

# Semi-supervised PolSAR Image Change Detection using Similarity Matching

Lei Wang<sup>1\*</sup>, Lingmu Peng<sup>1</sup>, Hanyu Hong<sup>1</sup>, Shuwei Zhao<sup>1</sup>, Qiwen Lv<sup>1</sup>, Rong Gui<sup>2</sup>

<sup>1</sup> Hubei Key Laboratory of Optical Information and Pattern Recognition,  
Wuhan Institute of Technology, Wuhan 430205, Hubei, China

<sup>2</sup> School of Geosciences and Info-physics, Central South University, Changsha 410083, China

**Keywords:** PolSAR, Semi-supervised, Image change detection, Pseudo-labels, SimMatch.

## Abstract

The lack of precisely labeled data limits the development of supervised polarimetric synthetic aperture radar (PolSAR) image change detection. Therefore, semi-supervised deep learning methods have recently demonstrated their significant capability for PolSAR image change detection. Similarity Matching (SimMatch) improves the performance of semi-supervised learning tasks across different benchmark datasets and different settings. Introducing SimMatch into the field of PolSAR image change detection can improve the performance of semi-supervised PolSAR image change detection under limited labeled data conditions. Usually, semi-supervision solves the problem of insufficient labeled data by generating pseudo-labels. However, when the pseudo-label method is simply applied, the model will fit on the confident but wrong pseudo-labels, resulting in poor performance. SimMatch offers a solution by requiring the strongly augmented view to share the same semantic similarity (i.e. label prediction) and instance characteristics (i.e. similarity between instances) with a weak augmented view for more intrinsic feature matching. Besides, by using a labeled memory buffer, the two similarities can be isomorphically transformed with each other by introducing the aggregating and unfolding techniques. Therefore, the semantic and instance pseudo-labels can be mutually propagated, and then, the detection performance of the PolSAR image change detection is improved. Experimental results on real PolSAR datasets demonstrated that SimMatch is an effective semi-supervised PolSAR change detection method and its performance surpasses some well-known change detection methods. Compared to the fully-supervised algorithm CWNN, the semi-supervised SimMatch algorithm can improve accuracy by up to 14.4%.

## 1. Introduction

In the past several years, polarimetric synthetic aperture radar (PolSAR) data have become easily accessible due to more and more satellites launched for data collection (Zhang et al., 2021). Taking the Gaofen-3 satellite as an example, the resolution can reach 1 meter and the maximum width can reach 650 kilometers. Benefiting from artificial intelligence (AI), the methods for performing image change detection tasks on PolSAR data have advanced rapidly (Chen et al., 2020).

However, a large volume of labeled data is very expensive to collect in a real-world scenario. Learning with few labeled data has been a longstanding problem in the PolSAR image change detection research community. Among various methods, semi-supervised learning (SSL) (Chapelle et al., 2006) has recently demonstrated its significant capability for PolSAR image change detection.

A simple but very effective semi-supervised learning method is to adopt the classical two-stage training paradigm: pre-training plus fine-tuning. Such as ResNet (He et al., 2016), Vision Transformer (ViT) (Dosovitskiy et al., 2020), and Swin-T (Liu et al., 2021) are commonly pretrained in a supervised manner on a large-scale dataset, and then transfer the learned representation by fine-tuning the pretrained model with a few labeled samples.

Instead of separate two-stage pretraining and fine-tuning, current popular methods directly involve the labeled data in a joint feature learning paradigm with pseudo labeling (Lee, 2013) or consistency regularization (Sajjadi et al., 2016). These methods train semantic classifiers with labeled samples and use predicted distributions as pseudo labels for unlabeled samples. Pseudo-labels typically come from weakly augmented views or the average predicted values of strongly augmented views. However, when there are only very limited labeled data, they suffer severe "overconfidence" issues, i.e., the model will fit on the confident but wrong pseudo-labels, resulting in poor performance. We notice that matching the similarity relationships of both semantic and instance levels simultaneously is extremely beneficial to decouple their

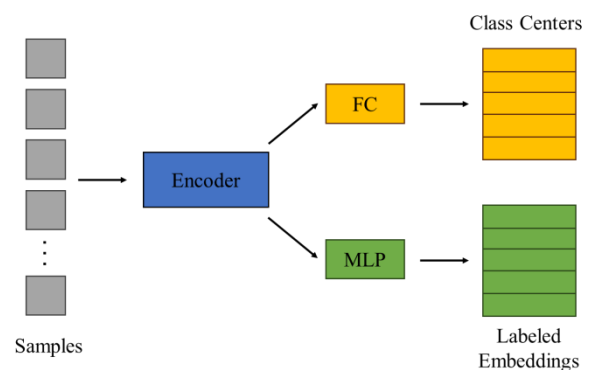


Figure 1. Consider the Fully-Connected layer vector as the semantic representation or class center of each category. SimMatch uses labeled memory buffers to fully utilize instance-level labels.

predictions as well as alleviate overfitting on noisy pseudo-labels.

