DiverseNet: Decision Diversified Semi-supervised Semantic Segmentation Networks for Remote Sensing Imagery

Wanli Ma, Oktay Karakuş, Paul L. Rosin
School of Computer Science and Informatics, Cardiff University, Cardiff, U.K.

***Abstract*—Semi-supervised learning (SSL) aims to help reduce the cost of the manual labelling process by leveraging a substantial pool of unlabelled data alongside a limited set of labelled data during the training phase. Since pixel-level manual labelling in large-scale remote sensing imagery is expensive and time-consuming, semi-supervised learning has become a widely used solution to deal with this. However, the majority of existing SSL frameworks, especially various teacher-student frameworks, are too bulky to run efficiently on a GPU with limited memory. There is still a lack of lightweight SSL frameworks and efficient perturbation methods to promote the diversity of training samples and enhance the precision of pseudo labels during training. In order to fill this gap, we proposed a simple, lightweight, and efficient SSL architecture named *DiverseHead*, which promotes the utilisation of multiple decision heads instead of multiple whole networks. Another limitation of most existing SSL frameworks is the insufficient diversity of pseudo labels, as they rely on the same network architecture and fail to explore different structures for generating pseudo labels. To solve this issue, we propose *DiverseModel* to explore and analyse different networks in parallel for SSL to increase the diversity of pseudo labels. The two proposed methods, namely *DiverseHead* and *DiverseModel*, both achieve competitive semantic segmentation performance in four widely used remote sensing imagery datasets compared to state-of-the-art semi-supervised learning methods. Meanwhile, the proposed lightweight DiverseHead architecture can be easily applied to various state-of-the-art SSL methods while further improving their performance. The code is available at Here.**

***Index Terms*—Semi-supervised Learning, Semantic Segmentation, Land over classification, Building Detection, Roadnet Detection**

I. Introduction

Supervised deep learning has become the dominant technique in computer vision during the last decade. Building on its success in computer vision, many remote sensing applications, such as land cover classification, change detection and object detection have seen significant improvements as similar tasks [1], [2], [3], [4], [5]. Nevertheless, supervised learning necessitates a substantial and meticulously labelled dataset. In the case of extensive remote sensing data, such as satellite imagery and drone-captured images in complex terrains, acquiring pixel-wise expert annotations is a time-consuming, labour-intensive, and costly process. While the field of computer vision provides numerous well-annotated datasets, transferring deep learning models trained on these datasets to the remote sensing domain is a formidable challenge. This is mainly due to the substantial differences between typical computer vision images and remote sensing data, such as hyperspectral and synthetic aperture radar (SAR) imagery, which often exhibit unconventional and non-intuitive characteristics. To address the issue of limited access to massive labelled datasets, SSL offers a viable solution by using a small amount of labelled data while leveraging the abundance of unlabelled data [6], [7], [8]. This is because large volumes of unlabelled imagery generally can be easily and freely accessed from open-access remote sensing data sources.

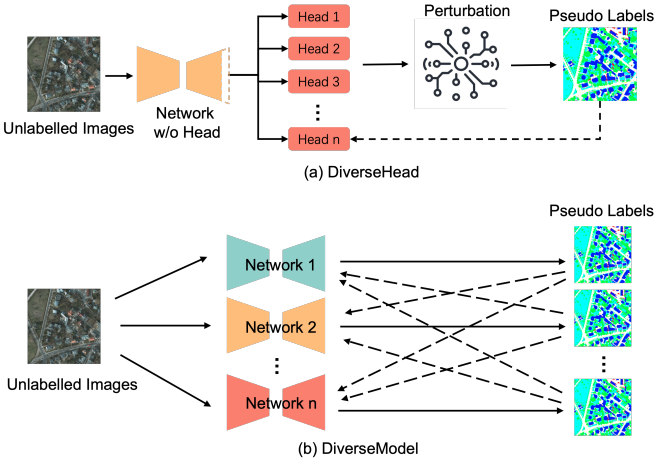

Fig. 1. Two kinds of pseudo label generation and usage methods for SSL based on (a) DiverseHead with multiple heads and (b) DiverseModel with multiple models. '⟶' means data stream, '⇢' means loss supervision. The *'dynamic freezing'* and *'dropout'* are used as perturbation methods in the DiverseHead framework.

SSL has become a widely used technique in computer vision to reduce the labour-intensive and costly annotation process for various applications such as image classification [9], [10], [11] and segmentation [12], [13], [14], whilst leading to competitive performance. Specifically, taking advantage of "pseudo" labels generated by the prediction of unlabelled data has become a mainstream class of SSL methods. Thus, the quality of pseudo labels becomes a crucial factor in determining the training effectiveness of the models. With the success of SSL in computer vision, many SSL semantic segmentation approaches for remote sensing imagery have been explored, especially during the last decade [15], [16], [17]. Specifically, based on consistency learning, U-MCL [18] proposes an uncertainty-aware technique integrating masks for SSL semantic segmentation of remote sensing imagery, and

[19] promotes the importance of bias-correction .

In the SSL literature, enhancing the accuracy and diversity of pseudo labels has become a major challenge and a central research focus. This is due to the fact that pseudo label accuracy is often inadequate, especially when the labelled training data is limited or incomplete. On the other hand, enhancing the diversity of pseudo labels becomes another key research focus in SSL to improve model robustness, particularly in the context of consistency regularisation. [20].

In order to improve the quality of pseudo labels and empower networks to harness the potential of unlabelled data, a notable technique known as consistency regularisation has emerged as a widely adopted method for SSL. [20], [21]. Specifically, consistency regularisation methods are built up on the theory of assumption of smoothness, which suggests that if two points lie in a high-density region of feature space and are close to each other, their corresponding labels should be the same or consistent [22], [23], [20]. In practice, consistency regularisation SSL executes this assumption by forcing networks to produce consistent predictions for modified versions of unlabelled data or their features using various perturbation techniques.

The strategies for the aforementioned perturbations can be categorised into three groups, namely *input, feature,* and *network* perturbations. Input perturbation involves modifying or altering input data, enabling SSL approaches to enforce consistency in predictions for these altered inputs. The widely used input perturbation method is adding artificial noise to input images. However, it might lead to incorrect or noisy pseudo labels for unlabelled examples and negatively impact the training efficiency by providing incorrect guidance to the model when training with these pseudo labels. Apart from input perturbation, feature perturbation methods introduce noise to both low- and high-level features. These perturbed features are then fed into multiple decoders to generate multiple outputs, followed by the enforcement of consistency among the outputs obtained from different decoders. However, similar to input perturbation, feature perturbation potentially generates inaccurate representations due to introducing noise that fails to accurately capture the underlying patterns in the original data. This may result in incorrect or noisy pseudo-labels for unlabelled examples, ultimately hindering the learning process. Lastly, network perturbation uses multiple networks to promote diversity of predictions [24]. Unlike the previous two perturbation methods, network perturbation techniques introduce perturbations in a more structured manner, generated by the model itself instead of artificial noise. However, they generally require significantly more computational resources due to the greater number of complete networks (or their internal stages) involved.

To further justify our design, our network perturbation strategies (dynamic freezing and dropout) introduce structured, model-driven diversity without adding external noise. Unlike input and feature perturbations, which risk corrupting pseudo-labels or internal representations due to artificial noise injection, our approach preserves semantic integrity while enhancing prediction diversity. Furthermore, DiverseHead applies lightweight perturbations to the model head, achieving a balance between diversity and efficiency without the heavy computational cost of maintaining multiple full models.

The previous overview emphasises that while earlier perturbation techniques in SSL provide certain benefits, they also have inherent limitations, including inefficiencies in generating high-quality pseudo labels and a high computational cost. Facing the challenges posed by perturbation-based SSL and its complexity, it becomes imperative to explore more efficient and lightweight approaches. This work proposes two perturbation-based semi-supervised network architectures, coined as *DiverseNet*, which consist of multiple head (DiverseHead) and multiple model (DiverseModel) based SSL frameworks for various semantic segmentation applications of remote sensing imagery. A brief demonstration of the proposed lightweight SSL framework called *DiverseHead* is shown in Figure 1-(a). In addition, we also further analysed a previously proposed cross-network based SSL structure called *DiverseModel* [25], as shown in Figure 1-(b) for scenarios equipped with high-memory computational resources. Specifically, the contributions of this work are as follows:

1) We introduce DiverseHead, a simple, lightweight, and efficient SSL framework which employs multiple decision heads within a single network. This structure is inspired by bagging (also called bootstrap aggregating), which helps enhance pseudo label quality by integrating perturbed parameters and features within the network architecture.
2) To introduce perturbation for diversifying decisions, we incorporate two key techniques, *dynamic freezing* and *dropout*, into the DiverseHead architecture, aiming to diversify the network's parameters and high-level features, respectively. The proposed perturbation strategies, incorporating multiple heads, are readily applicable to a variety of state-of-the-art SSL methods and can further enhance their performance.
3) We propose a dual voting mechanism, Mean Voting and Max Voting, to aggregate multihead predictions and produce high-fidelity pseudo labels for DiverseHead. This mechanism leverages both collective consensus and individual confidence to further enhance pseudo-label robustness during training.
4) We provide a more detailed comparison study for a previously proposed architecture DiverseModel [25] on various semantic segmentation datasets in this paper. Also, we use Grad-CAM [26] to verify the observation that different networks exhibit varied attention to the same input.

The rest of the paper is organised as follows: Section II discusses the related work on semi-supervised semantic segmentation in remote sensing and some basic knowledge on ensemble machine learning whilst in Section III and IV, the proposed algorithms DiverseHead and DiverseModel are presented. Section V describes the utilised segmentation dataset of remote sensing imagery. The experimental setting along with both the qualitative and quantitative analyses of the results, are presented in Section VI. Section VII concludes the paper with a summary.

