7881 (0.0113 higher than the model pre-trained with ImageNet weights). We proceeded to evaluate the efficacy of individual components by independently applying the jigsaw task (yielding a DSC score of 0.7753) and the masking task (yielding a DSC score of 0.7773).

In addition, we validated the effectiveness of the DSL module independently from CTFM, and obtained a DSC score of 0.7896. Notably, the combination of these two pivotal model components yielded a promising increase of approximately 6% from the baseline model and an enhancement of 3% when compared to the ImageNet-pre-trained model.

#### 4.5.2. SSL strategy

Our evaluation included four SOTA SSL algorithms: SimCLR [29], MoCo v2 [8], SimSiam [31] and BYOL [9]. In this ablation analysis, instead of utilising our MSF-WSI approach, we independently applied each SSL algorithm to context and target branches. The objective was to highlight the inherent advantages arising solely from SSL pre-training. The results are presented in Table 7.

A noteworthy observation emerges from the outcomes: SSL algorithms that avoid the need for negative samples attained superior performance compared to their counterparts. Specifically, SimSiam exhibited better efficacy, attaining a DSC score of 0.7671, closely followed by BYOL with a DSC score of 0.7661.

#### 4.5.3. Robustness of model selection

To evaluate the robustness of our proposed algorithm, we conducted evaluations with varying model selections, the results are shown in Table 8. The pre-training process was the same as MSF-WSI but we reduced the number of epochs to 300 to save computation time. Our assessment contained not only a deeper ResNet-34 architecture [20], but

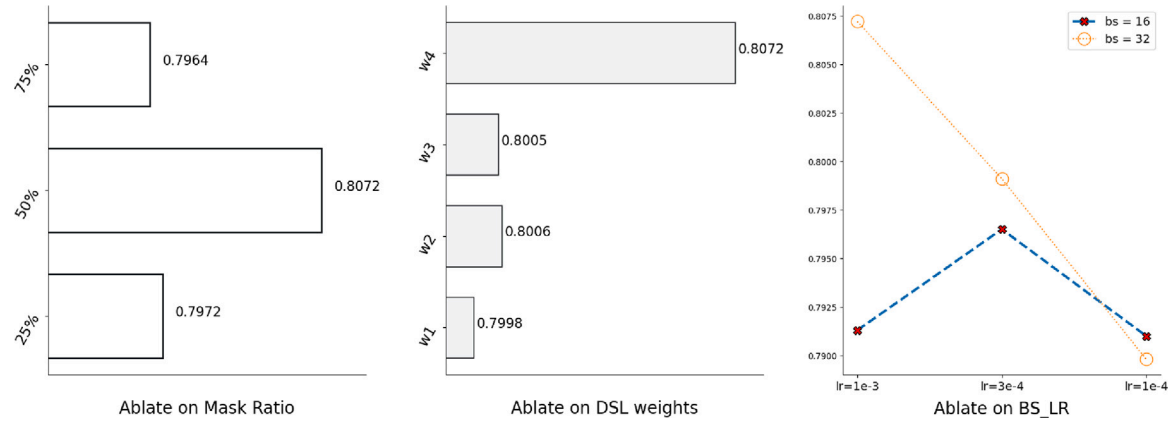


**Fig. 3.** Ablation study on hyperparameters, including random mask ratio of CTFM (25%, 50%, and 75 %), loss weights of DSL (w1 = [1.0, 1.0, 1.0, 1.0], w2 =[0.5, 0.7, 0.7, 1.0], w3 = [0.2, 0.5, 0.8, 1.0], and w4 = [0.1, 0.4, 0.7, 1.0]), and combinations of batch size (16 and 32) and learning rate (1e-3, 3e-4, 1e-4). Their corresponding experiment results are shown from left to right.

**Table 8**
Ablation study on model backbone. The best result is bold.

| Method | DSC | ImageNet Acc |
|---|---|---|
| ResNet-18 | 0.7851 | 69.758 |
| ResNet-34 | 0.7880 | 73.314 |
| RegNetY-008 [41] | 0.8004 | 76.314 |
| EfficientNet-B0 [42] | **0.8082** | **77.700** |
| SegFormer-B0 [43] | 0.7779 | N/A |

**Table 9**
Ablation study on model speed and memory via the total number of model parameters with the unit of Million (M), the model throughput with the unit of seconds per batch (sec./batch), and peak GPU memory with the unit of Gigabytes (GB).

| Method | Params (M) | Throughput(sec./batch) | Peak memory (GB) |
|---|---|---|---|
| Slf-Hist | 12.49 | 201.33 | 12.62 |
| DSMIL | 24.98 | 406.80 | 12.62 |
| SSL_CR_Histo | 11.93 | 212.34 | 12.58 |
| SimSiam | 29.70 | 417.39 | 14.02 |
| MSF-WSI | 123.55 | 419.18 | 14.02 |

also contained contemporary SOTA image recognition models, including RegNet [41], EfficientNet [42], and SegFormer [43] (a transformer-based semantic segmentation model). Subsequently, we substituted pre-trained model backbones with the encoder part of HookNet.