In this paper, we introduce Similarity Matching (SimMatch) (Zheng et al., 2022) to improve the performance of Semi-supervised PolSAR image change detection across different benchmark datasets and different settings. SimMatch, which has been shown in Figure 1, offers a solution by requiring the strongly augmented view to share the same semantic similarity (i.e. label prediction) and instance characteristics (i.e. similarity between instances) with a weak augmented view for more intrinsic feature matching. Besides, by using a labeled memory buffer, the two similarities can be isomorphically transformed with each other by introducing the aggregating and unfolding techniques. Therefore, the semantic and instance pseudo-labels can be mutually propagated, and then, the detection

performance of the PolSAR image change is improved. We extended SimMatch from red-green-blue (RGB) to PolSAR image change detection, which is a novel semi-supervised learning framework that considers both semantic similarity and instance similarity, achieving better results than other methods.

## 2. Related Work

### 2.1 Semi-Supervised Learning.

Consistency regularization and entropy minimization methods are widely used in semi-supervised learning. Consistency regularization ensures that when perturbations are applied to the input or model, the model's response to the input remains consistent (Sajjadi et al., 2016). Regularization can be achieved through the simplest form of loss terms:

$$\|p_{model}(y|A(x);\theta) - p_{model}'(y|A(x);\theta)\|_2^2, \quad (1)$$

$A(x)$  is stochastic transformation, which can be domain-specific data augmentation (Sajjadi et al., 2016), random max pooling (Sajjadi et al., 2016), or adversarial transformation (Miyato et al., 2018). A further approach is to perturb the model  $p_{model}'$ , rather than the input. The perturbation can be adversarial perturbations to model parameters  $\theta$  (Zhang and Qi, 2020), or a time ensembling for the model at different time steps (Tarvainen and Valpola, 2017). Also, entropy minimization utilizes unlabeled data in an explicit bootstrapping manner, assigning pseudo labels to unlabeled data for joint training with labeled data. Different from prior works, MixMatch (Berthelot, 2019), ReMixMatch (Berthelot et al., 2019), and FixMatch (Sohn et al., 2017) combine the advantages of these two methods and propose a hybrid framework to leverage unlabeled data from two perspectives. Specifically, MixMatch adopts sharpened average predictions from multiple strongly augmented views as pseudo labels and further enhances them by using the MixUp trick (Berthelot, 2019). ReMixMatch inherited this idea and proposed Augmentation Anchoring to generate pseudo labels with weakly augmented views. It also introduces a distribution alignment strategy that encourages pseudo-label distribution to match the edge distribution of ground-truth class labels (Berthelot, 2019). FixMatch simplified unnecessary mechanisms and achieved optimal performance by retaining only high-confidence pseudo labels and their corresponding unlabeled data. In recent work, FlexMatch (Zhang et al., 2021) utilized inherent learning states to filter low-confidence labels class-wise as a further extension.

### 2.2 The self-training via pseudo labeling.

Pseudo-labels are artificial labels generated by the model itself for further training. Lee (2013) selected the class with the highest prediction probability of the model as the pseudo label in his semi-supervised deep learning network. The low-density separation assumption requires that when minimizing the entropy on pseudo labels, the decision boundaries between clusters of unlabeled samples are in the low-density region. Filtering pseudo labels based on confidence threshold (Sohn et al., 2020) is a simple but effective extension, which defines the confidence of a pseudo label as the highest probability it is considered to be any class. Filter out pseudo labels with confidence levels below the threshold, allowing us to focus more on labels that are away from the decision boundaries but have high confidence (low entropy).

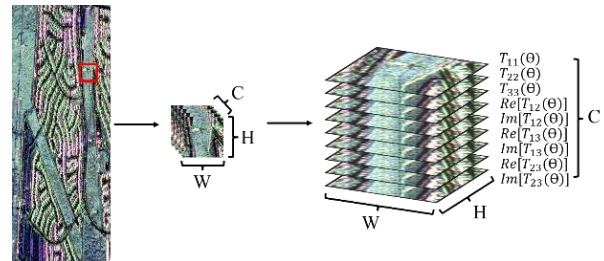


Figure 2. The input neighborhood window data of a pixel in a PolSAR image.  $C$  denotes the number of channels.  $H$  and  $W$  denote the height and width of the neighborhood window of a pixel.

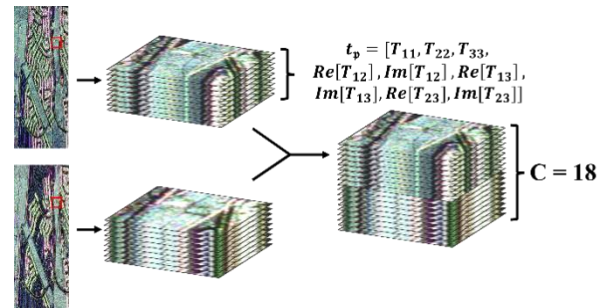


Figure 3. The input data formed by overlapping data of different periods.