II. Related Work

Semantic segmentation is rapidly developing in remote sensing with the success of deep learning in computer vision. Due to the strong task similarity, semantic segmentation techniques are used for various remote sensing applications, such as land cover classification/mapping [27], building change detection [28], road extraction [29], and marine debris detection [30], [31]. Specifically, Fully Convolutional Networks (FCNs) [32] have made a considerable contribution to various segmentation tasks either in remote sensing or computer vision. Following the FCNs' success, SegNet [33] and UNet [34] adopt a symmetrical encoder-decoder structure with skip connections, leveraging multi-stage features within the encoder. Alternatively, PSPNet [35] introduces a pyramid pooling structure that helps provide a global contextual understanding for pixel-level scene parsing. The DeepLab architecture [36] introduces atrous convolution and atrous spatial pyramid pooling (ASPP), allowing the network to adjust the spatial receptive field of convolution kernels by using different dilation rates. Then, DeepLab was extended to DeepLabv3+ [37] with an improved encoder-decoder structure, which is helpful to refine segmentation results, especially around object boundaries [38], [39]. Recently, GLOTS [40] was proposed for semantic segmentation of remote sensing images, aiming to acquire consistent feature representations by leveraging transformers in both the encoder and decoder. DeepLabv3+ is one of the most widely used networks in the literature for semi-supervised learning segmentation in the computer vision area.

Semi-supervised learning aims to alleviate the need for expensive annotation work by making use of both labelled and unlabelled data. Self-training [41] (also known as pseudo labelling) represents one of the primitive SSL strategies for both classifications [42] and segmentation [43]. It generates pseudo-labels using model predictions for unlabelled data, which are then utilised to retrain the model. Another widely developed SSL approach called consistency regularisation [20] is to force networks to give consistent predictions for unlabelled inputs that undergo diverse perturbations. In the context of the remote sensing field, Lu et al. [44] propose a weak-to-strong consistency learning for semi-supervised semantic segmentation. Building on the weak-to-strong consistency learning, Lv et al. [45] further explore the efficient exploitation of labelled and unlabelled images. Additionally, MIMSeg [46] integrates weak-to-strong consistency learning with masked image modelling, and DWL [47] combines it with a decoupled weighting learning framework for semi-supervised semantic segmentation of remote sensing imagery.

In computer vision, CCT [48] employs an encoder-decoder architecture with multiple auxiliary decoders. These decoders introduce diversity in the output by feature perturbations specific to each auxiliary decoder. They calculate the MSE loss between the predictions of the main decoder and each auxiliary decoder without creating pseudo labels. It is worth noting that the unsupervised loss is not used to supervise the main decoder. Following CCT, subsequent methods like GCT [49] and CPS [24] have been proposed to introduce network perturbation for consistency regularisation. Both of them use the same network structures but with different weight initialisation. CPS differs from GCT by using pseudo-labels generated from two networks to enforce consistency, whereas GCT achieves consistency regularisation by minimising the loss between the probability predictions of networks working in conjunction with a flow detector. Although the network perturbation with different weight initialisation provides some diversity for pseudo labels in GCT and CPS, the ability to generate diversity is still limited. Another group of SSL methods is built upon the teacher-student architecture. As a classic teacher-student model, MT [50] uses an EMA of the student's weights for the teacher, trains student models with augmented inputs, and applies a consistency cost to align their outputs. ICNet [6] was proposed to use teacher networks to improve the quality of pseudo labels and increase the model difference based on an iterative contrast network. Building on the teacher-student architecture, some of its variations have become SOTA. For example, iMAS [51] highlights the importance of instance differences and introduces instance-specific and model-adaptive supervision for semi-supervised semantic segmentation. AugSeg [52] employs an enhanced data augmentation technique for input data and adaptively injects labelled information into unlabelled samples based on the model's estimated confidence in each sample. These methods consistently rely on the same network architecture with different weights. The previously proposed DiverseModel argues that advocating the use of different models can obtain distinct and complementary features from these models, even with the same input data. Specifically, DiverseModel explores different networks in parallel to generate more diversity of pseudo labels to improve training effectiveness. Although these discussed methods achieve competitive performance, they consistently rely on multiple networks, making them bulky and potentially unsuitable for users with limited computational resources.

Ensemble machine learning is a concept that employs multiple learners and combines their predictions [53]. Bagging, a form of ensemble learning, is a technique aimed at reducing prediction variance by creating multiple iterations of a predictor and then utilising them to form an aggregated predictor [54]. Specifically, bagging creates sample subsets by randomly selecting from the training dataset and subsequently utilises these acquired subsets to train the foundational models for integration. When predicting a numerical outcome, aggregation is done by averaging the different versions, whereas for class prediction, it is based on a majority vote. Bagging is a commonly employed approach for enhancing the robustness and precision of machine learning algorithms for classification and regression [55].

III. DiverseHead (Cross-head Supervision)

This section introduces the details of the proposed DiverseHead SSL method, which uses a single network with multiple heads, illustrated in Figure 2. Each head has 2 convolutional layers. Unlike CCT, the proposed DiverseHead method treats all heads equally, avoiding a distinction between the main and auxiliary heads. Meanwhile, all heads benefit

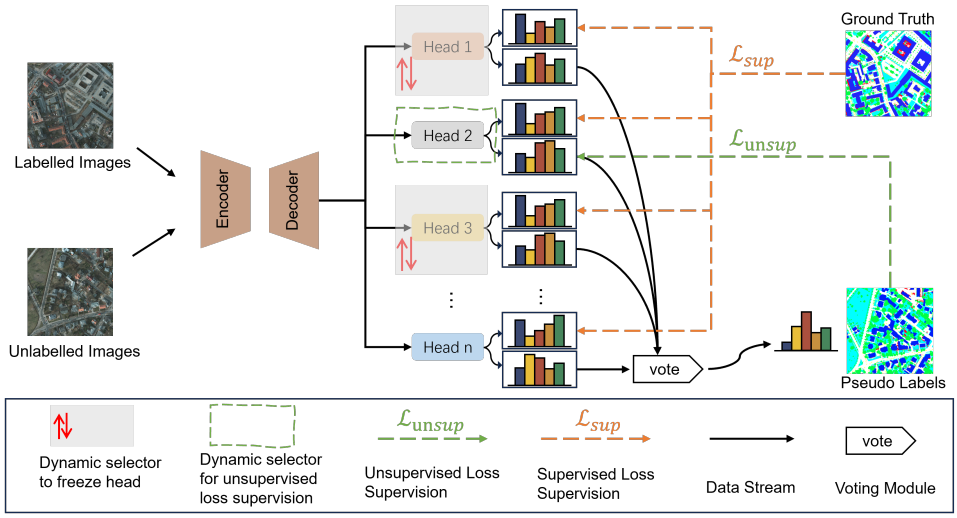

Fig. 2. DiverseHead: an online semi-supervised learning approach. This figure applies the dynamic freezing strategy: the freezers (Dynamic Selector in the figure) randomly select a certain number of heads to freeze the parameter of heads (not updated by backpropagation). Additionally, during every iteration, all heads undergo supervision through a supervised loss, yet each head is randomly chosen to be updated by an unsupervised loss.

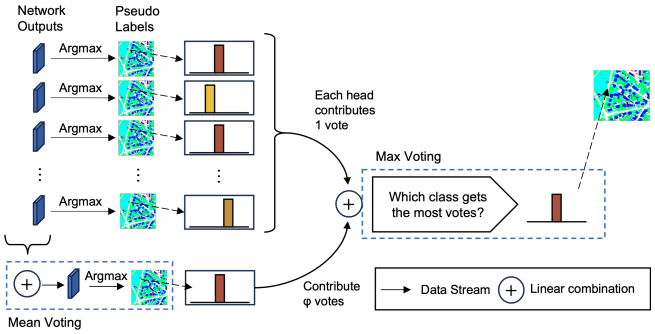

Fig. 3. The Proposed Voting Module: a voting mechanism for the pseudo label creation. In the unsupervised part, the voting module combines the mean output of multiple heads (mean voting) and individual pseudo labels (max voting) to generate more efficient pseudo labels. Argmax returns the indices of the maximum values of the prediction along the class dimension. The dashed arrow serves as an illustration of a pixel voting for its classification in a segmentation map.

from labelled data by applying supervised losses to each. This helps create better pseudo labels in the subsequent steps of the proposed method. On the other hand, rather than adding artificial noise to perturb head features as in CCT, DiverseHead introduces two perturbation strategies: Dynamic Freezing and Dropout. They are demonstrated in Figure 4 and explained in detail in Section III-A. During each training iteration, in addition to supervised losses, an unsupervised loss is computed between the pseudo-label and the prediction. The pseudo label is derived from an efficient voting module illustrated in Figure 3. Although non-differentiable operations (e.g. Voting and Argmax) could potentially limit optimisation during training, in our framework, parameter perturbations, feature perturbations, and independent initialisation across heads and models inject sufficient diversity and stochasticity into the training process. These factors enable effective gradient-based optimisation despite the presence of non-differentiable components. The proposed method can be seen as a combination of self-training and consistency regularisation, leveraging model predictions to supervise itself and forcing all perturbed heads to produce consistent outputs.

To provide a detailed description of the proposed semi-supervised framework, given both a labelled data set $\mathcal{B}^l = \{(x_i, y_i)\}_{i=1}^{M}$ containing M images and an unlabelled data set $\mathcal{B}^u = \{u_i\}_{i=1}^{N}$ with N images, the network Q is constructed with multiple heads denoted as $\{head^i\}_{i=1}^{L}$, where L is the number of heads. Each of these heads is initialised differently. The proposed SSL approach expects to gain the trained network Q leveraging the labelled and unlabelled data.

The output of each head refers to a probability-based prediction for all classifications. When working with labelled data, the supervised loss $\mathcal{L}_{sup,s}^{j}$ between the $s'th$ ground truth y_s and its corresponding prediction p_s^j for $j'th$ head is defined by using the standard cross-entropy loss function ℓ_{ce}:

$$\mathcal{L}_{sup,s}^{j} = \frac{1}{W \times H} \sum_{i=1}^{W \times H} \ell_{ce}\left(p_{i,s}^{j}, y_{i,s}\right), \tag{1}$$

where W and H refer to the width and height of input images. The final supervised loss for the $s'th$ labelled data is determined by the mean of the losses across all heads

$$\mathcal{L}_{sup,s} = \frac{1}{L} \sum_{j=1}^{L} \mathcal{L}_{sup,s}^{j} \tag{2}$$

where L is the number of heads.