Upon examining the outcomes detailed in Table 8, it becomes evident that the trend in DSC scores across different backbone models remained in alignment with their respective performances in the context of the ImageNet classification task. Remarkably, EfficientNet-B0 attained the highest result of 0.8082, which marked an improvement of approximately 0.02 in comparison to the performance of ResNet-18 (0.7851). It is noteworthy that SegFormer-B0 obtained the lowest performance, achieving a DSC score of 0.7779.

#### 4.5.4. Empirical evaluation of hyperparameters

We conducted an empirical study to select the hyperparameters for the model's pre-training performance. The results are shown in Fig. 3. We experimented with the random mask ratio of CTFM (25%, 50%, and 75%). Masking half of the target features obtained the best pre-training performance, yielding DSC of 0.8072 compared with 0.7972 from 25% mask ratio and 0.7964 from 75% mask ratio. We also evaluated loss weights of DSL (w1 = [1.0, 1.0, 1.0, 1.0], w2 = [0.5, 0.7, 0.7, 1.0], w3 = [0.2, 0.5, 0.8, 1.0], and w4 = [0.1, 0.4, 0.7, 1.0]), and observed the w4 achieved the best DSC of 0.8072 than the other three sets of loss weights. Additionally, we tested combinations of batch size (16 and 32) and learning rate (1e-3, 3e-4, 1e-4). We identified that using a batch size of 32 and a learning rate of 1e-3 gave the best result.

#### 4.5.5. Comparisons of memory and speed

We compared our MSF-WSI with other SSL pre-training algorithms in terms of GPU memory and speed, via the total number of model parameters with the unit of Million (M), the model throughput with the unit of seconds per batch (sec./batch), and peak GPU memory with the unit of Gigabytes (GB). We compared four SSL algorithms: Slf-Hist, DSMIL, SSL_CR_Histo, and SimSiam. All algorithms were run on a machine with Intel(R) Core(TM) i9-10900K CPU and a single Nvidia GeForce RTX 3090 24G. The experiment configurations were: batch size was maximum multiple of 2 fitting into the GPU, number of worker was 1, and using mixed precision with PyTorch version of 1.13.

Table 9 shows the results of GPU memory usage and speed among the comparison methods. Slf-Hist and SSL_CR_Histo had similar number of parameters which is around 12M along with similar peak memory usage of 13 GB. DSMIL and SimSiam run nearly half slower (around 400 s./batch) than Slf-Hist and SSL_CR_Histo (around 200 s./batch). SimSiam requires more GPU memory (14.02 GB) than DSMIL (12.62 GB). Our MSF-WSI contains largest model parameters (123.55M) due to additional MLPs in DSL. The peak memory and throughput are similar to SimSiam which is 14.02 GB and 419.18 s./batch, respectively.

### 4.6. Visualisation

We executed a qualitative assessment of MSF-WSI by visualising the top-4 performing methods including MSF-WSI-BCSS, MSF-WSI-C16, Cerberus, and Slf-Hist on the BCSS dataset. Fig. 4 illustrates the visualisations where each mask is colour-coded: blue represents Tumour, yellow for Stroma, green for Lymphocytic Infiltrate, purple for Necrosis, and orange for Other regions.

In the initial row, MSF-WSI exhibited fewer false positives compared to the other methods. For instance, Cerberus yielded more false positive predictions for the OTR class, while Slf-Hist produced more false positive predictions for the NEC class. In the second sample, our approach generated a more precise segmentation mask for the NEC class, as evident in the lower right corner. In the third example, MSF-WSI demonstrated superior performance in segmenting the TUM class in contrast to the other methods.