The use of pseudo labels for self-training is an explicit classical method proposed more than a decade ago (Lee, 2013). In recent years, it has received increasing attention from various fields, such as semi-supervised learning (Cascante-Bonilla et al., 2021), fully-supervised learning (Radosavovic et al., 2018), and domain adaptation (Kumar et al., 2020). Especially in semi-supervised learning, it has been refocused in some computer vision tasks, such as image detection.

## 3. Method

This section first introduces the input data types of PolSAR images, and then provides a basic framework for semi-supervised learning with augmentation anchoring to delve into the main components of SimMatch.

### 3.1 The input data of SimMatch

Under horizontal and vertical polarization bases ( $H, V$ ), PolSAR can obtain full polarization information of targets, which can be characterized by polarization scattering matrices:

$$S = \begin{bmatrix} S_{HH} & S_{HV} \\ S_{VH} & S_{VV} \end{bmatrix}, \quad (2)$$

where  $H$  is the horizontally polarized emission or reception of electromagnetic wave signals, and  $V$  is the vertically polarized emission or reception of electromagnetic wave signals.

Under the assumption of reciprocity ( $S_{VH} = S_{HV}$ ) in a single station scenario, the commonly used polarization coherence matrix in polarimetric SAR information processing can be obtained:

$$T = \langle k_p k_p^H \rangle = \begin{bmatrix} T_{11} & T_{12} & T_{13} \\ T_{21} & T_{22} & T_{23} \\ T_{31} & T_{32} & T_{33} \end{bmatrix}, \quad (3)$$

$$k_p = \frac{1}{\sqrt{2}} [S_{HH} + S_{VV} - S_{HV} - S_{VH}]^T. \quad (4)$$

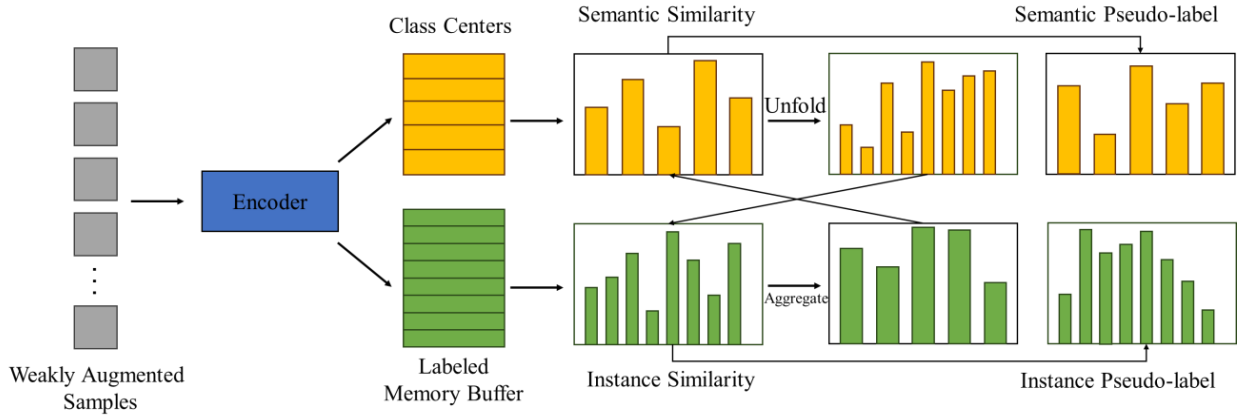


Figure 4. Overview of SimMatch pseudo label generation process. SimMatch first uses a semantic pseudo label and an instance pseudo label generated from weakly enhanced views to calculate the similarity between semantics and instances through the class center and tag embedding and then uses unfolding and aggregation operations to fuse these two similarities, ultimately obtaining the pseudo label.

In Eq. (3),  $k_p$  is the Pauli scattering vector, and  $(\cdot)^H$  represents conjugate transposition.

The elements in the  $T$  are complex numbers except for the diagonal ones. The polarimetric information of each pixel can be defined by a vector  $t_p$ .

$$t_p = [T_{11}, T_{22}, T_{33}, \text{Re}[T_{12}], \text{Im}[T_{12}], \text{Re}[T_{13}], \text{Im}[T_{13}], \text{Re}[T_{23}], \text{Im}[T_{23}]] \quad (5)$$

where  $T_{ij}$  are the elements in  $T$ .  $i, j = 1, 2, 3$ .  $\text{Re}[]$  and  $\text{Im}[]$  denote the real part and imaginary part of  $T_{ij}$ , respectively.

Typically, when classifying a pixel in a PolSAR image, the neighborhood window data of the pixel is also used as input, as shown in Figure 2. Let  $C$  denote the number of channels, also the number of elements in  $t_p$  and  $H$  and  $W$  denote the height and width of the neighborhood window data. Then the size of the input data is  $(C, H, W)$ .

The number of channels in a PolSAR image is 9. In order to detect change in the same pixel at different times, we overlay two PolSAR images with the same location but different phases, as shown in Figure 3. At this point, the number of channels for the input data is 18.