In each iteration, we also get a prediction r_k^j corresponding to the $k'th$ unlabelled data u_k^j from the $j'th$ head. Inspired by ensemble learning, a voting module is proposed to get high-precision pseudo labels. There are two voting mechanisms in the proposed voting module, called mean voting and max voting, respectively shown in Figure 3. The former aggregates the predictions from all heads to create a combined prediction $\hat{r}_k^{mean}$. Subsequently, this combined prediction is used to calculate the mean pseudo labels $\hat{y}_k^{mean}$ through an $argmax$ operation, which returns the indices of the maximum values

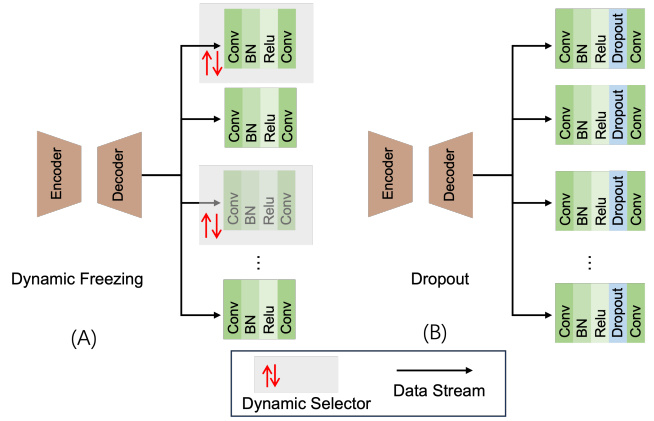

Fig. 4. The proposed two perturbation methods: (A) Dynamic Freezing and (B) Dropout. Dynamic Freezing was designed to enhance the parameter diversity across multiple heads. During each training iteration, a specific subset of heads is randomly selected (The Dynamic Selector in the figure is used for this purpose), and their parameters are frozen, meaning they are not updated by minimising the loss in that iteration. These parameters are unfrozen before the next iteration begins. Instead, for Dropout, each channel of features passed through each head is independently zeroed out with a dropout rate p during each forward pass.

of the prediction along the class dimension. Apart from the mean voting module, the individual pseudo label is generated from the output of each head by the $argmax$ function. The latter regards all pseudo labels as voters, in which the mean pseudo label contributes φ weight and each individual pseudo label contributes unit weight. φ is a learnable parameter and its value changes depending on the dataset and training process. The max voting module returns the class number that gets the most votes for each pixel. After max voting, an optimal pseudo label $\hat{y}_k^{final}$ is created to calculate unsupervised loss $\mathcal{L}_{unsup,k}$ by using the cross-entropy loss function:

$$\mathcal{L}_{unsup,k} = \frac{1}{W \times H} \sum_{i=1}^{W \times H} \ell_{ce} \left(r_{i,k}^{main}, \hat{y}_{i,k}^{final} \right), \quad (3)$$

where r_k^{main} is the prediction from the randomly selected single head.

Finally, the whole loss can be written as

$$\mathcal{L} = \mathcal{L}_{sup,s} + \lambda \mathcal{L}_{unsup,k}, \quad (4)$$

where λ is the trade-off weight between the supervised and unsupervised losses. We use different subscripts (s and k) for the supervised and unsupervised losses, respectively, to account for the difference in the numbers of labelled and unlabelled data. Specifically, the number of labelled data samples is smaller than that of unlabelled data samples. The solution to this issue is that, in each epoch, the labelled data set is repeatedly used in cycles for multiple iterations until all unlabelled data have been processed once.

A. Perturbation Methods

Based on the framework of DiverseHead, we propose a parameter perturbation method called dynamic freezing as shown in Figure 4 (a). The pseudocode of the algorithm for

Algorithm 1 DiverseHead Semi-Supervised Learning with Dynamic Freezing Pseudocode. The Labelled Training Dataset is defined as $\mathcal{B}^l$. Since the number of labelled data is smaller than that of unlabelled data, the labelled data is used in cycles for one epoch. We define it as $cycle\left(\mathcal{B}^l\right)$.

INITIALIZATION:
Randomly initialise model Q
Initialise backbone using ResNet-50
INPUT: Labelled Training Dataset $\mathcal{B}^l = \{(x_i, y_i)\}_{i=1}^{M}$
Unlabelled Training Dataset $\mathcal{B}^u = \{u_i\}_{i=1}^{N}$
$L = length(heads)$
for $\{(x_s, y_s), u_k\}_{k=1,s=k\%M}^{N} \in \{cycle\left(\mathcal{B}^l\right), \mathcal{B}^u\}$ **do**
  $\mathcal{R} = Randint(0, L, \frac{1}{2}L)$
  $Freeze(\{head^i\}_{i \in \mathcal{R}})$
  for $j \in \{1, ..., L\}$ **do**
    $p_s^j = Q^{head_j}(x_s^j)$
    $\mathcal{L}_{sup,s}^j = loss(p_s^j, y_s)$ based on (1)
    $r_k^j = Q^{head_j}(u_k^j)$
    $\hat{y}_k^j \leftarrow argmax(r_k^j)$
  $r_k^{mean} = sum\left(\{r_k^j\}_{j=1}^{L}\right)$
  $\hat{y}_k^{mean} \leftarrow argmax(r_k^{mean})$
  $\hat{y}_k^{final} \leftarrow voting(\{\hat{y}_k^j\}_{j=1}^{L}, \hat{y}_k^{mean})$
  $r_k^{main} \leftarrow sample\left(\{r_k^1, r_k^2, ..., r_k^L\}\right)$
  $\mathcal{L}_{unsup,k} \leftarrow loss(r_k^{main}, \hat{y}_k^{final})$ based on (3)
  $\mathcal{L} \leftarrow \frac{1}{L}\sum_{j=1}^{L} \mathcal{L}_{sup,s}^j + \lambda \mathcal{L}_{unsup,k}$
  Minimize $\mathcal{L}$ to update model Q
  $Unfreeze(\{head^i\}_{i=R})$
OUTPUT: Trained model Q

DiverseHead with the perturbation of dynamic freezing is given in **Algorithm** 1. Specifically, we use DeepLabv3+ with a ResNet-50 backbone as the segmentation network in our framework, ensuring consistency with the methods compared in this paper. The backbone is pretrained on ImageNet, and other parameters are initialised by Kaiming Initialization. Both labelled and unlabelled data are input to the framework, then the first step of dynamic freezing refers to a process where half of the heads are randomly selected to be frozen during each iteration. This implies that the parameters within these selected heads will remain unchanged and not undergo updates for the current iteration. Every head has an equal probability of being chosen. Following the cross-head supervision above, the segmentation model is updated by the supervised and unsupervised losses. Before going to the next iteration, those frozen heads are unfrozen.

Another form of perturbation involves the use of dropout layers in the proposed DiverseHead structure to increase the diversity of features during training. We introduce a dropout layer after each convolutional block in each segmentation head of the network, as shown in Figure 4 (b). The pseudocode of this method is given in **Algorithm** 2. As both dynamic freezing and dropout perturbation utilise the same semi-supervised learning framework, DiverseHead, the network initialisation and supervised loss calculation procedures stay the same. Using dropout in DiverseHead, specific components of the weights in the heads of the network are randomly assigned a value of zero with a dropout rate (determining the probability), using samples derived from a Bernoulli distribution. This dropout operation is employed to enhance the variability of

Algorithm 2 DiverseHead Semi-Supervised Learning with Dropout Pseudocode. The Labelled Training Dataset is defined as $\mathcal{B}^l$. Since the number of labelled data is smaller than that of unlabelled data, the labelled data is used in cycles for one epoch. We define it as $cycle\left(\mathcal{B}^l\right)$.

INITIALIZATION:
Randomly initialise model Q
Initialise backbone using ResNet-50
Add dropout layer before the last convolutional layer for each head
INPUT: Labelled Training Dataset $\mathcal{B}^l = \{(x_i, y_i)\}_{i=1}^{M}$
Unlabelled Training Dataset $\mathcal{B}^u = \{u_i\}_{i=1}^{N}$
$L = length(heads)$
for $\{(x_s, y_s), u_k\}_{k=1,s=k\%M}^{N} \in \{cycle\left(\mathcal{B}^l\right), \mathcal{B}^u\}$ **do**
  for $j \in \{1, ..., L\}$ **do**
    $p_s^j = Q^{head_j}(x_s^j)$
    $\mathcal{L}_{sup,s}^j = loss(p_s^j, y_s)$ based on (1)
    $r_k^j = Q^{head_j}(u_k^j)$
    $\hat{y}_k^j \leftarrow argmax(r_k^j)$
  $r_k^{sum} = sum\left(\{r_k^j\}_{j=1}^{L}\right)$
  $\hat{y}_k^{mean} \leftarrow argmax(r_k^{mean})$
  $\hat{y}_k^{final} \leftarrow voting(\{\hat{y}_k^j\}_{j=1}^{L}, \hat{y}_k^{mean})$
  $r_k^{main} \leftarrow sample\left(\{r_k^1, r_k^2, ..., r_k^L\}\right)$
  $\mathcal{L}_{unsup,k} \leftarrow loss(r_k^{main}, \hat{y}_k^{final})$ based on (3)
  $\mathcal{L} \leftarrow \frac{1}{L}\sum_{j=1}^{L}\mathcal{L}_{sup,s}^j + \lambda\mathcal{L}_{unsup,k}$
  Minimize $\mathcal{L}$ to update model Q
OUTPUT: Trained Model Q

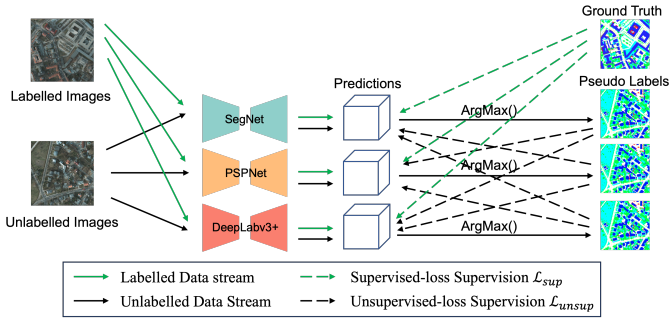

Fig. 5. DiverseModel: an online semi-supervised learning approach.

the output predictions. In the conducted experiments, the dropout rate is set as a hyperparameter with a value of 0.5. Although the dropout-based approach adheres to the same training pipeline with dynamic freezing outlined above, the distinction is that all heads remain unfrozen.