## 5. Discussion

The primary findings are as follows: (i) In contrast to conventional SSL methods, our CTFM and DSL enabled multi-resolution learning during pre-training, effectively leveraging complementary multi-resolution

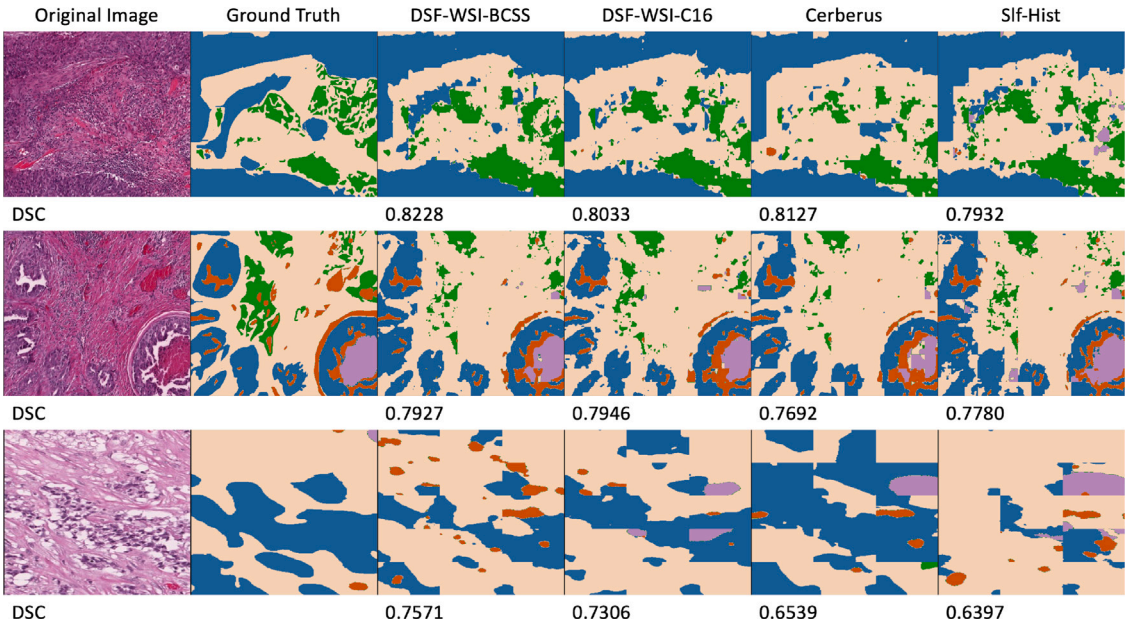


**Fig. 4.** Visualisation of semantic segmentation results in BCSS. Blue is for Tumour, yellow is for Stroma, green is for Lymphocytic Infiltrate, purple is for Necrosis, and orange is for Others. Corresponding DSC are shown under each of predictions. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

features for subsequent segmentation fine-tuning; (ii) Our MSF-WSI pre-training strategy consistently enhanced histopathology image segmentation, particularly in cases of testing, where both pre-training and fine-tuning occur on the same datasets and; (iii) In scenarios with constrained labelled training data, SSL pre-training aided in model convergence, most notably when the distribution of the data in pre-training datasets closely aligns with that of the fine-tuning datasets.

### *5.1. Comparing to existing methods on different datasets*

Our MSF-WSI pre-training consistently outperformed all other methods on both the PAIP2019 and BCSS datasets, as shown in Table 1. We observed that pre-training on histopathology datasets yielded more improvements than using ImageNet pre-training. For instance, Cerberus achieved substantial enhancements, improving the ImageNet-initialised U-Net with a higher DSC of 0.6% in PAIP2019 and 3.6% in BCSS. The gap between PAIP2019 and BCSS could be attributed to differences in data distribution; notably, TCGA, one of Cerberus's pre-training datasets, serves as the parent set of BCSS. When compared to U-Net, the baseline model HookNet exhibited comparable performance when trained from scratch, while the more advanced architecture msY-Net achieved better results when initialised randomly. This indicates that integrating context features from the encoder path is superior to doing so from the decoder path.

In contrast, SSL pre-training proved effective in enhancing model performance compared to the baseline HookNet. For instance, SSL_CR_Histo led to improved model performance in the PAIP2019 dataset, and DSMIL contributed to model improvement for both the PAIP2019 and BCSS datasets. A trend emerged where the SSL pre-training seemed to offer more significant benefits in the BCSS dataset than in the PAIP2019 dataset. This could be attributed to the fact that PAIP2019 with its 3 target classes, is comparatively simpler than the BCSS dataset with 5 target classes. Another factor is that Camelyon16 pre-training dataset comprises of breast WSIs, similar to the BCSS which is also a breast dataset, while PAIP2019 contains liver WSIs. This performance discrepancy could be mitigated by either pre-training on diverse external datasets like Slf-Hist or by pre-training on the same dataset like SimSiam.