### 3.2 SimMatch semi-supervised learning framework

The semi-supervised image detection problem can be defined as follows. Randomly apply a weak augmentation function  $T_w(\cdot)$  to a batch of labeled data  $X = \{x_b: b \in (1, \dots, B)\}$  to obtain weakly augmented samples. Then, use the encoder  $F(\cdot)$  of the convolutional neural network to obtain feature information:  $h = f(T(x))$ , and map  $h_b$  to semantic similarity  $p = \phi(h)$  using the fully connected class prediction head  $\phi(\cdot)$ . Similarly, applying weak augmented  $T_w(\cdot)$  and strong augmented  $T_s(\cdot)$  [5, 6] to unlabeled data  $u = \{u_b: b \in (1, \dots, \mu B)\}$  randomly, the same processing is performed to obtain the semantic similarity for weak augmented sample  $p^w$  (pseudo label) and strong augmented sample  $p^s$ .

SimMatch also considers instance similarity, which encourages strong augmented views and weak augmented views to have similar similarity distributions. Assuming a nonlinear projection head  $g(\cdot)$  is mapped to a low dimensional embedding  $z_b = g(h_b)$ , based on augmented anchoring,  $z_b^w$  and  $z_b^s$  are used to represent the embeddings of weakly augmented views and strongly augmented views. For  $k$  weakly augmented

embeddings of different samples  $\{z_k: k \in (1, \dots, K)\}$ , the similarity function  $\text{sim}(\cdot)$  is used to calculate the similarity between  $z^w$  and the  $i$ -th instance. This similarity function represents the  $L_2$  normalization vector  $\text{sim}(u, v) = u^T v / \|u\| \|v\|$ . Process the obtained similarity in the softmax layer to obtain the similarity distribution, where  $t$  is the temperature parameter for adjusting the sharpness of the distribution:

$$q_i^w = \frac{\exp(\text{sim}(z_b^w, z_i) / t)}{\sum_{k=1}^K \exp(\text{sim}(z_b^w, z_k) / t)} \quad (6)$$

Similarly, calculating the similarity  $\text{sim}(z_b^s, z_i)$  between the strongly augmented views  $z^s$  and  $z_i$  yields a similarity distribution as follows:

$$q_i^s = \frac{\exp(\text{sim}(z_b^s, z_i) / t)}{\sum_{k=1}^K \exp(\text{sim}(z_b^s, z_k) / t)} \quad (7)$$

#### 3.2.1 Label Propagation through SimMatch

In the previous text, we considered instance-level consistency regularization, but due to the completely unsupervised instance pseudo-label  $q^w$ , it caused a great waste of labeled information. SimMatch improves the quality of pseudo-labels by utilizing labeled information on the instance level and a strategy that allows for interaction between semantic similarity and instance similarity.

All annotated examples are saved in a labeled memory buffer, as shown in Figure 4 (green branch). In Eq. (6) and Eq. (7) each  $z_k$  can be assigned to a specific class. The vector in  $\phi$  is interpreted as a "centered" class reference and the embeddings in the labeled memory buffer can be considered as a set of "single" class references.

By using a weakly augmented sample, the semantic similarity  $p^w \in \mathbf{R}^{1 \times L}$  and instance similarity  $q^w \in \mathbf{R}^{1 \times K}$  can be calculated first. (Note that since each class requires at least one sample,  $L$  is usually much smaller than  $K$ .) We need to **unfold**  $p^w$  into  $K$ -dimensional space (represented as  $p^{unfold}$ ) to calibrate  $q^w$ . By matching the corresponding semantic similarity embedded in each tag, this can be achieved:

$$p_i^{unfold} = p_j^w, \text{ where } \text{class}(q_j^w) = \text{class}(p_i^w), \quad (8)$$

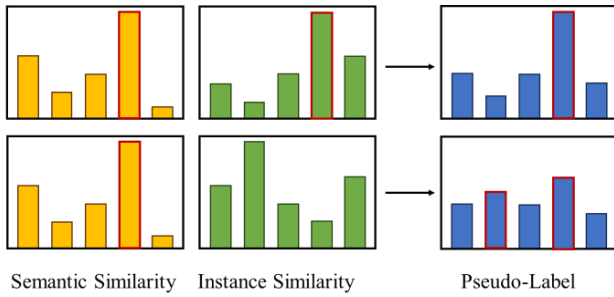


Figure 5. The label propagation. When the semantic and instance similarity are similar, the resulting pseudo labels will be clearer and when these two similarity points are different, the generated pseudo labels will be flatter.

among them,  $class(\cdot)$  is a function that returns the basic truth class. Specifically,  $class(q_j^w)$  represents the label of the  $j^{th}$  element in the memory buffer, and  $class(q_i^w)$  represents the  $i^{th}$  class. Then use  $p^{unfold}$  to **scale**  $q^w$  to regenerate the calibrated instance pseudo labels, which can be represented as:

$$\hat{q}_i = \frac{q_i^w p_i^{unfold}}{\sum_{k=1}^K q_k^w p_k^{unfold}}. \quad (9)$$

