Edge Knowledge Distillation Guided Lightweight Change Detection Network

Tingyu Ji[1,2†], Yixin Chen[3,4†], Ruiqian Zhang[1*], Xiaogang Ning[1], Xiao Huang[5], Hanchao Zhang[1,7], Weibin Ma[1], Chunquan Cheng[1], Jiaming Wang[6]

[1] State Key Laboratory of Spatial Datum, Chinese Academy of Surveying and Mapping, Beijing, China - jity1215@163.com; zhangrq@casm.ac.cn; ningxg@casm.ac.cn; zhanghc@casm.ac.cn; 2093535923@qq.com
[2] College of Geodesy and Geomatics, Shandong University of Science and Technology, Qingdao, China
[3] Sichuan Institute of Land Science and Technology (Sichuan Center of Satellite Application Technology), Chengdu, China - chenyx0326@163.com
[4] Key Laboratory of Investigation, Monitoring, Protection and Utilization for Cultivated Land Resources, MNR, Chengdu, China
[5] Department of Environmental Sciences, Emory University, Atlanta, GA, USA - xiao.huang2@emory.edu
[6] Hubei Key Laboratory of Intelligent Robot, Wuhan Institute of Technology, Wuhan, China - wjmecho@wit.edu.cn
[7] Joint Laboratory of Spatial Intelligent Perception and Large Model Application

Keywords: Change Detection, Knowledge Distillation, Remote Sensing.

Abstract

Deep-learning methods dominate remote-sensing change detection (CD), yet state-of-the-art models remain parameter-heavy and struggle with crisp boundaries, limiting their use on edge devices. We present LEDGNet, a Lightweight, Edge-knowledge-Distillation-Guided CD Network, that reconciles accuracy, boundary fidelity, and efficiency. LEDGNet integrates three purpose-built components: 1) an Edge Distillation Module that mines multi-scale boundary cues from a high-capacity teacher and transfers them to a compact student through an edge-aware loss; 2) StarLite, a depth-wise separable encoder that preserves fine spatial detail while minimizing floating-point operations; and 3) LiteDecoder, an inexpensive feature-fusion head that restores full resolution without bulky up-sampling. This design halves the parameters and inference time of mainstream fine-grained CD networks while enhancing edge sharpness. On the CDD and LEVIR-CD benchmarks, LEDGNet achieves competitive F1 performance while maintaining a compact footprint of 20.58 M parameters and 35.18 G FLOPs. With an inference time of 255 ms, it strikes a balance between resource consumption and detection efficiency, making it well-suited for high-efficiency remote sensing monitoring.