While existing SSL histopathology methods used multi-resolution WSIs independently, they struggled to achieve competitive performance compared to Cerberus. In contrast, our MSF-WSI pre-training achieved superior results, whether by transferring from Camelyon16 or pre-training on the same dataset. Specifically, MSF-WSI pre-trained on Camelyon16 outperformed other methods, enhancing baseline HookNet with a 1.7% higher DSC, 2.2% higher IoU, and 1.2% higher Acc on the PAIP2019 dataset, and 4% higher DSC, 6.1% higher IoU, and 1.6% higher Acc on the BCSS dataset. Furthermore, through pre-training on the same dataset, our approach achieved additional performance gains, surpassing the best existing method msY-Net by 1.3% higher DSC, 1.7% higher IoU, and 0.9% higher Acc on the PAIP2019 dataset, and Cerberus by 0.9% higher DSC, 1.1% higher IoU, and 0.4% higher Acc on the BCSS dataset. These experimental results underscored the significance of learning the correlations between resolutions during SSL pre-training, which is crucial for multi-resolution models to effectively utilise complementary information between multi-resolution WSIs for improved segmentation performance.

### *5.2. Analysis of semi-supervised performance*

Our method consistently achieved superior results across all metrics on the PAIP2019 dataset, while maintaining a top-three position on the BCSS dataset, as indicated in Tables 4 and 5. From these results, it is evident that randomly-initialised and ImageNet-initialised models experienced notable performance degradation as the available training data diminished. Specifically, there was an average drop of 10% in DSC on the PAIP2019 dataset and a 20% DSC drop on the BCSS dataset when only 1% of the data was utilised for training.

In alignment with the main fine-tuning experiments, the transfer from the distinct WSI dataset Camelyon16 (breast cancer) for the PAIP2019 dataset (liver cancer) did not outperform pre-training on the same dataset. Nevertheless, our MSF-WSI pre-trained on Camelyon16 exhibited the best performance among external testing methods, compared to the baseline HookNet, it improved DSC by 1.4% when using 50% of the data and by 2% when using 10% of the data. This suggests that pre-training on the same dataset is more efficient than pre-training on a larger but different dataset. Furthermore, when we pre-trained MSF-WSI on the PAIP2019 training set, our method enhanced HookNet with a higher DSC of 2.5% using 50% of the data, a higher DSC of 4% using 10% of the data, and a higher DSC of 6.4% using only 1% of

the data. Notably, after pre-training with our MSF-WSI, a mere 10% of the training data proved sufficient to surpass the performance of a random-initialised HookNet trained with 100% of the data.

For the BCSS dataset, Cerberus consistently achieved better results than other methods under all semi-supervised settings. Despite this, our MSF-WSI pre-trained on the BCSS dataset obtained the first position when the training data was 50%, and our method remained within the top three even in the 10% and 1% settings. The superior performance of Cerberus may be attributed to (1) ResNet-34's enhanced image feature representation, and (2) the model's robust representation owing to in-domain knowledge from the TCGA pre-training dataset and its exposure to a larger number of pre-training histopathology datasets.

In summary, our method demonstrated the effectiveness of learned representations, proving efficient for subsequent model fine-tuning even when working with partial datasets.

### 5.3. Analysis of model components

The contributions of CTFM and DSL components are presented in Table 6. It is noticeable that the combination of CTFM and DSL yielded the highest outcome, achieving 6% improvement over the baseline random initialisation. When compared to the randomly initialised model, CTFM improved the DSC by 4.1%, and DSL improved DSC by 4.2%. CTFM introduced a pretext task that demands the model to comprehend multi-resolution WSI features during SSL pre-training. Unlike the independent pretext tasks of jigsaw and masking that is common in SSL [25,26], our CTFM creates a more challenging task by generating complex samples involving multi-resolution WSIs. This, in turn, facilitates the multi-resolution model in learning how to learn complementary multi-resolution features, leading to an improvement of over 1% compared to both jigsaw and masking tasks. DSL strengthens the model's ability for representation learning by enabling SSL training in the early stages of the model. This is valuable since low-level image features can enhance segmentation performance. The standard SimSiam learning typically employs features solely from the last model layer, which often contains high-level, semantic-relevant representations for the entire input. However, in histopathology segmentation, low-level features like edges, colours, and curves, are also important. Additionally, since ROIs, such as tumour cells, might constitute a small portion of the image, later model layers may not adequately learn the ROI features. Enabling feature learning in the early stages of the model, as done in our DSL approach, resulted in a 2.3% increase in DSC when compared to SimSiam that focuses only on the last layer outputs.