The pseudo label  $\hat{q}$  of the calibrated instance will replace the old one  $q^w$ . On the other hand, in order to use instance similarity to adjust semantic similarity, it is necessary to first **aggregate**  $q$  into the  $L$  dimensional space, represent it as  $q^{agg}$ , and then achieve sharing the same ground-truth labels by summing instance similarity:

$$q_i^{agg} = \sum_{j=0}^K 1(class(p_i^w) = class(q_j^w)) q_j^w. \quad (10)$$

The semantic pseudo labels readjusted by using  $q^{agg}$  **smoothing**  $p^w$  can be written as:

$$\hat{p}_i = \alpha p_i^w + (1-\alpha) q_i^{agg}, \quad (11)$$

among  $\alpha$  it is a hyper-parameter that controls the weight and instance information of semantics. Similarly, the adjusted semantic pseudo labels will replace the old one  $p_i^w$ . In this way, pseudo labels  $\hat{p}$  and  $\hat{q}$  will contain both semantic and instance-level information. As shown in Figure 5, if these two similarities are different, the result will be flatter and do not contain high probability values. On the other hand, when the similarity between semantics and instances is close, it means that these two distributions are consistent with each other's predictions, resulting in sharper results and high confidence for certain classes.

### 3.2.2 Efficient Memory Buffer

As mentioned earlier, SimMatch has a memory buffer to store feature embeddings and basic truth labels of labeled data. Specifically, SimMatch defines a feature memory buffer  $Q_f$  and a label memory buffer  $Q_l$ , where  $K$  is the number of labeled data and  $D$  is the embedding size. For  $Q_f$ , only one scalar needs to be stored for each label, and we can achieve aggregation and unfolding operations through the gather and scatter add functions in the deep learning library (Zheng et al., 2022).

In SimMatch, two different implementations were adopted for different buffer sizes. When  $K$  is large, **MoCo** (He et al.,

2020) is followed to utilize a student-teacher-based framework, represented as  $F_s$  and  $F_t$ .

In this case, the labeled data and strongly augmented data will be passed into  $F_s$ , while weakly augmented data will be fed into  $F_t$  to generate pseudo labels.

$$F_t \leftarrow m F_t + (1-m) F_s. \quad (12)$$

On the other hand, when  $K$  is small, a time integration strategy (French et al., 2017) is only needed to smooth the features in the memory buffer, which can be written as:

$$z_t \leftarrow m z_{t-1} + (1-m) z_t. \quad (13)$$

In this case, all samples will be directly passed to the same encoder.

### 3.3 Loss

As can be seen from the previous text, the labeled samples are directly optimized through cross-entropy loss with ground-truth labels:

$$L_s = \frac{1}{B} \sum H(y, p). \quad (14)$$

Unsupervised loss can be defined by the cross entropy between the semantic similarity for weak augmented sample  $p^w$  (pseudo label) and strong augmented sample  $p^s$ .

$$L_u = \frac{1}{\mu B} \sum 1(max DA(p^w) > \tau) H(DA(p^w), p^s), \quad (15)$$

next, we only retain the maximum class probability in the pseudo label that is greater than the confidence threshold  $\tau$  unlabeled samples (Sohn et al., 2020).  $DA(\cdot)$  represents a distribution alignment strategy (Berthelot et al., 2019), which balances the distribution of pseudo labels. We just need to maintain the moving-average of  $p_{avg}^w$  and use  $Normalize(p^w/p_{avg}^w)$  to adjust the current  $p^w$  (Li et al., 2021). Additionally, it should be noted that  $DA(p^w)$  will be used directly as a pseudo label instead of using the sharpened or hot version of  $p^w$ .

Besides, we achieve consistency regularization by minimizing the difference between  $q^s$  and  $q^w$ . Here, cross-entropy loss is used:

$$L_{in} = \frac{1}{\mu B} \sum H(q^w, q^s). \quad (16)$$

The overall training objective of our model is:

$$L_{overall} = L_s + \lambda_u L_u + \lambda_{in} L_{in}. \quad (17)$$

## 4. Experiments

### 4.1 Dataset

The dataset was generated by unmanned aerial vehicle synthetic aperture radar (UAVSAR) sensors capturing different regions of Los Angeles (LA), with each dataset consisting of two sets of images from April 23, 2009, and May 11, 2015. Among them, LA1 has a height of 786 pixels and a width of 300 pixels, and LA2 has a height of 766 pixels and a width of 300 pixels. Figure 6 shows the Pauli pseudo-color maps and their ground-truth (GT) maps of LA1 and LA2 at different times.