To evaluate the efficacy of individual perturbation methods in conjunction with the proposed DiverseHead techniques, each method is applied independently within the proposed framework. The performance of each combination is assessed in Section VI. Considering that various datasets may require differing levels of diversity in pseudo labels within the proposed DiverseHead framework, adjustments can be made by altering the number of frozen heads and dropout rates in the methods of dynamic freezing and dropout, respectively.

IV. Cross-model Supervision (DiverseModel)

The proposed DiverseModel differs from CPS, exploring different networks in parallel to generate comprehensive pseudo labels. As shown in Figure 5, the DiverseModel structure includes three distinct networks, which can be various semantic segmentation networks. In this paper, we choose three widely used segmentation networks for experiments, which are PSPNet [35], UNet [34], SegNet [33]. **Algorithm** 3 presents the pseudocode of the DiverseModel method. Since different networks pay different and complementary attention to the same input, this offers the basis for they are able to benefit from each other. In order to provide evidence for this claim, we executed the Gradient-weighted Class Activation Mapping (Grad-CAM) [26] technique for every network employed within the framework of the DiverseModel architecture. Grad-CAM visualises the areas of an image that are important to the model predictions from each network. Figure 6 depicts an example grad-CAM analysis for the BUILDING class in the Potsdam data set. Examining Figure 6, we can see that different networks pay different and complementary attention to the same input, and the pseudo labels from the DiverseModel prediction show the highest quality.

The labelled data is used in a regular supervised learning manner to train these models by using the standard cross-entropy loss function ℓ_{ce}. The supervised loss $\mathcal{L}_{sup,s}$ is expressed as:

$$\mathcal{L}_{sup,s} = \frac{1}{3}\sum_{n=1}^{3}\frac{1}{W \times H}\sum_{i=1}^{W\times H}\ell_{ce}\left(p_{i,s}^n, y_{i,s}\right), \tag{5}$$

where $p_{i,s}^n$ represents the $i\ th$ predicted pixel of the $s\ th$ sample from the $n\ th$ network.

In addition, unlabelled data is used to generate pseudo labels, which are then exploited for cross-supervision to inform each network. Different from the version presented in [25], all loss calculations in this work solely focus on cross-entropy loss to avoid the variations in performance resulting from different types of loss functions. The predictions obtained by each network are denoted as $\{p^1, p^2, p^3\}$, which are used for generating pseudo labels $\{\hat{r}^1, \hat{r}^2, \hat{r}^3\}$ through the $argmax$ operation. For instance, the cross pseudo supervision loss $\mathcal{L}_{unsup}^{12}$ between the prediction p^1 from the first network and the pseudo label r^2 generated by the second network is defined as:

$$\mathcal{L}_{unsup,k}^{12} = \frac{1}{W \times H}\sum_{i=1}^{W\times H}\ell_{ce}\left(p_{i,k}^1, \hat{r}_{i,k}^2\right). \tag{6}$$

where, $p_{i,k}^1$ represents the $i\ th$ predicted pixel of the $k\ th$ sample from the $1\ st$ network.

The cross-pseudo supervision among the three networks creates six losses in the same way. The unsupervised loss $\mathcal{L}_{unsup}$ is the average of the six individual losses, as shown below

$$\begin{aligned}\mathcal{L}_{unsup,k} = \frac{1}{6}\Big(&\mathcal{L}_{unsup,k}^{12} + \mathcal{L}_{unsup,k}^{13} + \mathcal{L}_{unsup,k}^{21} \\ &+ \mathcal{L}_{unsup,k}^{23} + \mathcal{L}_{unsup,k}^{31} + \mathcal{L}_{unsup,k}^{32}\Big).\end{aligned} \tag{7}$$

The total loss $\mathcal{L}$ is the linear addition of $\mathcal{L}_{sup,s}$ and $\mathcal{L}_{unsup,k}$, which is previously given in equation (4)

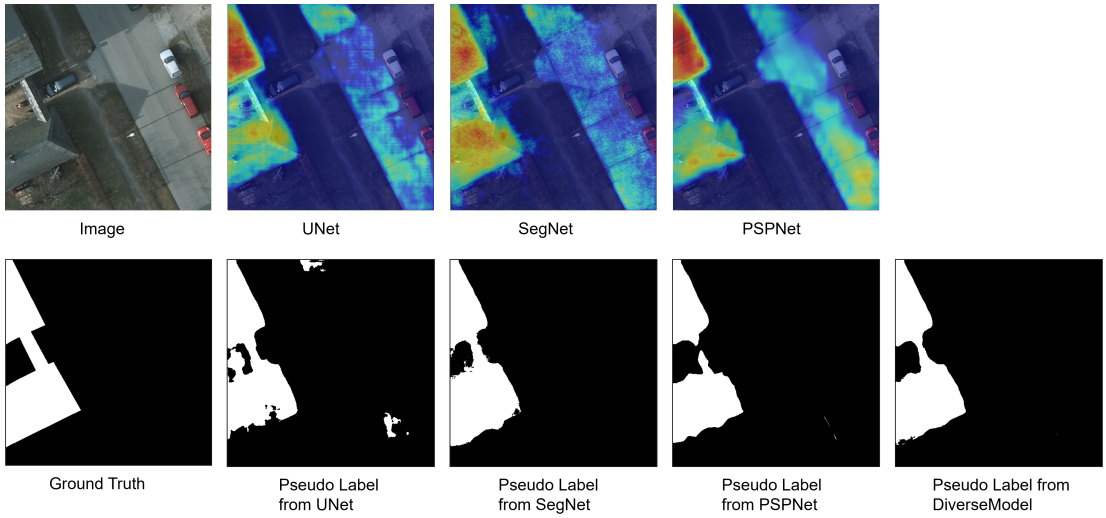

Fig. 6. The upper section displays Grad-CAM outputs from individual networks within the DiverseModel architecture using the Potsdam dataset. The lower section showcases both the ground truth and pseudo labels generated through predictions from each network. In the lower right corner, the pseudo-label from DiverseModel is presented.

Algorithm 3 DiverseModel Semi-supervised Learning Pseudocode. The Labelled Training Dataset is defined as $\mathcal{B}^l$. Since the number of labelled data is smaller than that of unlabelled data, the labelled data is used in cycles for one epoch. We define it as $cycle\left(\mathcal{B}^l\right)$.

INITIALIZATION:
Randomly initialise three models, PSPNet Q^1, UNet Q^2, SegNet Q^3
INPUT: Labelled Training Dataset $\mathcal{B}^l = \{(x_i, y_i)\}_{i=1}^{M}$
Unlabelled Training Dataset $\mathcal{B}^u = \{u_i\}_{i=1}^{N}$
for $\{(x_s, y_s), u_k\}_{k=1,s=k\%M}^{N} \in \left\{cycle\left(\mathcal{B}^l\right), \mathcal{B}^u\right\}$ **do**
$\mathcal{L}_{sup,s} = loss(Q^1(x_s), y_s) + loss(Q^2(x_s), y_s) + loss(Q^3(x_s), y_s)$ based on (1)
$\hat{r}_k^1 \leftarrow argmax(Q^3(u_k))$
$\hat{r}_k^2 \leftarrow argmax(Q^3(u_k))$
$\hat{r}_k^3 \leftarrow argmax(Q^3(u_k))$
$\mathcal{L}_{unsup,k} = loss(Q^1(u_k), \hat{r}_k^2) + loss(Q^1(u_k), \hat{r}_k^3) + loss(Q^2(u_k), \hat{r}_k^1) + loss(Q^2(u_k), \hat{r}_k^3) + loss(Q^3(u_k), \hat{r}_k^1) + loss(Q^3(u_k), \hat{r}_k^2)$
$\mathcal{L} \leftarrow \mathcal{L}_{sup,s} + \lambda \mathcal{L}_{unsup,k}$
Minimize $\mathcal{L}$ to update model Q^1, Q^2, Q^3
OUTPUT: Trained Model Q^1, Q^2, Q^3

V. Dataset Description

We employed diverse remote sensing datasets to assess both the proposed techniques and state-of-the-art methods, specifically including (1) the ISPRS Potsdam dataset [56], (2) the DFC2020 dataset [57], (3) the RoadNet dataset [58], and (4) the Massachusetts Buildings dataset [59]. In the sequel, we share the details of each dataset utilised in this paper.

1) ISPRS Potsdam Semantic Labelling dataset is an open-access benchmark dataset provided by the International Society for Photogrammetry and Remote Sensing (ISPRS). The true orthophoto (TOP) and DSM modalities have a ground sampling distance of 5 cm. Six land cover classes were identified by hand annotation of this dataset: *impervious surfaces, buildings, low vegetation, trees, cars*, and *clutter/background*. This dataset has 38 patches, each measuring 6000×6000 pixels. The patches include orthorectified optical pictures of red, green, and blue bands, as well as infrared (IR) and matching digital surface models (DSM). All of these data tiles were divided into 512×512 patches for computational reasons, yielding 3456, 201, and 1815 samples for the training, validation, and test sets, respectively. We randomly select a quarter of the training data as labelled data and use the remaining three quarters as unlabelled data for all SSL approaches.

2) DFC2020 is the 2022 IEEE GRSS Data Fusion Contest dataset, which is based on the SEN12MS dataset [60]. It provides Sentinel-1 SAR imagery, Sentinel-2 multispectral imagery, and corresponding land cover maps with ten coarser-grained classes. The size of all patches is 256×256 pixels. There are 6112, 986 and 5127 images for training, validation and test sets, respectively. In this paper, one-fifth of the labelled data is for training, while the remaining four-fifths of the data is employed as unlabelled data.

3) RoadNet is a benchmark dataset for roadnet detection with 0.21-m spatial resolution.It includes RGB images and related road surface maps for the segmentation task. The number of samples for training, validation and testing is 410, 45 and 387, respectively. A quarter of the annotated training data is used for the supervised part, while the remaining three-quarters of the training data are employed as unlabelled data for SSL training purposes.