In this study, we adopted the SimSiam method as our contrastive learning strategy due to its property of being negative-free, wherein the loss function maximises similarities between positive samples without minimising dissimilarities between negative samples. This characteristic offers two essential advantages: (1) it eliminates the need for a large batch size, reducing GPU memory demands, and (2) it relaxes the assumption for patch-based WSI methods, which require patches from the same WSI to be categorised as "positive" samples. This assumption was overlooked in previous works [12,13,34], leading to misclassification of patches from the same WSI (positive samples) as negatives if allocated to the same mini-batch. This misclassification hindered the model from receiving accurate updates from calculated losses. Our validation results in Table 7 demonstrated that negative-sample-free algorithms, such as BYOL and SimSiam, exhibited superior performance.

The assessments of backbone robustness, as presented in Table 8, highlighted the adaptability of our MSF-WSI to various backbone architectures. Generally, the segmentation performance aligns with the backbone's ImageNet classification accuracy, where EfficientNet-B0 demonstrated optimal performance and consistent superiority over other models. Notably, SegFormer exhibited the lowest performance despite being designed for the segmentation task. This might be attributed to the SegFormer decoder's decoupling from its optimally designed encoder. To ensure fair comparisons with previous SOTA methods, we used the standard ResNet-18 as the backbone in our main experiments.

Due to the advanced designs for processing multi-resolution inputs, we introduced extra hyperparameters for model pre-training which can sensitively affect the pre-training representation performance. Therefore, we conducted empirical evaluations on these hyperparameters and show results in Fig. 3. We found that using 50% masking ratio performed the best. We suggest that small masking ratio may reduce the difficulty of the pretext task and prevent the model from learning complementary multi-resolution features, whereas high masking ratio can remove essential details in target features. For the loss weights in DSL, we observed that using identical loss weights (w1) yielded the worst result. This is as expected since shallow layers learning of low-level features should have less effect than deep layers learning of high-level features [44]. Gradually increasing the weights from shallow to deep layers with an identical gap 0.3 (w4) seems to give a better result than other strategies. Additionally, we explored the effects of batch size and learning rate and found that larger batch size of 32 and learning rate of 1e-3 have advantages over other combinations.

In addition, we compared our algorithm with others in terms of memory and speed as shown in Table 9. Compared with algorithms (Slf-Hist and SSL_CR_Histo) which shared the same model for different patch resolutions, DSMIL and SimSiam doubled the number of parameters by adopting two unshared networks for multi-resolution inputs. Our MSF-WSI had more model parameters due to the introduce of MLPs in DSL. Nevertheless, the peak memory and throughput were similar among DSMIL, SimSiam and MSF-WSI. We conclude that the complexity of MSF-WSI arises from two aspects: (a) the data processing time is increased due to the on-the-fly generation and pre-processing of target patches, including shuffling and random masking during the pre-training; and (2) our model complexity is increased as distinct models are employed for multiple resolution patch feature extraction. Additionally, DSL requries additional predictors (3-layer MLP) and projectors (2-layer MLP) for the contrastive pre-training. It is noteworthy that after the pre-training, MLPs introduced in DSL and pre-processing of target features are removed and thus keep the efficiency for the later fine-tuning.

## 6. Conclusion

In this paper, we introduced a new SSL framework designed to exploit the multi-resolution information in WSIs for histopathology. Specifically, the learning of multi-resolution features was enabled by the proposed CTFM and masked jigsaw pretext task during the self-supervised pre-training. This compelled the model to understand the relationships between different WSI resolutions. Our experiment results showed that the proposed CTFM with masked jigsaw pretext task facilitate the learning of complementary histopathology information between multi-resolution WSIs, yielding superior representations compared to the original SimSiam and single masking or jigsaw pretext tasks. Furthermore, we found that maximising the feature similarities from early model layers in SSL pre-training is beneficial for learning low-level image features and contributing to the segmentation performance in histopathology. Our MSF-WSI was evaluated in three public datasets on breast and liver cancer segmentation tasks with internal and external testing settings, and it outperformed other SOTA SSL methods under different fractions of training data.

There are two limitations to our method that require further consideration for future works. One identified limitation is that the framework was evaluated with two fixed resolutions, and it can be generalised to use more WSI resolutions which provides richer hierarchy information than two-resolution inputs. Additionally, the model was pre-trained on a single dataset, potentially restricting the generalisability of learned representations. Addressing this limitation could involve combining diverse WSI datasets with various tissue types to create a large-scale

hybrid dataset, enabling the development of WSI-general-purpose pre-trained weights for more effective transfer learning. Moreover, future works could involve clinical validation assess the performance and reliability of our model in real clinical settings which is essential for its translation into clinical practice

## CRediT authorship contribution statement

**Hao Wang:** Writing – review & editing, Writing – original draft, Visualization, Validation, Project administration, Methodology, Formal analysis, Data curation, Conceptualization. **Euijoon Ahn:** Writing – review & editing, Writing – original draft, Supervision, Project administration, Formal analysis. **Jinman Kim:** Writing – review & editing, Supervision, Resources, Investigation, Funding acquisition.