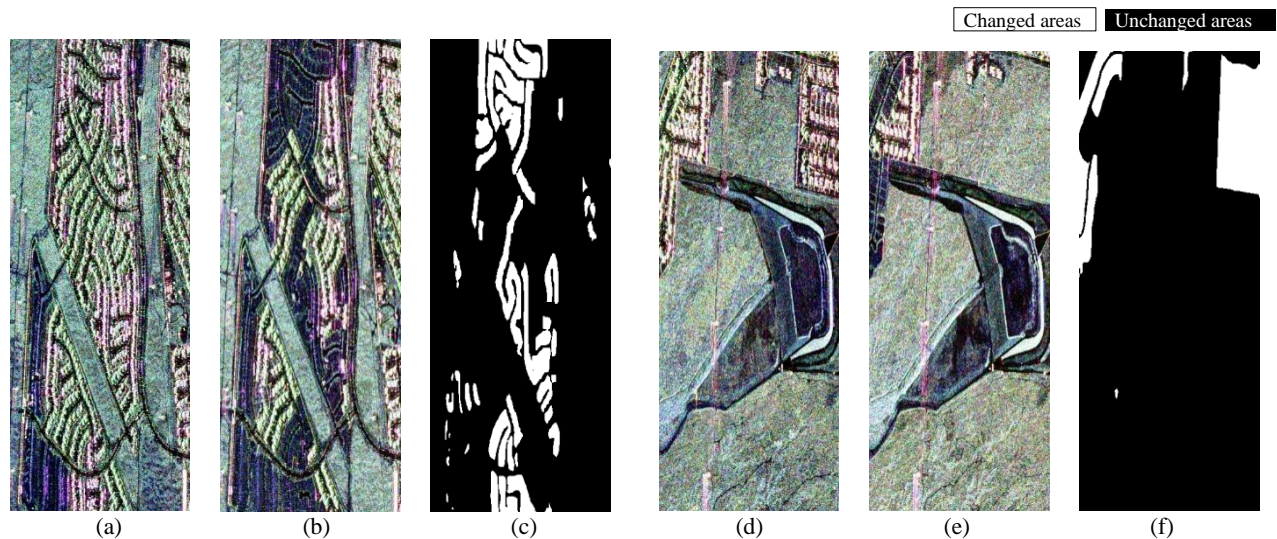


Figure 6. Pauli pseudo-color images of the UAVSAR images. Black denotes unchanged areas, and white denotes changed areas. (a)(b) Pauli pseudo-color images of UAVSAR LA1. (c) GT of LA1. (d)(e) Pauli pseudo-color images of UAVSAR LA2. (f) GT of LA2.

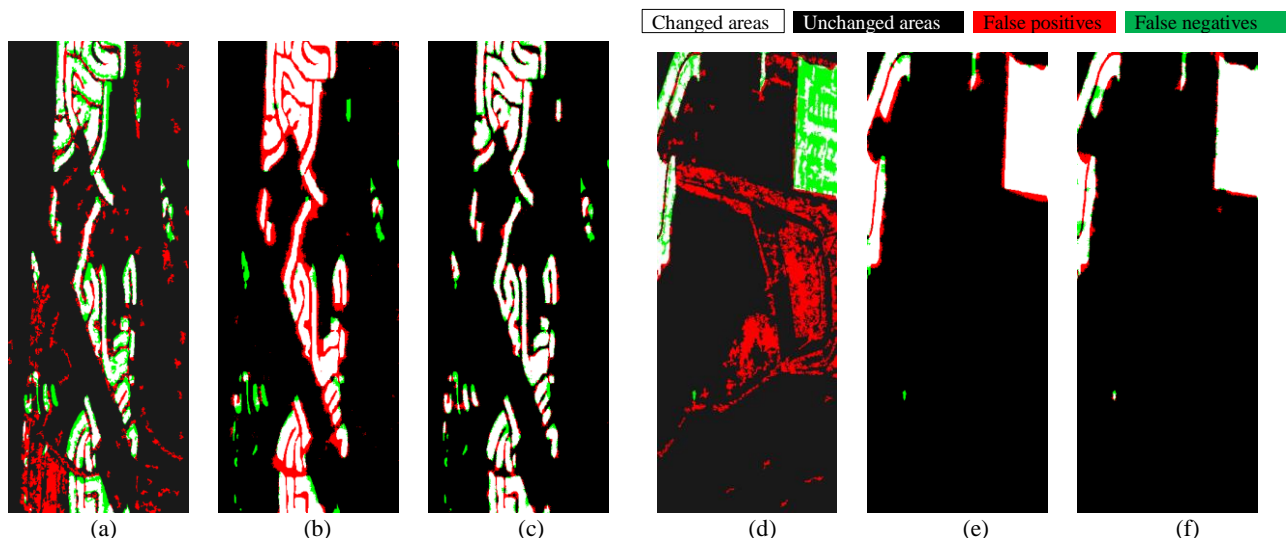


Figure 7. Change detection results on the UAVSAR LA1 dataset and the UAVSAR LA2 dataset. (a) CWNN on LA1. (b) SimMatch on LA1 (100 samples). (c) SimMatch on LA1 (3664 samples, 5%). (d) CWNN on LA2. (e) SimMatch on LA2 (100 samples). (f) SimMatch on LA2 (2318 samples, 5%).