4) Massachusetts Buildings dataset predominantly encompasses urban and suburban regions, encompassing structures of varying scales. It consists of 151 aerial RGB images with a resolution of 1 m^2 per pixel and corresponding building masks with the size of 1500×1500 pixels. The dataset was split into 137 training, 10 testing, and 4 validation images with labels. One-quarter of the annotated training images are used for supervision, while the remaining three-quarters are treated as unlabelled for SSL.

VI. Experimental analysis

A. Implementation Details

Our approaches are implemented using PyTorch. Following [37], we employed a polynomial learning rate policy

with a mini-batch SGD optimizer, where the current learning rate is calculated as the initial learning rate multiplied by $\left(1-\frac{\text{iter}}{\text{max-iter}}\right)^{\text{power}}$. The initial learning rate and power are set to 0.01 and 0.9, respectively. All experiments were conducted on the GW4 Isambard with an NVIDIA A100-sxm GPU and an AMD EPYC 7543P CPU [61].

For a fair comparison, both the state-of-the-art methods and the proposed DiverseHead use DeepLabv3+ with a ResNet-50 backbone pre-trained on ImageNet as the semantic segmentation network. We comprehensively analysed all methods by quantifying performance via class-related measures, including overall accuracy (OA), user's accuracy (UA), producer's accuracy (PA), mean intersection over union (mIoU), and F_1-score. Expressions of all five performance metrics are given as follows: $\text{OA} = \frac{TP+TN}{TP+TN+FP+FN}$, $\text{UA} = \frac{TP}{TP+FP}$, $\text{PA} = \frac{TP}{TP+FN}$, $\text{mIoU} = \frac{|\text{TP}|}{|\text{TP+FN+FP}|}$, $F_1 = \frac{2 \cdot PA \cdot UA}{PA+UA}$, where *TP*, *TN*, *FP*, and *FN* refer to the numbers of pixels that are true positives, true negatives, false positives, and false negatives for each class, respectively.

B. Quantitative Results and Analysis

We evaluate the proposed approaches, along with classic and state-of-the-art SSL methods, using the five performance metrics outlined above, across four remote sensing imagery segmentation datasets: Potsdam, DFC2020, RoadNet, and Massachusetts Building. The average results across four datasets are presented in Table I. It presents an overall performance of the proposed methods using a single network in comparison to traditional SSL methods that employ dual networks. Since some state-of-the-art SSL methods in computer vision, such as UniMatch, iMAS, and AugSeg, are designed for RGB image segmentation and rely on RGB-based datasets for pretraining, these methods are not included in the average performance across the four datasets, including multi-band remote sensing data. However, a detailed performance comparison of these state-of-the-art and classic SSL methods across various datasets, including both multi-band and RGB remote sensing imagery, will be presented and discussed later in this section. From the average results in Table I, the proposed DiverseHead (DF) delivers the best overall performance, achieving the highest score in 4 out of 5 metrics, which are highlighted in red. DiverseModel demonstrates the second-best performance across most of the metrics. In particular, for metrics of UA, the proposed DiverseModel exhibits an improvement of over 3.48% compared to another network-perturbation-based approach, CPS. Although the performance of DiverseHead (DT) is slightly lower than that of DiverseHead (DF) and DiverseModel, these three methods show similar average performance, but both of them outperform the other compared methods. With these used remote sensing datasets, MT demonstrates notably inferior average performance across various metrics. CCT, GCT, and CPS exhibit comparable performance, yet the performance superiority of CPS is notably evident in its PA metric. It is important to highlight that DiverseHead is a very lightweight method and reaches a very competitive average performance compared to others.

TABLE I
AVERAGE PERFORMANCE COMPARISON WITH THE CLASSIC SSL METHODS WITH MULTIPLE NETWORKS ON FOUR DATASETS. DT AND DF INDICATE DROPOUT AND DYNAMIC FREEZING, RESPECTIVELY.

Models	OA	UA	PA	mIoU	F_1
MT [50]	86.35%	74.93%	81.53%	66.03%	77.99%
CCT [48]	87.14%	76.49%	82.50%	67.76%	79.23%
GCT [49]	87.45%	75.35%	82.31%	67.19%	78.65%
CPS [24]	88.06%	75.97%	85.27%	68.23%	80.22%
DiverseModel [25]	88.77%	**79.54%**	85.18%	70.92%	82.02%
DiverseHead (DT)	88.69%	78.51%	85.42%	70.63%	81.64%
DiverseHead (DF)	**89.00%**	79.14%	**85.83%**	**71.28%**	**82.17%**

Table II presents the number of parameters of SSL frameworks during training. Apart from DiverseHead, all other classic SSL approaches typically involve parameters exceeding three hundred megabytes. Among these approaches, DiverseModel is the largest architecture; however, it outperforms other network-perturbation-based methods, including MT, CCT, GCT, and CPS, as demonstrated in Table I. In contrast, the proposed DiverseHead is highly lightweight, using just a single segmentation model with multiple heads, each consisting of only 2 convolutional layers, thus eliminating the need for multiple networks during training. Specifically, the parameter size of the DiverseHead (DT&DF) is only 16% bigger than that of the single network (Base in Table II), whereas the parameter size of other reference semi-supervised architectures is at least twice that of the single network. Despite requiring fewer parameters, the proposed DiverseHead method surpasses the classic SSL methods by at least 1% in accuracy and 3% in mIoU, while achieving performance comparable to the largest method, DiverseModel.

TABLE II
THE REQUIRED PARAMETER SIZE OF EACH SEMI-SUPERVISED LEARNING APPROACH. BASE MEANS SEGMENTATION NETWORK DEEPLABV3+. DT AND DF PRESENT DROPOUT AND DYNAMIC FREEZING, RESPECTIVELY.

	CPS [24]	MT [50]	CCT [48]	GCT [49]
Size (MB)	303.378	303.378	337.655	335.049
	DiverseModel [25]	DiverseHead (DT)	DiverseHead (DF)	Base
Size (MB)	936.262	175.806	175.806	151.689

Specifically, we present the results of different semi-supervised learning methods applied to the two multi-band remote sensing imagery datasets, namely Potsdam and DFC2020, in the table III. The performance of the *DiverseNet*, namely DiverseHead and DiverseModel, is noticeably superior to that of the other listed semi-supervised learning methods across these evaluated datasets. In particular, for the Potsdam datasets, the proposed DiverseHead (DF) attains the highest performance across 4 out of 5 segmentation metrics, whilst DiverseModel emerges as the second-best method based on its results. However, in the case of the DFC2020 dataset, DiverseModel reaches the best for most performance metrics, securing the top position whilst DiverseHead closely follows as the second-best method, despite their similar performance.

TABLE III
PERFORMANCE COMPARISON WITH THE STATE-OF-THE-ART ON DATASETS CONTAINING IMAGES WITH MORE THAN THREE BANDS, NAMELY POTSDAM AND DFC2020. DT AND DF PRESENT DROPOUT AND DYNAMIC FREEZING, RESPECTIVELY.

	Model	OA	UA	PA	mIoU	F_1
Potsdam	MT[50]	81.98%	73.66%	78.39%	63.07%	75.95%
	CCT[48]	82.66%	74.64%	77.62%	64.16%	76.10%
	GCT[49]	83.99%	75.65%	80.81%	65.80%	78.14%
	CPS[24]	85.00%	75.76%	82.94%	66.69%	79.19%
	DiverseModel[25]	85.76%	76.75%	**83.45%**	67.85%	79.96%
	DiverseHead (DT)	84.66%	77.04%	80.78%	67.12%	78.87%
	DiverseHead (DF)	**85.98%**	**79.15%**	82.87%	**69.63%**	**80.97%**
DFC2020	MT[50]	78.64%	59.59%	73.57%	50.44%	65.85%
	CCT[48]	79.71%	59.87%	76.40%	51.04%	67.13%
	GCT[49]	80.84%	61.47%	71.43%	52.17%	66.07%
	CPS[24]	81.49%	61.74%	79.46%	53.20%	69.49%
	DiverseModel[25]	81.87%	**62.20%**	**80.69%**	53.71%	**70.25%**
	DiverseHead (DT)	**82.02%**	62.13%	80.61%	**53.81%**	70.18%
	DiverseHead (DF)	81.78%	61.81%	80.21%	53.46%	69.82%

Images from Potsdam consist of 4 bands, while images from DFC2020 contain 13 bands. The DiverseModel exhibits greater proficiency in processing datasets with more bands, while Diverse is more effective in handling images with fewer bands.

The proposed DiverseHead with two perturbation methods is a foundational framework, requiring fewer parameters and making it highly suitable for users with limited computational resources. Also, this proposed foundational framework can be easily combined with various teacher-student SSL methods to improve their performance. Table IV presents the results of the proposed methods and various SSL methods, including state-of-the-art methods in computer vision, which incorporate various enhancement strategies, on RGB remote sensing image segmentation datasets, specifically RoadNet and Massachusetts Building. DiverseHead demonstrates superior performance compared to other classic frameworks, such as MT, CCT, GCT, CPS, FixMatch and UniMatch, while maintaining a lower parameter requirement and quicker training process. Although some currently proposed SSL methods, iMAS, AugSeg and DWL, outperform the proposed DiverseHead, integrating these state-of-the-art methods with the proposed DiverseHead idea leads to further performance improvement and reaches the best result in Table IV.

To further support the computational efficiency of the proposed method, we measured the wall-clock training time required for 100 iterations across various state-of-the-art SSL frameworks, using identical hardware and implementation settings. As shown in Figure 7, *DiverseHead (dropout)* achieves the lowest training time (40.65 seconds) among all compared methods. This demonstrates that our approach introduces minimal computational overhead, providing strong empirical evidence of its lightweight nature during training.