## Declaration of competing interest

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

## Data availability

The authors do not have permission to share data.

## Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.patcog.2024.110621.

## References


[1] M. Ding, A. Qu, H. Zhong, Z. Lai, S. Xiao, P. He, An enhanced vision transformer with wavelet position embedding for histopathological image classification, Pattern Recognit. 140 (2023) 109532.

[2] P. Morales-Álvarez, A. Schmidt, J.M. Hernández-Lobato, R. Molina, Introducing instance label correlation in multiple instance learning. application to cancer detection on histopathological images, Pattern Recognit. 146 (2024) 110057.

[3] Y. Liu, K. Gadepalli, M. Norouzi, G.E. Dahl, T. Kohlberger, A. Boyko, S. Venugopalan, A. Timofeev, P.Q. Nelson, G.S. Corrado, J.D. Hipp, L.H. Peng, M.C. Stumpe, Detecting cancer metastases on gigapixel pathology images, 2017, ArXiv arXiv:1703.02442.

[4] L. Chan, M. Hosseini, C. Rowsell, K. Plataniotis, S. Damaskinos, HistoSegNet: Semantic segmentation of histological tissue type in whole slide images, in: 2019 IEEE/CVF International Conference on Computer Vision, ICCV, 2019, pp. 10661–10670, http://dx.doi.org/10.1109/ICCV.2019.01076.

[5] X. Zhang, X. Zhu, K. Tang, Y. Zhao, Z. Lu, Q. Feng, Ddtnet: A dense dual-task network for tumor-infiltrating lymphocyte detection and segmentation in histopathological images of breast cancer, Med. Image Anal. 78 (2022) 102415.

[6] F. Gu, N. Burlutskiy, M. Andersson, L.K. Wilén, Multi-resolution networks for semantic segmentation in whole slide images, in: D. Stoyanov, Z. Taylor, F. Ciompi, Y. Xu, A. Martel, L. Maier-Hein, N. Rajpoot, J. van der Laak, M. Veta, S. McKenna, D. Snead, E. Trucco, M.K. Garvin, X.J. Chen, H. Bogunovic (Eds.), Computational Pathology and Ophthalmic Medical Image Analysis, Springer International Publishing, Cham, 2018, pp. 11–18.

[7] R. Schmitz, F. Madesta, M. Nielsen, J. Krause, S. Steurer, R. Werner, T. Rösch, Multi-scale fully convolutional neural networks for histopathology image segmentation: From nuclear aberrations to the global tissue architecture, Med. Image Anal. 70 (2021) 101996, http://dx.doi.org/10.1016/j.media.2021.101996.

[8] K. He, H. Fan, Y. Wu, S. Xie, R.B. Girshick, Momentum contrast for unsupervised visual representation learning, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2019, pp. 9726–9735.

[9] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P.H. Richemond, E. Buchatskaya, C. Doersch, B.A. Pires, Z.D. Guo, M.G. Azar, B. Piot, K. Kavukcuoglu, R. Munos, M. Valko, Bootstrap your own latent a new approach to self-supervised learning, in: Proceedings of the 34th International Conference on Neural Information Processing Systems, NIPS '20, Curran Associates Inc., Red Hook, NY, USA, 2020.

[10] H. Wang, E. Ahn, J. Kim, Self-supervised representation learning framework for remote physiological measurement using spatiotemporal augmentation loss, Proc. AAAI Conf. Artif. Intell. 36 (2) (2022) 2431–2439, http://dx.doi.org/10.1609/aaai.v36i2.20143, ISSN: 2374-3468, 2159-5399.

[11] H. Wang, E. Ahn, L. Bi, J. Kim, Self-supervised multi-modality learning for multi-label skin lesion classification, 2023, arXiv:2310.18583.

[12] O. Ciga, T. Xu, A.L. Martel, Self supervised contrastive learning for digital histopathology, Mach. Learn. Appl. 7 (2022) 100198, http://dx.doi.org/10.1016/j.mlwa.2021.100198.

[13] N.A. Koohbanani, B. Unnikrishnan, S.A. Khurram, P. Krishnaswamy, N.M. Rajpoot, Self-path: Self-supervision for classification of pathology images with limited annotations, IEEE Trans. Med. Imaging 40 (2020) 2845–2856.