Methods	Samples	OA(%)	Kappa	Precise	Recall	F1
CWNN	3664	88.17	0.579	0.601	0.706	0.649
<b>SimMatch</b>	<b>100</b>	<b>90.35</b>	<b>0.686</b>	<b>0.979</b>	<b>0.905</b>	<b>0.941</b>
SimMatch	3664	95.08	0.815	0.974	0.967	0.971

Table 1. The change detection performance evaluation on the UAVSAR LA1 detest.

Methods	Samples	OA(%)	Kappa	Precise	Recall	F1
CWNN	2318	83.38	0.282	0.300	0.487	0.371
<b>SimMatch</b>	<b>100</b>	<b>97.78</b>	<b>0.885</b>	<b>0.996</b>	<b>0.979</b>	<b>0.988</b>
SimMatch	2318	98.31	0.910	0.995	0.987	0.991

Table 2. The change detection performance evaluation on the UAVSAR LA2 detest.

The experiment uses 100 labeled change pixels as the number of training samples for semi-supervised change detection. For each dataset, we randomly select the same number of training samples representing both changed and unchanged types, with the remaining pixels used for performance evaluation. The OA, Kappa coefficient, precision, recall, and F1 score were used for the change detection performance evaluation (Lee, 2013). We

compare the well-known fully-supervised change detection method CWNN (Gao et al., 2019) with SimMatch: a SAR image sea ice change detection method based on a convolutional-wavelet neural network.

## 4.2 Results

The change detection results and performance evaluation on the UAVSAR LA1 dataset are shown in Figure 7 and Table 1. The CWNN achieved impressive results, but some unchanged areas were detected as changing areas, which affected the performance of change detection. The SimMatch algorithm proposed in this article achieved excellent performance on this dataset, with slightly better accuracy than CWNN even under the constraint of semi-supervised learning with small samples. Most of the changed and unchanged regions were correctly identified. The OA, Kappa coefficient, recall, and F1 scores are 2.18%, 0.107, 0.199, and 0.031 higher than those of CWNN, respectively.

Figure 7 and Table 2 show the change detection results of the LA2 dataset. The performance of CWNN is very low, with

many unchanged areas identified as changed areas, while the changed area in the upper right corner is almost not recognized.

The performance of the proposed SimMatch is also much better than CWNN in semi-supervised situations. Compared with CWNN, OA has improved by 14.4%, the Kappa coefficient has increased by 0.603, the recall has increased by 0.492, and the F1 score has increased by 0.617.

To further explore the optimal performance of SimMatch, we conducted a fully-supervised experiment on the UAVSAR dataset using 5% of labeled samples. We use 3664 samples on the LA1 dataset and 2318 samples on the LA2 dataset. SimMatch achieved excellent results on both datasets. The OA has reached its highest level with 95.08% and 98.31% respectively. The experiment further validated the excellent performance of SimMatch under fully-supervised learning.

The above experiments and analysis indicate that the SimMatch proposed in this paper can achieve good performance in PolSAR image change detection. Even with a small number of training samples, the performance surpasses advanced fully-supervised change detection methods. This framework considers both semantic-level and instance-level consistency regularization and has memory buffers to fully utilize instance-level data annotations, with strong feature extraction capabilities.

## 5. Conclusion

This paper introduces the new semi-supervised learning framework SimMatch into the field of SAR image change detection. In SimMatch, consistency regularization is applied simultaneously at the semantic-level and instance-level, associating semantic similarity with instance similarity. The two types of similarity can be propagated through unfolding and aggregation operations. At the same time, it also has a labeled memory buffer that can fully utilize the basic fact labels at the instance-level. Extensive experiments have shown that SimMatch can generate higher quality and more reliable matching targets with a small amount of labeled data. The change detection results on the UAVSAR dataset demonstrate the most advanced performance of semi-supervised learning.

## Acknowledgments

This work was supported by the Startup Foundation of the Wuhan Institute of Technology under Grant No. 21QD23.