In order to further prove the efficiency of multi-head supervision, we considered a downgraded version of the current DiverseHead approach, called single-head supervision (SHS), which consists of only a single head. During each training iteration, the current model's predictions for unlabelled data are used to generate pseudo labels through an $argmax$ operation. These generated pseudo labels are then used to calculate the unsupervised loss. Also, we provide the performance of the segmentation network namely DeepLabv3+, called Base in Table V, only trained by labelled data (part of each dataset) without using unlabelled data. The performance of the Base and SHS, along with the proposed DiverseHead, are presented in Table V to provide experimental evidence of the improvement and efficiency of the use of multiple heads in the DiverseHead architecture. While the single-head supervision benefits from the pseudo labels and exhibits better performance compared to the Base model, especially improving the metric of PA, both versions of DiverseHead approaches demonstrate a further significant improvement by taking advantage of multiple heads. Specifically, the mIoU of DiverseHead (DF) is 6.91% higher than that of Base and 5.12% higher than that of SHS for Potsdam. While the performance of DiverseHead (DT) is slightly lower than that of DiverseHead (DF), its performance is still much better than that of SHS and Base. Similarly, Both DiverseHead versions demonstrate superior performance on the RoadNet dataset compared to SHS and Base, particularly evident in the mIoU metric, where DiverseHead (DT) exhibits a significant improvement of 6.8% over the Base.

Fig. 7. Training time (in seconds) for 100 iterations across different SSL methods. DiverseHead refers to the dropout version of our proposed method.

TABLE IV
PERFORMANCE COMPARISON WITH THE STATE-OF-THE-ART ON RGB-BAND DATASETS, NAMELY ROADNET AND MASSACHUSETTS. DT AND DF PRESENT DROPOUT AND DYNAMIC FREEZING, RESPECTIVELY.

	Model	OA	UA	PA	mIoU	F_1
RoadNet	MT[50]	94.61%	86.87%	88.18%	79.04%	87.52%
	CCT[48]	95.58%	88.67%	90.74%	82.18%	89.69%
	GCT[49]	95.29%	85.53%	91.99%	80.35%	88.64%
	CPS[24]	95.81%	89.10%	91.36%	82.96%	90.21%
	FixMatch [62]	96.02%	89.91%	91.63%	83.81%	90.76%
	UniMatch [63]	95.93%	90.58%	90.79%	83.71%	90.69%
	ICNet [62]	97.15%	93.48%	93.52%	88.19%	93.50%
	iMAS [51]	97.15%	94.15%	93.03%	88.32%	93.59%
	AugSeg [52]	97.36%	94.26%	93.79%	89.06%	94.03%
	DWL [47]	97.88%	94.81%	**95.48%**	90.96%	95.14%
	DiverseModel[25]	96.84%	93.12%	92.58%	87.11%	92.84%
	DiverseHead(DT) w/ Sngl-model	96.85%	92.14%	93.31%	86.91%	92.72%
	DiverseHead(DF) w/ Sngl-model	96.81%	92.60%	92.80%	86.88%	92.70%
	DiverseHead(DT) w/ iMAS	97.47%	94.24%	94.20%	89.40%	94.22%
	DiverseHead(DT) w/ AugSeg	97.70%	**95.85%**	93.96%	90.50%	94.90%
	DiverseHead(DT) w/ DWL	**97.96%**	95.33%	95.36%	**91.32%**	**95.35%**
Massachusetts	MT[50]	90.16%	79.58%	85.97%	71.55%	82.65%
	CCT[48]	90.59%	82.77%	85.23%	73.67%	83.98%
	GCT[49]	89.67%	78.75%	85.01%	70.42%	81.76%
	CPS[24]	89.95%	77.26%	87.33%	70.07%	81.98%
	FixMatch [62]	90.00%	81.34%	84.41%	72.13%	82.85%
	UniMatch [63]	89.86%	79.95%	84.79%	71.28%	82.30%
	ICNet [62]	92.24%	84.81%	88.64%	77.34%	86.68%
	iMAS [51]	92.40%	83.89%	89.90%	77.26%	86.79%
	AugSeg [52]	91.27%	80.92%	88.68%	73.98%	84.62%
	DWL [47]	91.92%	88.29%	86.00%	77.98%	87.13%
	DiverseModel[25]	90.62%	86.07%	84.01%	75.00%	85.03%
	DiverseHead(DT) w/ Sngl-model	91.21%	82.73%	86.98%	74.67%	84.80%
	DiverseHead(DF) w/ Sngl-model	91.43%	83.00%	87.45%	75.16%	85.17%
	DiverseHead(DT) w/ iMAS	92.92%	85.43%	**90.27%**	78.84%	87.78%
	DiverseHead(DT) w/ AugSeg	**93.35%**	87.60%	89.93%	**80.47%**	**88.75%**
	DiverseHead(DT) w/ DWL	92.57%	**88.49%**	87.37%	79.25%	87.93%

To evaluate the effectiveness of the proposed perturbation methods, DT and DF, in comparison to input perturbation, we conducted supplementary experiments involving data perturbation by introducing Gaussian noise to the input images. Consistency regularisation is achieved by enforcing the network to produce consistent predictions for the original and noisy input images, which is termed Input Perturbation in Table V. Both versions of DiverseHead exhibit better performance than Input Perturbation, whose performance is similar to SHS. To use the optimal Hyperparameter applied for input perturbation for fair comparison, hyperparameter tuning experiments are conducted on two datasets to confirm the standard deviation (SD) of Gaussian noise. Figure 8 illustrates the variation in mIoU as the SD changes. The mIoU value for both Potsdam and Roadnet peaked at an SD of 0.01. Thus, we set the SD of Gaussian noise for input perturbation 0.01 for the Potsdam and RoadNet datasets.

To investigate the influence of head count in the DiverseHead framework on performance, an ablation study was conducted on the Potsdam and RoadNet datasets using the proposed DiverseHead framework with dynamic freezing perturbation. The results in Table VI show that using 10 heads for DiverseHead achieves the best average performance across most metrics, despite the small performance differences among the variants. This supports the choice of using 10 heads as an effective design decision.

TABLE V
PERFORMANCE COMPARISON WITH SINGLE-HEAD SUPERVISION (SHS) AND BASELINE MODEL WHICH IS ONLY SUPERVISED BY LABELLED DATA. DT AND DF PRESENT DROPOUT AND DYNAMIC FREEZING, RESPECTIVELY.

	Model	OA	UA	PA	mIoU	F_1
Potsdam	Base	81.64%	73.86%	76.05%	62.72%	74.94%
	SHS	83.43%	74.57%	79.93%	64.51%	77.16%
	Input Perturbation	83.89%	75.73%	80.54%	65.69%	78.06%
	DiverseHead (DT)	84.66%	77.04%	80.78%	67.12%	78.87%
	DiverseHead (DF)	**85.98%**	**79.15%**	**82.87%**	**69.63%**	**80.97%**
RoadNet	Base	95.17%	87.62%	89.81%	80.73%	88.71%
	SHS	95.44%	87.31%	91.17%	81.37%	89.20%
	Input Perturbation	95.31%	87.42%	90.52%	81.03%	88.94%
	DiverseHead (DT)	**96.85%**	92.14%	**93.31%**	**86.91%**	**92.72%**
	DiverseHead (DF)	96.81%	**92.60%**	92.80%	86.88%	92.70%

To show the efficiency of DiverseModel, consisting of three different segmentation networks, we also evaluated the performance of each member network within the DiverseModel on the Potsdam dataset in Table VII. The component models UNet, SegNet, and PSPNet use unlabelled data through the

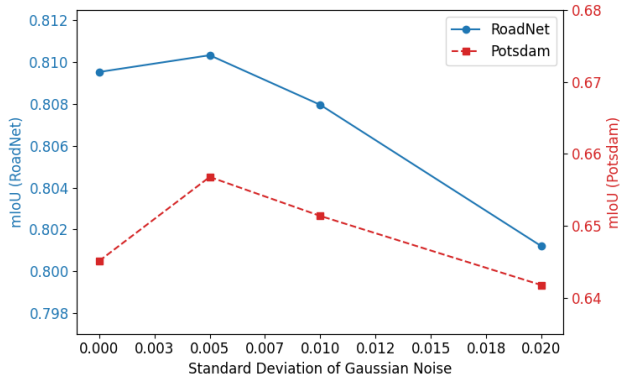

Fig. 8. Variation in mIoU as the standard deviation of the Gaussian noise added for input perturbation increases

TABLE VI
ABLATION STUDY ON THE EFFECT OF HEAD NUMBER IN DIVERSEHEAD WITH DYNAMIC FREEZING.

	# of heads	OA	UA	PA	mIoU	F_1
Potsdam	8	85.92%	77.86%	83.86%	68.91%	80.75%
	10	85.98%	79.15%	82.87%	69.63%	80.97%
	12	85.70%	78.54%	82.55%	68.96%	80.50%
RoadNet	8	96.63%	92.15%	92.42%	86.21%	92.29%
	10	96.81%	92.60%	92.80%	86.88%	92.70%
	12	96.80%	92.59%	92.77%	86.84%	92.68%
Average	8	91.27%	85.01%	**88.14%**	77.56%	86.52%
	10	**91.39%**	**85.88%**	87.83%	**78.25%**	**86.83%**
	12	91.25%	85.57%	87.66%	77.90%	86.59%

way of SHS. DiverseModel demonstrates a significant improvement across all performance metrics in the segmentation task compared to each network. In particular, the metric of PA experiences a notable improvement of 4.57% compared to the best-performing individual component model. This phenomenon can be attributed to the enhancement of pseudo-label diversity through cross-model supervision, resulting in a significant improvement in PA (recall) during the test phase. The findings suggest that the cross-supervision of different networks has the potential to achieve superior performance compared to the best-performing individual component.

C. Qualitative Results and Analysis

The visible results are presented in Figures 9. For each dataset, we randomly selected 2 cases of the original image, its ground truth, and predictions for the proposed methods and various classic methods. Also, the mIoU scores are calculated for all predictions with the ground truth for each case, which are shown in the sub-caption of each prediction. Based on the IoU values of visual predictions, the highest score is obtained by either DiverseModel or DiverseHead (DT&DF) for most cases. Especially for cases of the RoadNet dataset, both the DiverseModel and DiverseHead families achieve a mIoU of over 90% surpassing other methods by at least 6.35%. Visually, the segmentation maps of DiverseModel and DiverseHead display better overall similarity to the ground truth than other methods.