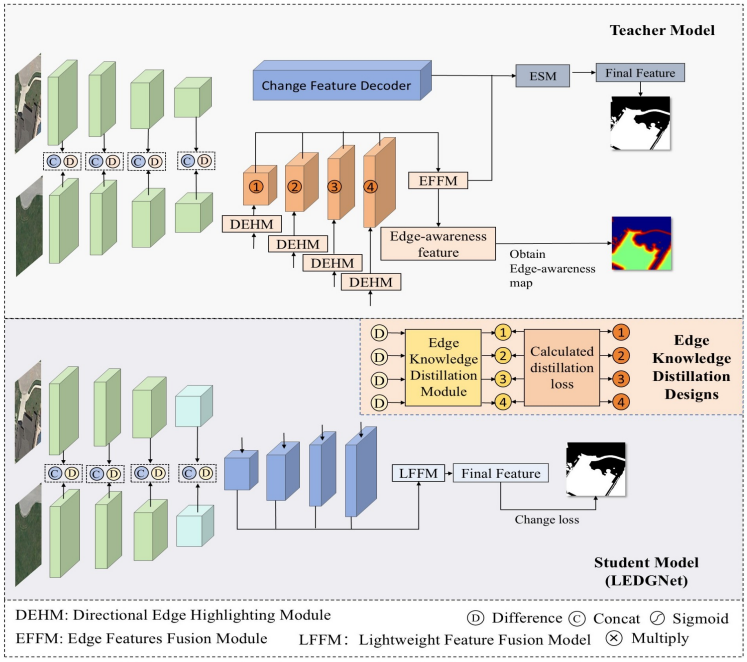
1. Introduction

As rapid population growth and intensified economic activity reshape Earth’s surface, land-cover patterns are evolving at an unprecedented pace. Timely, high-fidelity identification of both land-cover classes and their dynamics is therefore vital for safeguarding ecosystem health and socio-economic stability. Remote sensing change detection (CD) addresses this need by analyzing multi-temporal imagery of the same geographic region to discover, characterize, and interpret surface changes (Zhu et al., 2022). The escalating resolution and sheer volume of contemporary satellite data, however, exceed the capacity of traditional visual inspection and classical machine-learning pipelines, which lack the speed and scalability required for large-area, high-precision, efficient monitoring. In contrast, deep neural networks can learn discriminative, multi-level spatio-temporal representations end-to-end, obviating hand-crafted features and now constitute the mainstream solution for high-resolution CD (Carion et al., 2020).To boost accuracy, recent methods embed fine-grained feature extractors and sophisticated enhancement modules, yet these advances typically inflate parameter counts and FLOPs, complicating deployment on mobile or resource-constrained platforms. Consequently, research has pivoted toward lightweight CD models that balance detection fidelity with computational efficiency.

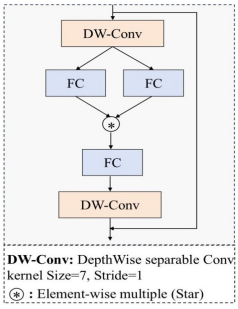
Lightweighting strategies fall into two broad categories: 1) architectural redesign and 2) post-hoc compression and acceleration. Architectural work centres on substituting heavy backbones with streamlined alternatives such as MobileNet (Howard et al., 2017) and ShuffleNet (Zhang et al., 2018). For instance, A2Net (Li et al., 2023) and the encoder of Liu et al. (Liu et al., 2025) leverage depthwise separable convolutions to slash computation while maintaining accuracy, whereas RFA (You et al., 2024)and ELW CDNet (Liu et al., 2023a) introduce attentive, multi-scale fusion modules that preserve semantic richness without excessive complexity. Compression approaches, notably pruning and knowledge distillation (KD), further reduce redundancy. Yang et al. (Yang et al., 2022) prune superfluous connections; Lei et al. (Lei et al., 2024) streamline CNN–MLP hybrids by removing skip links; and KD frameworks transfer the knowledge of large teachers to compact students, as in Wang et al.’s prototype-comparison and channel-space distillation scheme (Wang et al., 2024). Despite these advances, existing lightweight models still suffer from diminished feature-representation capacity, blurry boundaries, and poor small-object sensitivity.

Therefore, to address the above problems, we propose the Edge Knowledge Distillation Guided Lightweight Change Detection Network (LEDGNet). LEDGNet effectively balances network feature extraction and detail feature perception capabilities with model lightweight performance while maintaining high CD accuracy. This is achieved through the design of a lightweight network structure and an innovative edge knowledge distillation approach. Specifically, the network consists of a lightweight siamese encoder, a lightweight feature fusion decoder(LiteDecoder), and an edge knowledge distillation module(EDM). The siamese encoder uses convolutional substitution and computational optimization based on the ResNet18

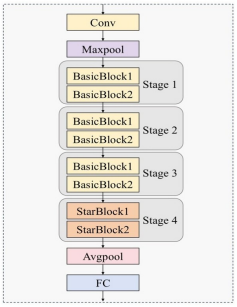
† These authors contributed equally to this work
* Corresponding author



(a) Teacher-Student Edge Knowledge Distillation Framework.



(b)Structure of the StarBlock.



(c) Structure of the StarLite.

Figure 1. Overall framework and components of StarLite.

backbone network to effectively reduce computational overhead while maintaining detection accuracy. The LiteDecoder efficiently fuses features under the premise of inter-layer information interaction and feature correlation, significantly reducing network computation and inference time. The EDM receives edge knowledge from the teacher model while retaining the ability to perceive fine-grained change edge features.

Our contribution can be summarized as follows:

(1) We design a teacher–student edge-knowledge-distillation framework in which the EDM and an edge-distillation loss enable hierarchical transfer and reuse of boundary features, substantially lowering model complexity while retaining accuracy.

(2) We lighten the encoder by replacing the final ResNet-18 BasicBlock with a StarBlock that enhances non-linear mapping, and by substituting standard convolutions with grouped convolutions, thereby cutting parameters without sacrificing representational power.

(3) We propose LiteDecoder with MGFRM, which synergizes grouped feature attention and multi-scale reshaping to emphasize informative context, suppress redundancy, and preserve small-target cues during fusion.