## References

- Berthelot, D., Carlini, N., Goodfellow, I., Papernot, N., Oliver, A., Raffel, C. A., 2019: Mixmatch: A holistic approach to semi-supervised learning. *Advances in neural information processing systems*, 32.
- Berthelot, D., Carlini, N., Cubuk, E.D., Kurakin, A., Sohn, K., Zhang, H., Raffel, C., 2019: Remixmatch: Semi-supervised learning with distribution alignment and augmentation anchoring. *arXiv preprint arXiv:1911.09785*.
- Cascante-Bonilla, P., Tan, F., Qi, Y., Ordonez, V., 2021: Curriculum labeling: Revisiting pseudo-labeling for semi-supervised learning. In *Proceedings of the AAAI conference on artificial intelligence*, 35(8), 6912-6920.
- Chapelle, O., Chi, M., Zien, A., 2006: A continuation method for semi-supervised SVMs. In *Proceedings of the 23rd international conference on Machine learning*, 185-192.
- Cheng, G., Han, J. W., 2016: A survey on object detection in optical remote sensing images. *ISPRS journal of photogrammetry and remote sensing*, 117, 11-28.
- Cheng, G., Xie, X., Han, J., Guo, L., Xia, G., 2020: Remote sensing image scene classification meets deep learning: Challenges, methods, benchmarks, and opportunities. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13, 3735-3756.
- Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houshy, N., 2020: An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.
- French, G., Mackiewicz, M., Fisher, M., 2017: Self-ensembling for visual domain adaptation. *arXiv preprint arXiv:1706.05208*.
- Gao, F., Wang, X., Gao, Y., Dong, J., Wang, S., 2019: Sea ice change detection in SAR images based on convolutional-wavelet neural networks. *IEEE Geoscience and Remote Sensing Letters*, 16(8), 1240-1244.
- Han, J., Ding, J., Li, J., Xia, G., 2021: Align deep features for oriented object detection. *IEEE transactions on geoscience and remote sensing*, 60, 1-11.
- He, K., Fan, H., Wu, Y., Xie, S., Girshick, R., 2020: Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9726-9735.
- He, K., Zhang, X., Ren, S., Sun, J., 2017: Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1141-1149.
- Kumar, A., Ma, T., Liang, P., 2020: Understanding self-training for gradual domain adaptation. *International conference on machine learning*. PMLR, 5468-5479.
- Lee, D.H., 2013: Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. *Workshop on challenges in representation learning*. ICML, 3(2), 896.
- Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B., 2021: Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, 9992-10002.
- Li, J., Xiong, C., Hoi, S.C.H., 2021: Comatch: Semi-supervised learning with contrastive graph regularization. In *Proceedings of the IEEE/CVF international conference on computer vision*, 9475-9484.
- Lu, X., Sun, X., Diao, W., Feng, Y., Wang, P., Fu, K., 2021: LIL: Lightweight incremental learning approach through feature transfer for remote sensing image scene classification. *IEEE Transactions on Geoscience and Remote Sensing*, 60, 1-20.
- Miyato, T., Maeda, S., Koyama, M., Ishii, S., 2018: Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 41(8), 1979-1993.

- Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimeshain, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S., 2019: Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.
- Prakash, V.J., Nithya, D.L.M., 2014: A survey on semi-supervised learning techniques. *arXiv preprint arXiv:1402.4645*.
- Radosavovic, I., Dollár, P., Girshick, R., Gkioxari, G., He, K., 2018: Data distillation: Towards omni-supervised learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4119-4128.
- Sajjadi, M., Javanmardi, M., Tasdizen, T., 2016: Regularization with stochastic transformations and perturbations for deep semi-supervised learning. *Advances in neural information processing systems*, 29.
- Sun, X., Wang, P., Yan, Z., Xu, F., Wang, R., Diao, W., Chen, J., Li, J., Feng, Y., Xu, T., Weinmann, M., Hinz, S., Wang, C., Fu, K., 2022: FAIRIM: A benchmark dataset for fine-grained object recognition in high-resolution remote sensing imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 184, 116-130.
- Sun, X., Wang, P., Wang, C., Liu, Y., Fu, K., 2021: PBNNet: Part-based convolutional neural network for complex composite object detection in remote sensing imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 173, 50-65.
- Sohn, K., Berthelot, D., Carlini, N., Zhang, Z., Zhang, H., Raffel, C.A., Cubuk, E.D., Kurakin, A., Li, C. L., 2020: Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *Advances in neural information processing systems*, 33, 596-608.
- Tarvainen, A., Valpola, H., 2017: Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Advances in neural information processing systems*, 30.
- Van Engelen, J.E., Hoos, H.H., 2020: A survey on semi-supervised learning. *Machine learning*, 109(2), 373-440.
- Zhang, B., Wang, Y., Hou, W., Wu, H., Wang, J., Okumura, M., Shinozaki, T., 2021: Flexmatch: Boosting semi-supervised learning with curriculum pseudo labelling. *Advances in Neural Information Processing Systems*, 34, 18408-18419.
- Zhang, Y., Li, W., Tao, R., Peng, J., Du, Q., Cai, Z., 2021: Cross-scene hyperspectral image classification with discriminative cooperative alignment. *IEEE Transactions on Geoscience and Remote Sensing*, 59(11), 9646-9660.
- Zhang, L., Qi, G.J., 2020: Wcp: Worst-case perturbations for semi-supervised deep learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3911-3920.
- Zhao, X., Tao, R., Li, W., Li, H., Du, Q., Liao, W., Philips, W., 2020: Joint classification of hyperspectral and LiDAR data using hierarchical random walk and deep CNN architecture. *IEEE Transactions on Geoscience and Remote Sensing*, 58(10), 7355-7370.
- Zheng, M., You, S., Huang, L., Wang, F., Qian, Xu, C., 2022: SimMatch: Semi-supervised Learning with Similarity Matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14451-14461.
- Zhu, X., Goldberg, A.B., 2009: Introduction to semi-supervised learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 3(1), 130.