TABLE VII
PERFORMANCE COMPARISON OF THE DIVERSEMODEL WITH ITS CONSTITUENT NETWORKS ON THE POTSDAM DATASET

Model	OA	UA	PA	mIoU	F_1
UNet	83.28%	74.18%	78.88%	64.51%	76.46%
SegNet	82.37%	73.53%	77.46%	63.24%	75.45%
PSPNet	81.96%	74.23%	76.99%	63.08%	75.58%
DiverseModel	**85.76%**	**76.75%**	**83.45%**	**67.85%**	**79.96%**

VII. CONCLUSION

In this paper, we proposed a lightweight and efficient semi-supervised learning approach based on a multi-head structure called DiverseHead. Based on the multi-head structure, we provide two perturbation methods, namely dynamic freezing and dropout. Taking inspiration from the theory of bagging, a voting mechanism is proposed to generate beneficial pseudo-labels in the training stage. This simple and lightweight semi-supervised learning framework shows competitive performance for the segmentation of remote sensing imagery. Also, It can be easily combined with state-of-the-art SSL methods and further improve their performance. Furthermore, we conducted further analysis and additional evaluation on the previously proposed multi-network-based semi-supervised learning method known as DiverseModel. Based on the results obtained from the aforementioned four remote sensing datasets, DiverseHead and DiverseModel demonstrate comparable performance while significantly outperforming various classic semi-supervised learning frameworks. From the application perspective, the proposed DiverseNet could theoretically be utilised for a wide range of image-based tasks, including medical imaging, remote sensing, and natural image segmentation. Having evaluated the methods on datasets with multi-band images, our approach is likely to perform competitively in MRI and other multi-band image segmentation tasks. We will explore more applications using the proposed method in the future. For technical improvement, future work could explore investigating the integration of the two proposed and other perturbation strategies in SSL, and designing more sophisticated interaction mechanisms among component models and optimising the hyperparameters to enhance their synergy.

REFERENCES

[1] A. Vali, S. Comai, and M. Matteucci, "Deep learning for land use and land cover classification based on hyperspectral and multispectral earth observation data: A review," Remote Sens., vol. 12, no. 15, p. 2495, 2020.
[2] D. Zhang, F. Wang, L. Ning, Z. Zhao, J. Gao, and X. Li, "Integrating sam with feature interaction for remote sensing change detection," IEEE Transactions on Geoscience and Remote Sensing, 2024.
[3] J. Wang, M. Ma, P. Hang, S. Mei, L. Zhang, and H. Wang, "Remote sensing small object detection based on multi-contextual information aggregation," IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 2025.
[4] Z. Zhang, S. Mei, M. Ma, and Z. Han, "Adaptive composite feature generation for object detection in remote sensing images," IEEE Transactions on Geoscience and Remote Sensing, 2024.
[5] Z. Hu, J. Gao, Y. Yuan, and X. Li, "Contrastive tokens and label activation for remote sensing weakly supervised semantic segmentation," IEEE Transactions on Geoscience and Remote Sensing, 2024.

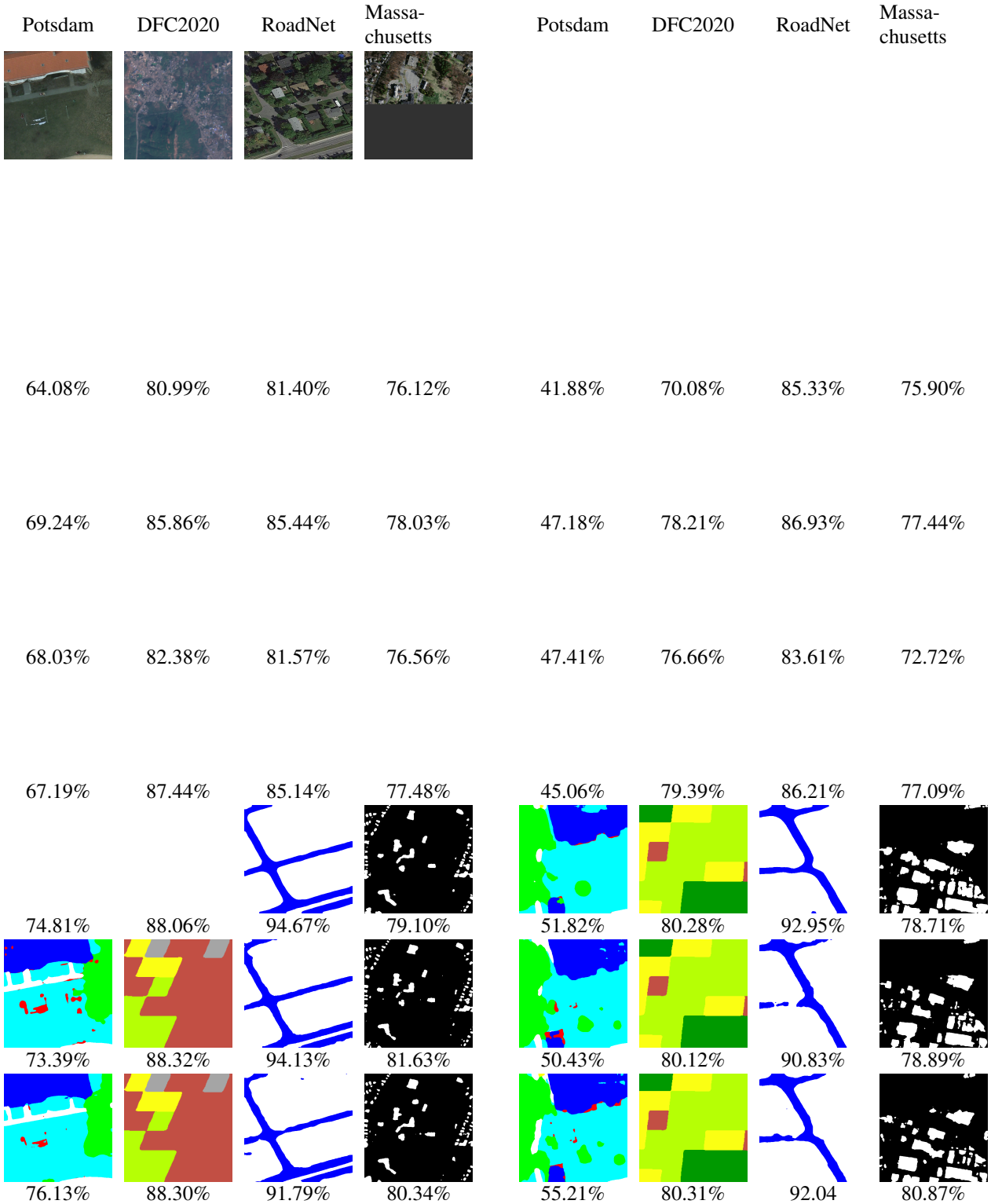

Fig. 9. Visual results of each method on four datasets. The mIoU score of the prediction for each method is shown in its sub-caption. The numbers in the first column are – (1): Image (2): Ground Truth (3): MT (4): CCT (5): GCT (6): CPS (7): DiverseModel (8): DiverseHead(DT) (9): DiverseHead(DF)

[6] J.-X. Wang, S.-B. Chen, C. H. Ding, J. Tang, and B. Luo, "Semi-supervised semantic segmentation of remote sensing images with iterative contrastive network," IEEE Geosci. Remote Sens. Lett., vol. 19, pp. 1–5, 2022.

[7] Z. Yang, Z. Yan, W. Diao, Q. Zhang, Y. Kang, J. Li, X. Li, and X. Sun, "Label propagation and contrastive regularization for semi-supervised semantic segmentation of remote sensing images," IEEE Trans. Geosci. Remote Sens., vol. 61, pp. 1–18, 2023.

[8] S. Wang, X. Huang, W. Han, J. Li, X. Zhang, and L. Wang, "Lithological mapping of geological remote sensing via adversarial semi-supervised segmentation network," Int. J. Appl. Earth Obs. Geoinf., vol. 125, p. 103536, 2023.

[9] B. Zhang, Y. Wang, W. Hou, H. Wu, J. Wang, M. Okumura, and T. Shinozaki, "FlexMatch: Boosting semi-supervised learning with curriculum pseudo labeling," Adv. Neural Inf. Process. Syst., vol. 34, pp. 18 408–18 419, 2021.

[10] A. Huang, Z. Wang, Y. Zheng, T. Zhao, and C.-W. Lin, "Embedding regularizer learning for multi-view semi-supervised classification," IEEE Trans. Image Process., vol. 30, pp. 6997–7011, 2021.

[11] L. Wang, Y. Liu, H. Di, C. Qin, G. Sun, and Y. Fu, "Semi-supervised dual relation learning for multi-label classification," IEEE Trans. Image Process., vol. 30, pp. 9125–9135, 2021.

[12] H. Hu, F. Wei, H. Hu, Q. Ye, J. Cui, and L. Wang, "Semi-supervised semantic segmentation via adaptive equalization learning," Adv. Neural Inf. Process. Syst., vol. 34, pp. 22 106–22 118, 2021.

[13] Y. Wang, H. Wang, Y. Shen, J. Fei, W. Li, G. Jin, L. Wu, R. Zhao, and X. Le, "Semi-supervised semantic segmentation using unreliable pseudo-labels," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2022, pp. 4248–4257.

[14] C. Wang, S. Zhao, L. Zhu, K. Luo, Y. Guo, J. Wang, and S. Liu, "Semi-supervised pixel-level scene text segmentation by mutually guided network," IEEE Trans. Image Process., vol. 30, pp. 8212–8221, 2021.

[15] J.-X. Wang, S.-B. Chen, C. H. Ding, J. Tang, and B. Luo, "Ranpaste: Paste consistency and pseudo label for semisupervised remote sensing image semantic segmentation," IEEE Transactions on Geoscience and Remote Sensing, vol. 60, pp. 1–16, 2021.

[16] B. Zhang, Y. Zhang, Y. Li, Y. Wan, H. Guo, Z. Zheng, and K. Yang, "Semi-supervised deep learning via transformation consistency regularization for remote sensing image semantic segmentation," IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 16, pp. 5782–5796, 2022.