(4) Experiments on CDD and LEVIR-CD confirm that LEDGNet achieves competitive F1 scores, superior edge quality, and robust small-change recognition while realizing significant reductions in parameters, computation, and inference time, it establishes a highly efficient framework well-suited for potential deployment in resource-constrained remote sensing scenarios.

The remainder of this paper is organized as follows: Section 2 details the proposed method; Section 3 describes datasets, parameter setting, metrics, baselines, and results; Section 4 concludes the study.

2. Methodology

Since the methodology of this study is based on a teacher-student edge knowledge distillation framework, we will firstly give a brief introduction to the overall structure of this framework. Secondly, we will introduce the LEDGNet proposed by us, which is the student model of this framework. Subsequently, we will detail the key components of the implementation of the edge knowledge distillation strategy: the edge knowledge distillation module of the student model, the edge-aware features of the teacher model, and the edge distillation loss.

2.1 Teacher-Student Edge Knowledge Distillation Framework

Figure 1 (a) illustrates the teacher-student edge knowledge distillation framework, whose core architecture consists of two parts: the teacher model and the student model. The teacher model (ESMII-Net (Chen et al., 2025)) comprises Siamese encoders, a change feature decoder, an edge aware decoder, and an Edge-Synergy Module, which is able to extract both hierarchical change features and change edge-aware features, and has an excellent performance in change edge awareness, which is a refined CD network. The teacher model adopts the full-parameter ESMII-Net model in order to fully continue its powerful feature representation capability, while the student model consists of a lightweight siamese encoder, a LiteDecoder, and an EDM.

2.2 Lightweight Feature Extraction Backbone

The feature extraction backbone of the student model in this framework is aligned with that of the teacher model, ESMII-Net, both based on the classic ResNet18 architecture, but with lightweight improvements made on this basis. Specifically, the network replaces the last two BasicBlocks of ResNet18 with StarBlock, allowing the StarLite to reduce computational complexity while maintaining feature extraction capabilities. The StarBlock is inspired by StarNet (Ma et al., 2024b) and its structure is shown in Figure 1 (b).

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-XI-3-2026-133-2026 | © Author(s) 2026. CC BY 4.0 License.

The calculation process of StarBlock is as follows: first, $F_1 = \text{DWConv}_1(F)$, where DWConv_1 is the first depth-separable convolution (kernel size 7, stride 1). Subsequently, the feature F_1 is passed through two different fully connected layers to obtain $F_2 = \text{FC}_1(F_1)$ and $F_3 = \text{FC}_2(F_1)$. These two features are then multiplied element-wise: $F_{\text{mul}} = F_2 \times F_3$. This result is passed through another fully connected layer and a depth-separable convolution: $F_{\text{ful}} = \text{DWConv}_2(\text{FC}_2(F_{\text{mul}}))$. Finally, a residual connection is added to the initial features: $F_{\text{final}} = F + F_{\text{ful}}$. This module introduces a star operation (element-wise multiplication) and a grouped convolution mechanism. Previous studies have shown that element-wise multiplication in neural networks can aggregate features more effectively than summation. Meanwhile, it can improve model accuracy while maintaining low latency, few parameters, and low complexity. Therefore, this study introduces it into the feature extraction stage of LEDGNet to constitute the StarLite (Figure 1 (c)).

2.3 Lightweight Feature Fusion Decoder

Currently there are two mainstream feature fusion schemes exist in CD: hierarchical upsampling with shallow-deep concatenation, and direct upsampling to a unified scale. While effective for multi-scale feature integration, both involve heavy convolutions and high computation overhead. A careful trade-off between fusion performance and efficiency is thus essential.

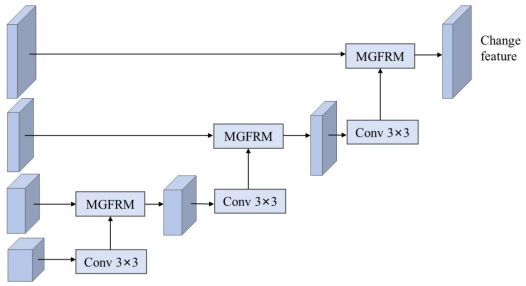


Figure 2. Structure of the LiteDecoder.