[14] M.K.K. Niazi, A.V. Parwani, M.N. Gurcan, Digital pathology and artificial intelligence, Lancet Oncol. 20 (5) (2019) e253–e261, http://dx.doi.org/10.1016/S1470-2045(19)30154-8.

[15] B. Gecer, S. Aksoy, E. Mercan, L.G. Shapiro, D.L. Weaver, J.G. Elmore, Detection and classification of cancer in whole slide breast histopathology images using deep convolutional networks, Pattern Recognit. 84 (2018) 345–356.

[16] L. Hou, V. Nguyen, A.B. Kanevsky, D. Samaras, T.M. Kurc, T. Zhao, R.R. Gupta, Y. Gao, W. Chen, D. Foran, et al., Sparse autoencoder for unsupervised nucleus detection and representation in histopathology images, Pattern Recognit. 86 (2019) 188–200.

[17] H. Chen, X. Qi, L. Yu, Q. Dou, J. Qin, P.-A. Heng, DCAN: Deep contour-aware networks for object instance segmentation from histology images, Med. Image Anal. 36 (2017) 135–146, http://dx.doi.org/10.1016/j.media.2016.11.004.

[18] Y.-R. Van Eycke, C. Balsat, L. Verset, O. Debeir, I. Salmon, C. Decaestecker, Segmentation of glandular epithelium in colorectal tumours to automatically compartmentalise IHC biomarker quantification: A deep learning approach, Med. Image Anal. 49 (2018) 35–45, http://dx.doi.org/10.1016/j.media.2018.07.004.

[19] 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.

[20] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2015, pp. 770–778.

[21] S. Graham, Q.D. Vu, S.E.A. Raza, A. Azam, Y.-W. Tsang, J.T. Kwak, N.M. Rajpoot, Hover-Net: Simultaneous segmentation and classification of nuclei in multi-tissue histology images, Med. Image Anal. 58 (2018) 101563.

[22] S. Graham, Q.D. Vu, M. Jahanifar, S.E.A. Raza, F. Minhas, D. Snead, N. Rajpoot, One model is all you need: Multi-task learning enables simultaneous histology image segmentation and classification, Med. Image Anal. 83 (2023) 102685, http://dx.doi.org/10.1016/j.media.2022.102685.

[23] G. Nir, S. Hor, D. Karimi, L. Fazli, B.F. Skinnider, P. Tavassoli, D. Turbin, C.F. Villamil, G. Wang, R.S. Wilson, K.A. Iczkowski, M.S. Lucia, P.C. Black, P. Abolmaesumi, S.L. Goldenberg, S.E. Salcudean, Automatic grading of prostate cancer in digitized histopathology images: Learning from multiple experts, Med. Image Anal. 50 (2018) 167–180.

[24] M. van Rijthoven, M.C.A. Balkenhol, K. Silina, J. van der Laak, F. Ciompi, HookNet: multi-resolution convolutional neural networks for semantic segmentation in histopathology whole-slide images, Med. Image Anal. 68 (2020) 101890.

[25] C. Doersch, A.K. Gupta, A.A. Efros, Unsupervised visual representation learning by context prediction, in: 2015 IEEE International Conference on Computer Vision, ICCV, 2015, pp. 1422–1430.

[26] M. Noroozi, P. Favaro, Unsupervised learning of visual representations by solving Jigsaw puzzles, in: European Conference on Computer Vision, 2016.

[27] R. Zhang, P. Isola, A.A. Efros, Colorful image colorization, in: European Conference on Computer Vision, 2016.

[28] S. Gidaris, P. Singh, N. Komodakis, Unsupervised representation learning by predicting image rotations, in: International Conference on Learning Representations, 2018, URL: https://openreview.net/forum?id=S1v4N2l0-.

[29] T. Chen, S. Kornblith, M. Norouzi, G. Hinton, A simple framework for contrastive learning of visual representations, in: Proceedings of the 37th International Conference on Machine Learning, ICML '20, JMLR.org, 2020.

[30] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, A. Joulin, Unsupervised learning of visual features by contrasting cluster assignments, in: Proceedings of the 34th International Conference on Neural Information Processing Systems, NIPS '20, Curran Associates Inc., Red Hook, NY, USA, 2020.

[31] X. Chen, K. He, Exploring simple siamese representation learning, in: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2020, pp. 15745–15753.

[32] A. van den Oord, Y. Li, O. Vinyals, Representation learning with contrastive predictive coding, Cornell University Library, Ithaca, 2019, arXiv.org.