[17] X. Lu, L. Jiao, F. Liu, S. Yang, X. Liu, Z. Feng, L. Li, and P. Chen, "Simple and efficient: A semisupervised learning framework for remote sensing image semantic segmentation," IEEE Transactions on Geoscience and Remote Sensing, vol. 60, pp. 1–16, 2022.

[18] X. Lu, L. Li, L. Jiao, X. Liu, F. Liu, W. Ma, and S. Yang, "Uncertainty-aware semi-supervised learning segmentation for remote sensing images," IEEE Transactions on Multimedia, 2025.

[19] L. Zhang, Z. Tan, Y. Zheng, G. Zhang, W. Zhang, and Z. Li, "A bias correction semi-supervised semantic segmentation framework for remote sensing images," IEEE Transactions on Geoscience and Remote Sensing, 2024.

[20] G. French, S. Laine, T. Aila, M. Mackiewicz, and G. Finlayson, "Semi-supervised semantic segmentation needs strong, varied perturbations," arXiv preprint arXiv:1906.01916, 2019.

[21] D. Filipiak, P. Tempczyk, and M. Cygan, "n-CPS: Generalising cross pseudo supervision to n networks for semi-supervised semantic segmentation," arXiv preprint arXiv:2112.07528, 2021.

[22] K. Chen and S. Wang, "Semi-supervised learning via regularized boosting working on multiple semi-supervised assumptions," IEEE Trans. Pattern Anal. Mach. Intell., vol. 33, no. 1, pp. 129–143, 2010.

[23] Y. Luo, J. Zhu, M. Li, Y. Ren, and B. Zhang, "Smooth neighbors on teacher graphs for semi-supervised learning," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 8896–8905.

[24] X. Chen, Y. Yuan, G. Zeng, and J. Wang, "Semi-supervised semantic segmentation with cross pseudo supervision," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2021, pp. 2613–2622.

[25] W. Ma, O. Karakuş, and P. L. Rosin, "Confidence guided semi-supervised learning in land cover classification," in IEEE Int. Geosci. Remote. Sens. Symposium, 2023, pp. 5487–5490.

[26] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," Int. J. Comput. Vis., vol. 128, no. 2, pp. 336–359, 2020.

[27] S. Dong, Y. Zhuang, Z. Yang, L. Pang, H. Chen, and T. Long, "Land cover classification from VHR optical remote sensing images by feature ensemble deep learning network," IEEE Geosci. Remote Sens. Lett., vol. 17, no. 8, pp. 1396–1400, 2019.

[28] H. Zheng, M. Gong, T. Liu, F. Jiang, T. Zhan, D. Lu, and M. Zhang, "HFA-Net: high frequency attention siamese network for building change detection in VHR remote sensing images," Pattern Recognit., vol. 129, p. 108717, 2022.

[29] H. Ghandorh, W. Boulila, S. Masood, A. Koubaa, F. Ahmed, and J. Ahmad, "Semantic segmentation and edge detection—approach to road detection in very high resolution satellite images," Remote Sens., vol. 14, no. 3, p. 613, 2022.

[30] K. Kikaki, I. Kakogeorgiou, P. Mikeli, D. E. Raitsos, and K. Karantzalos, "MARIDA: A benchmark for Marine Debris detection from Sentinel-2 remote sensing data," PloS One, vol. 17, no. 1, p. e0262247, 2022.

[31] H. Booth, W. Ma, and O. Karakuş, "High-precision density mapping of marine debris and floating plastics via satellite imagery," Sci. Rep., vol. 13, no. 1, p. 6822, 2023.

[32] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2015, pp. 3431–3440.

[33] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for image segmentation," IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 12, pp. 2481–2495, 2017.

[34] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in Proc. 18th Int. Conf. Med. Image Comput. Comput.-Assist. Interv. Springer, 2015, pp. 234–241.

[35] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2017, pp. 2881–2890.

[36] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," IEEE Trans. Pattern Anal. Mach. Intell., vol. 40, no. 4, pp. 834–848, 2017.

[37] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in Proc. Eur. Conf. Comput. Vis., 2018, pp. 801–818.

[38] M. Xia, T. Wang, Y. Zhang, J. Liu, and Y. Xu, "Cloud/shadow segmentation based on global attention feature fusion residual network for remote sensing imagery," Int. J. Remote Sens., vol. 42, no. 6, pp. 2022–2045, 2021.

[39] C. Zhang, G. Li, and S. Du, "Multi-scale dense networks for hyperspectral remote sensing image classification," IEEE Trans. Geosci. Remote Sens., vol. 57, no. 11, pp. 9201–9222, 2019.

[40] Y. Liu, Y. Zhang, Y. Wang, and S. Mei, "Rethinking transformers for semantic segmentation of remote sensing images," IEEE Transactions on Geoscience and Remote Sensing, vol. 61, pp. 1–15, 2023.

[41] D.-H. Lee et al., "Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks," in Workshop Chall. Represent. Learn., ICML, vol. 3, no. 2. Atlanta, 2013, p. 896.

[42] J. Kim, Y. Hur, S. Park, E. Yang, S. J. Hwang, and J. Shin, "Distribution aligning refinery of pseudo-label for imbalanced semi-supervised learning," Adv. Neural Inf. Process. Syst., vol. 33, pp. 14 567–14 579, 2020.

[43] Y. Zhu, Z. Zhang, C. Wu, Z. Zhang, T. He, H. Zhang, R. Manmatha, M. Li, and A. J. Smola, "Improving semantic segmentation via efficient self-training," IEEE Trans. Pattern Anal. Mach. Intell., vol. 46, no. 3, pp. 1589–1602, 2021.

[44] X. Lu, L. Jiao, L. Li, F. Liu, X. Liu, S. Yang, Z. Feng, and P. Chen, "Weak-to-strong consistency learning for semisupervised image segmentation," IEEE Transactions on Geoscience and Remote Sensing, vol. 61, pp. 1–15, 2023.

[45] L. Lv and L. Zhang, "Advancing data-efficient exploitation for semi-supervised remote sensing images semantic segmentation," IEEE Transactions on Geoscience and Remote Sensing, 2024.

[46] M. Cai, H. Chen, T. Zhang, Y. Zhuang, and L. Chen, "Consistency regularization based on masked image modeling for semi-supervised remote sensing semantic segmentation," IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 2024.

[47] W. Huang, Y. Shi, Z. Xiong, and X. X. Zhu, "Decouple and weight semi-supervised semantic segmentation of remote sensing images," ISPRS Journal of Photogrammetry and Remote Sensing, vol. 212, pp. 13–26, 2024.

[48] Y. Ouali, C. Hudelot, and M. Tami, "Semi-supervised semantic segmentation with cross-consistency training," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2020, pp. 12 674–12 684.

[49] Z. Ke, D. Qiu, K. Li, Q. Yan, and R. W. Lau, "Guided collaborative training for pixel-wise semi-supervised learning," in Proc. Eur. Conf. Comput. Vis. Springer, 2020, pp. 429–445.
[50] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," Adv. Neural Inf. Process. Syst., vol. 30, 2017.
[51] Z. Zhao, S. Long, J. Pi, J. Wang, and L. Zhou, "Instance-specific and model-adaptive supervision for semi-supervised semantic segmentation," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2023, pp. 23 705–23 714.
[52] Z. Zhao, L. Yang, S. Long, J. Pi, L. Zhou, and J. Wang, "Augmentation matters: A simple-yet-effective approach to semi-supervised semantic segmentation," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2023, pp. 11 350–11 359.
[53] M. Sewell, "Ensemble learning," RN, vol. 11, no. 02, pp. 1–34, 2008.
[54] L. Breiman, "Bagging predictors," Mach. Learn., vol. 24, pp. 123–140, 1996.
[55] Y. Ren, L. Zhang, and P. N. Suganthan, "Ensemble classification and regression-recent developments, applications and future directions," IEEE Comput. Intell. Mag., vol. 11, no. 1, pp. 41–53, 2016.
[56] F. Rottensteiner, G. Sohn, J. Jung, M. Gerke, C. Baillard, S. Benitez, and U. Breitkopf, "The ISPRS benchmark on urban object classification and 3D building reconstruction," ISPRS Annals of the Photogrammetry, ISPRS Ann. photogramm., Remote Sens. Spat. Inf. Sciences, vol. 1, no. 1, pp. 293–298, 2012.
[57] C. Robinson, K. Malkin, N. Jojic, H. Chen, R. Qin, C. Xiao, M. Schmitt, P. Ghamisi, R. Hänsch, and N. Yokoya, "Global land-cover mapping with weak supervision: Outcome of the 2020 IEEE GRSS data fusion contest," IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens., vol. 14, pp. 3185–3199, 2021.
[58] Y. Liu, J. Yao, X. Lu, M. Xia, X. Wang, and Y. Liu, "RoadNet: Learning to comprehensively analyze road networks in complex urban scenes from high-resolution remotely sensed images," IEEE Trans. Geosci. Remote Sens., vol. 57, no. 4, pp. 2043–2056, 2018.
[59] V. Mnih, "Machine learning for aerial image labeling," Ph.D. dissertation, University of Toronto, 2013.
[60] M. Schmitt, L. H. Hughes, C. Qiu, and X. X. Zhu, "SEN12MS–a curated dataset of georeferenced multi-spectral sentinel-1/2 imagery for deep learning and data fusion," arXiv preprint arXiv:1906.07789, 2019.
[61] S. McIntosh-Smith, "GW4 Isambard," https://gw4.ac.uk/isambard/, 2014, accessed: 2023-10-1.
[62] K. Sohn, D. Berthelot, N. Carlini, Z. Zhang, H. Zhang, C. A. Raffel, E. D. Cubuk, A. Kurakin, and C.-L. Li, "FixMatch: Simplifying semi-supervised learning with consistency and confidence," Adv. Neural Inf. Process. Syst., vol. 33, pp. 596–608, 2020.
[63] L. Yang, L. Qi, L. Feng, W. Zhang, and Y. Shi, "Revisiting weak-to-strong consistency in semi-supervised semantic segmentation," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2023, pp. 7236–7246.