In order to strike a balance between the effectiveness of feature fusion and computational efficiency, this study designs the LiteDecoder (Figure 2) in the student model. In order to fully ensure the interactivity of the interlayer information, therefore the layer-wise upsampling is still adopted in the interlayer feature fusion. Meanwhile, a novel lightweight fusion module is designed in this study for feature fusion in the neighboring layers, which is named MGFRM in this study because the module mainly relies on the grouped feature attention mechanism with the multiscale feature reshaping strategy.

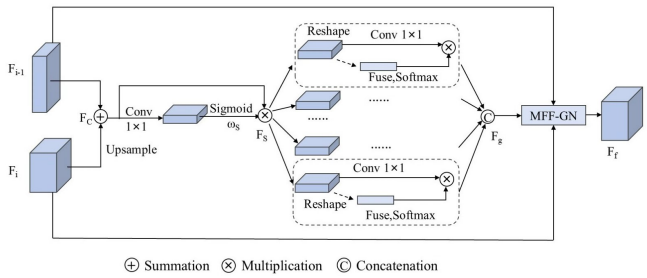


Figure 3. Structure of the Grouped Attention Module.

The core module of the decoder, MGFRM, which is used to fuse the interlayer information to obtain an effective representation of the change features. In order to enhance the semantic characterization of features, this study firstly designed the grouped attention module (Figure 3) for obtaining global information and enhancing the correlation between fused features. First, the change features F_i and F_{i-1} of the neighboring layers are input, F_i is twice upsampled, and then a simple feature fusion of the two features is performed using element-wise addition, a process denoted as:

$$F_c = 2 \times \text{Upsample}(F_i) + F_{i-1} \tag{1}$$

In order to optimize the features and obtain more effective context-aware information, the features are compressed to a single channel using 1×1 convolution, after which a Sigmoid function is used to generate the spatial attention weights ω_s, which in turn generates the features F_s that contain the spatial information, and the above process can be expressed as:

$$\omega_s = \text{Sigmoid}(F_c) \tag{2}$$

$$F_s = F_c \times \omega_s \tag{3}$$

where “×” denotes element-wise multiplication.

To enhance the correlation between neighboring features, this study divides the spatially aggregated features into n groups along the channel dimension and performs feature interactions on a per-group basis. Specifically, a convolution module is used to refine the feature information of neighboring channels in each group of $[F_s]^g_{n=i} \in \mathbb{R}^{C\times H\times W}$. The global features of different channels in $[F_s]^g_{n=i}$ are transformed to generate an attention mask ω_g that captures the correlation of features between channels. This mask ω_g is then applied to refine the features. Finally, the features in each group are connected to form aggregated, highly correlated neighboring features $F_g \in \mathbb{R}^{C\times H\times W}$. The entire calculation process is as follows:

$$F_{gi} = \text{Softmax}\left(\Phi\left([F_s]^g_{n=i}\right)\right) \times \Psi\left([F_s]^g_{n=i}\right) \tag{4}$$

$$F_g = F_{g1} \cup F_{g2} \cup \cdots \cup F_{gi} \tag{5}$$

Finally, the grouped aggregated features F_g are embedded into a normalization layer with multiple layers of original feature fusion. The features F_g are normalized using their mean and standard deviation to incorporate more spatial location information from smaller targets. The feature F_f with strong feature correlation and rich spatial information is obtained through multi-layer original feature fusion, which can be expressed as:

$$F_f = \frac{F_g - \text{mean}(F_i + F_{i-1})}{\text{std}(F_i + F_{i-1})} \tag{6}$$

where $mean$ and std represent mean and standard deviation operations. This approach makes full use of the semantic information of neighboring layers and extracts the relevant features of different channels, thus enhancing the overall feature representation.

In addition, to reduce the fusion and extraction of irrelevant features and minimize the loss of valid information in the deep network, this study designs a Multi-Scale Feature Reshaping Module (Figure 4). The purpose of reshaping features is to separate and process independently the rich information contained in the feature maps of different stages of the backbone network from the weaker information, which maximizes the retention of rich features and reduces the utilization of computational resources.

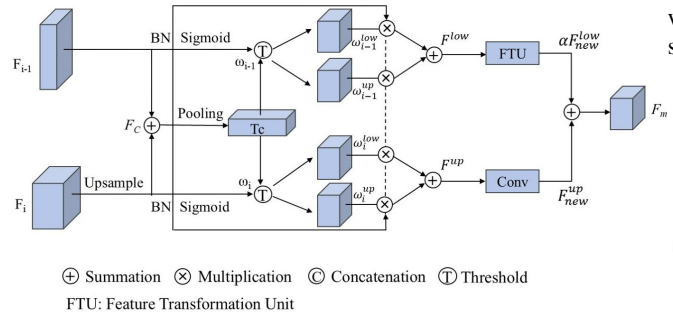


Figure 4. Structure of the Multi-Scale Feature Reshaping Module.