[33] S. Azizi, B. Mustafa, F. Ryan, Z. Beaver, J. von Freyberg, J. Deaton, A. Loh, A. Karthikesalingam, S. Kornblith, T. Chen, V. Natarajan, M. Norouzi, Big self-supervised models advance medical image classification, in: 2021 IEEE/CVF International Conference on Computer Vision, ICCV, 2021, pp. 3458–3468.

[34] B. Li, Y. Li, K.W. Eliceiri, Dual-stream multiple instance learning network for whole slide image classification with self-supervised contrastive learning, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, vol. 2021, United States, 2021, pp. 14318–14328.

[35] X. Wang, S. Yang, J. Zhang, M. Wang, J. Zhang, W. Yang, J. Huang, X. Han, Transformer-based unsupervised contrastive learning for histopathological image classification, Med. Image Anal. 81 (2022) 102559.

[36] B. Ehteshami Bejnordi, M. Veta, P. Johannes van Diest, B. van Ginneken, N. Karssemeijer, G. Litjens, J.A.W.M. van der Laak, the CAMELYON16 Consortium, Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer, JAMA 318 (22) (2017) 2199–2210, http://dx.doi.org/10.1001/jama.2017.14585.

[37] Y.J. Kim, H. Jang, K. Lee, S. Park, S.-G. Min, C. Hong, J.H. Park, K. Lee, J. Kim, W. Hong, H. Jung, Y. Liu, H. Rajkumar, M. Khened, G. Krishnamurthi, S. Yang, X. Wang, C.H. Han, J. Choi, PAIP 2019: Liver cancer segmentation challenge, Med. Image Anal. 67 (2020) 101854.

[38] M. Amgad, H. Elfandy, H. Hussein, L.A. Atteya, M.A.T. Elsebaie, L.S.A. Elnasr, R.A. Sakr, H.S.E. Salem, A.F. Ismail, A.M. Saad, J. Ahmed, M.A.T. Elsebaie, M. Rahman, I.A. Ruhban, N.M. Elgazar, Y. Alagha, M.H. Osman, A.M. Alhusseiny, M.M. Khalaf, A.-A.F. Younes, A. Abdulkarim, D.M. Younes, A.M. Gadallah, A.M. Elkashash, S.Y. Fala, B.M. Zaki, J.D. Beezley, D.R. Chittajallu, D. Manthey, D.A. Gutman, L.A.D. Cooper, Structured crowdsourcing enables convolutional segmentation of histology images, Bioinformatics 35 (2019) 3461–3467.

[39] A. Prat Aparicio, Comprehensive molecular portraits of human breast tumours, Nature 490 (7418) (2012) 61–70.

[40] C.L. Srinidhi, S.W. Kim, F.-D. Chen, A.L. Martel, Self-supervised driven consistency training for annotation efficient histopathology image analysis, Med. Image Anal. 75 (2022) 102256, http://dx.doi.org/10.1016/j.media.2021.102256.

[41] I. Radosavovic, R.P. Kosaraju, R.B. Girshick, K. He, P. Dollár, Designing network design spaces, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2020, pp. 10425–10433.

[42] M. Tan, Q. Le, EfficientNet: Rethinking model scaling for convolutional neural networks, in: K. Chaudhuri, R. Salakhutdinov (Eds.), Proceedings of the 36th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 97, PMLR, 2019, pp. 6105–6114.

[43] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J.M. Álvarez, P. Luo, SegFormer: Simple and efficient design for semantic segmentation with transformers, in: Neural Information Processing Systems, 2021.

[44] A. Kaku, S. Upadhya, N. Razavian, Intermediate layers matter in momentum contrastive self supervised learning, Adv. Neural Inf. Process. Syst. 34 (2021) 24063–24074.

**Hao Wang** is a Ph.D. candidate at the Biomedical Data Analysis and Visualization (BDAV) Lab, School of Computer Science, Faculty of Engineering, The University of Sydney, Australia. His research interests include deep learning, video analysis, medical image classification, segmentation, and analysis.

**Euijoon Ahn** is a Lecturer at the College of Science and Engineering, James Cook University, Cairns, Australia. He obtained the Ph.D. degree in Computer Science from The University of Sydney in 2020. His research interest is computer vision, focusing on unsupervised and self-supervised deep learning models for biomedical image analysis.

**Jinman Kim** is a Professor of computer science at the University of Sydney. Prof Kim received his Ph.D. degrees in computer science from the University of Sydney in 2006. He worked as a Senior Lecturer (2013), A/Prof (2016), and Prof (2022) at the University of Sydney.