As previously mentioned, intermediate features F_c can be obtained from hierarchical features F_i and F_{i-1} by operations such as upsampling, convolution, element-wise addition, etc. Next, average pooling and Sigmoid function are applied to generate the information weights on each channel as the feature weight threshold T_c, and this process can be expressed as:

$$T_c = \text{Sigmoid}\left(\text{Avg}(F_c)\right) \tag{7}$$

where Avg represents the average pooling operation. The single-layer features F_i and F_{i-1} are processed by batch normalization (BN), respectively, and activated by a Sigmoid function that generates unique weight information for each spatial location, where $\omega_i \in \mathbb{R}^{C\times H\times W}$ and $\omega_{i-1} \in \mathbb{R}^{C\times H\times W}$ represent the attention weights of the i-th and (i-1)-th layers, respectively:

$$\omega_i = \text{Sigmoid}\left(\text{BN}(F_i)\right) \tag{8}$$

$$\omega_{i-1} = \text{Sigmoid}\left(\text{BN}(F_{i-1})\right) \tag{9}$$

Next, the weight information ω_i and ω_{i-1} from different stages are compared with the feature weight threshold T_c to obtain an attention map capturing the strength of spatial information. Subsequently, strong and weak features from different layers are aggregated to obtain rich and weak features, respectively:

$$\left(\omega_i^{up}, \omega_i^{low}\right) = \text{Threshold}(\omega_i, T_c) \tag{10}$$

$$\left(\omega_{i-1}^{up}, \omega_{i-1}^{low}\right) = \text{Threshold}(\omega_{i-1}, T_c) \tag{11}$$

where $Threshold$ is the threshold for separating strong and weak information.

After that, the strong attention maps ω_i^{up} and ω_{i-1}^{up} are mapped onto the feature F_c respectively, and then these two feature parts are fused to generate rich features. Similarly, the weak attention map is mapped onto F_c to generate weak features. The whole computation process is shown below:

$$F^{up} = (\omega_i^{up} \times F_c) + (\omega_{i-1}^{up} \times F_c) \tag{12}$$

$$F^{low} = (\omega_i^{low} \times F_c) + (\omega_{i-1}^{low} \times F_c) \tag{13}$$

where $F^{up} \in \mathbb{R}^{C\times H\times W}$ denotes rich features generated by reconstruction and $F^{low} \in \mathbb{R}^{C\times H\times W}$ denotes weak ones.

Thereafter the features F^{up} and F^{low} are transformed respectively. For the rich features, 1×1 convolution is used to generate the feature mapping F_{new}^{up} showing more detailed information. For weak features, F^{low} is fed into the Feature Transformation Unit (FTU, Figure 5), which aims to generate feature mappings with richer semantic information using less computational resources.

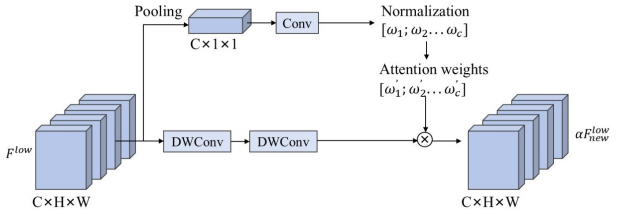


Figure 5. Structure of the FTU.

Subsequently, depth-separable convolution is used to reduce computational complexity and the number of parameters. Since depth-separable convolution disrupts the information flow between channels, feature modulation needs to be generated between channels. After depth-separated convolutional operations, this study performs weighted mapping to enhance the information flow between channels. The weighted feature α is processed by adaptive average pooling and convolutional layers, represented as:

$$\alpha = \text{Softmax}\left(\Gamma\left(\text{A}(F^{low})\right)\right) \tag{14}$$

where Γ represents the convolutional transformation layer, and A represents the adaptive average pooling layer.

Finally the feature F_{new}^{low} processed through the feature transformation unit is merged with the feature map F_{new}^{up} showing more detailed information in order to generate the feature F_m. The feature contains detailed and cross-channel information exchange. The computation of F_m is shown below:

$$F_m = \alpha F_{new}^{low} + F_{new}^{up} \tag{15}$$

In summary, this study uses a multi-scale feature reshaping module to merge features from two different layers together, thus obtaining richer features with more detail while reducing the use of computational resources. This approach allows specific transformations to be performed on individual features, thus minimizing the generation of redundant features.

After going through the Grouped Attention Module and the Multi-Scale Feature Reshaping Module, we sum up the features obtained from the above two parts to obtain the output features of this layer, which is represented by the formula:

$$F_{out} = F_m + F_f \tag{16}$$

After passing the neighboring layer features sequentially through the MGFRM, the fused CD output features are obtained and eventually participate in the loss calculation.

2.4 Edge Knowledge Distillation Module

Existing knowledge distillation methods can be divided into target distillation and feature distillation (Gou et al., 2021). In this paper, we innovatively adopt the edge knowledge distillation method based on feature distillation, by designing the EDM in the student model to obtain the hierarchical edge features, and corresponding them one-to-one with the hierarchical edge-aware features of the teacher model, constructing a multi-level edge knowledge transfer channel between the teacher model and the student model, so as to solve the problem of insufficient expression of edge features in the lightweight student model.

Method	Overall accuracy evaluation				Only for small objects			
	P(%)	R(%)	F1(%)	IoU(%)	P(%)	R(%)	F1(%)	IoU(%)
FC_Siam_diff	82.35	73.94	77.92	63.83	38.40	54.84	45.17	29.18
SNUNet	**96.29**	95.29	**95.78**	**91.91**	78.43	79.72	79.07	65.39
ChangeFormer	93.52	87.57	90.45	82.57	69.60	47.44	56.42	39.30
BIT-CD	95.50	90.78	93.07	87.05	80.71	53.26	64.17	47.25
AMT_Net	94.55	93.80	94.17	88.98	**88.16**	66.01	75.48	60.63
STADE-CDNet	95.48	92.73	94.08	88.83	71.04	76.91	73.85	58.55
FTAN	95.86	95.62	95.74	91.82	**86.24**	**82.75**	**84.45**	**73.10**
HCGMNet	93.56	**96.21**	94.87	90.23	83.77	78.98	81.30	68.49
CGNet	93.47	**95.86**	94.65	89.84	82.30	78.90	80.56	67.45
LEDGNet	**96.08**	95.57	**95.82**	**91.98**	85.68	**82.06**	**83.83**	**72.16**

*Red highlights the best results, and blue highlights the second-best results.

Table 1. Comparative Experiment Results on CDD Dataset

Method	Overall accuracy evaluation				Only for small objects			
	P(%)	R(%)	F1(%)	IoU(%)	P(%)	R(%)	F1(%)	IoU(%)
FC_Siam_diff	88.51	87.96	88.24	78.95	65.92	89.56	75.95	61.23
SNUNet	90.67	89.58	90.12	82.02	80.99	89.13	84.87	73.71
ChangeFormer	90.41	89.31	89.86	81.58	87.33	86.83	87.08	77.11
BIT-CD	90.76	87.95	89.34	80.73	**91.83**	86.17	88.91	80.04
AMT_Net	88.36	**92.06**	90.17	82.10	88.73	89.23	88.98	80.15
STADE-CDNet	**92.43**	88.70	90.53	82.69	88.24	87.95	88.10	78.73
FTAN	91.56	90.54	91.05	83.57	**91.73**	89.46	**90.58**	**82.79**
HCGMNet	92.05	90.12	91.07	83.61	86.33	**91.93**	89.05	82.25
CGNet	**93.30**	90.71	**91.99**	**85.16**	91.46	**92.30**	**91.88**	**84.97**
LEDGNet	91.51	**91.33**	**91.41**	**84.19**	90.69	90.04	90.36	82.42

* Red highlights the best results, and blue highlights the second-best results.

Table 2. Comparative Experiment Results on LEVIR-CD Dataset

The EDM consists of two parts: Sobel gradient computation and Gaussian blur. The input feature is the difference (F_{diff}) between the hierarchical features of the bi-temporal images (F_1 and F_2).

The Sobel operator computes the horizontal and vertical gradients of the feature map. The horizontal and vertical gradients are represented as follows:

$$G_x = F_{\text{diff}} \times S_x \qquad G_y = F_{\text{diff}} \times S_y \tag{17}$$

$$S_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \qquad S_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \tag{18}$$

S_x and S_y represent the horizontal and vertical gradient convolution kernels, respectively.

The gradient magnitude is calculated by taking the square root of the sum of squares of the horizontal and vertical gradients:

$$G = \sqrt{G_x^2 + G_y^2} \tag{19}$$

To smooth the gradient magnitude and reduce noise, the gradient magnitude G is subjected to Gaussian blurring. Gaussian blurring is implemented through a convolution operation, with the formula:

$$K_{\text{Gaussian}}(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \tag{20}$$

$$F_{\text{edge}} = G \times K_{\text{Gaussian}}(x, y) \tag{21}$$

where σ is the standard deviation of the Gaussian kernel, which controls the degree of fuzziness, and F_{edge} is the hierarchical edge feature of the student model.

2.5 Loss Function

The loss function of the LEDGNet is divided into a change detection loss and our core edge distillation loss. The latter is designed to supervise the EDM in LEDGNet, forcing it to acquire accurate edge knowledge from the teacher model's features.

(1)**Edge knowledge distillation loss**: This is a composite loss containing MSE Loss and Gradient Loss. The MSE Loss aligns the global feature distribution, defined as $\text{MSELoss} = \frac{1}{N}\sum_{i=1}^{N} \|F_t^{(i)} - F_s^{(i)}\|_2^2$, where $F_t^{(i)}$ and $F_s^{(i)}$ represent the feature values from the teacher and student models, and N is the total number of features. The Gradient Loss is used to align the local structural information of edge features:

$$\begin{aligned} \text{GradientLoss} = \frac{1}{N}\sum_{i=1}^{N} \Big(&\|G_x(F_t^{(i)}) - G_x(F_s^{(i)})\| + \\ &\|G_y(F_t^{(i)}) - G_y(F_s^{(i)})\| \Big) \end{aligned} \tag{22}$$

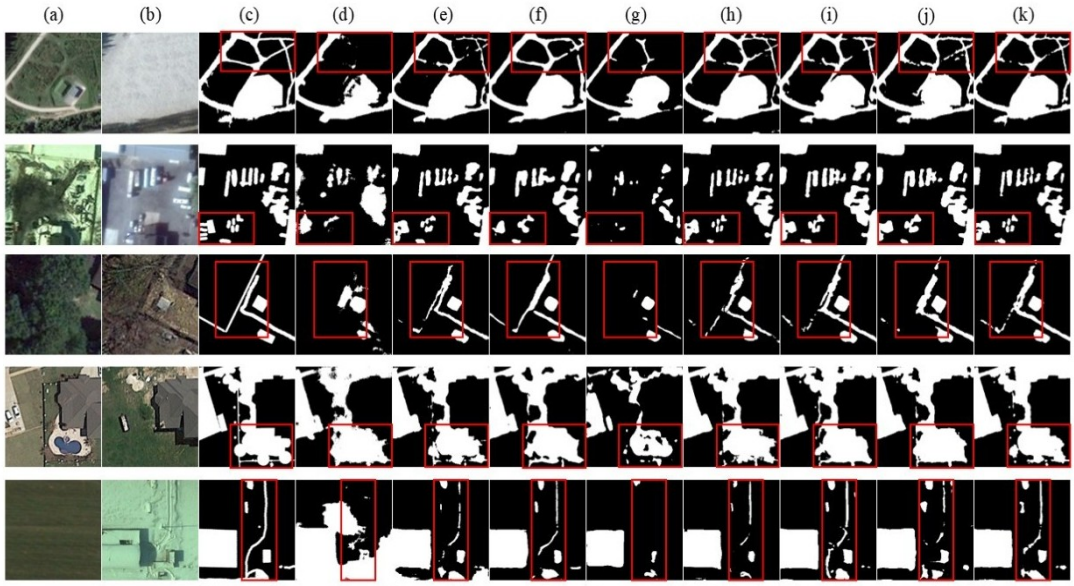
The final edge distillation loss is the weighted sum $L_{\text{edge\_distill}} = \xi_1 L_{\text{MSE}} + \xi_2 L_{\text{Gradient}}$, where we set $\xi_1 = 0.3$ and $\xi_2 = 0.7$ in our experiments.

(2)**Change detection loss**:The CD loss includes Focal loss and Dice loss.The CD loss can be expressed as: $L_{\text{change}} = \psi_1 L_{\text{Focal}} + \psi_2 L_{\text{Dice}}$, where we set $\psi_1 = 0.5$ and $\psi_2 = 0.5$.

3. Experiments

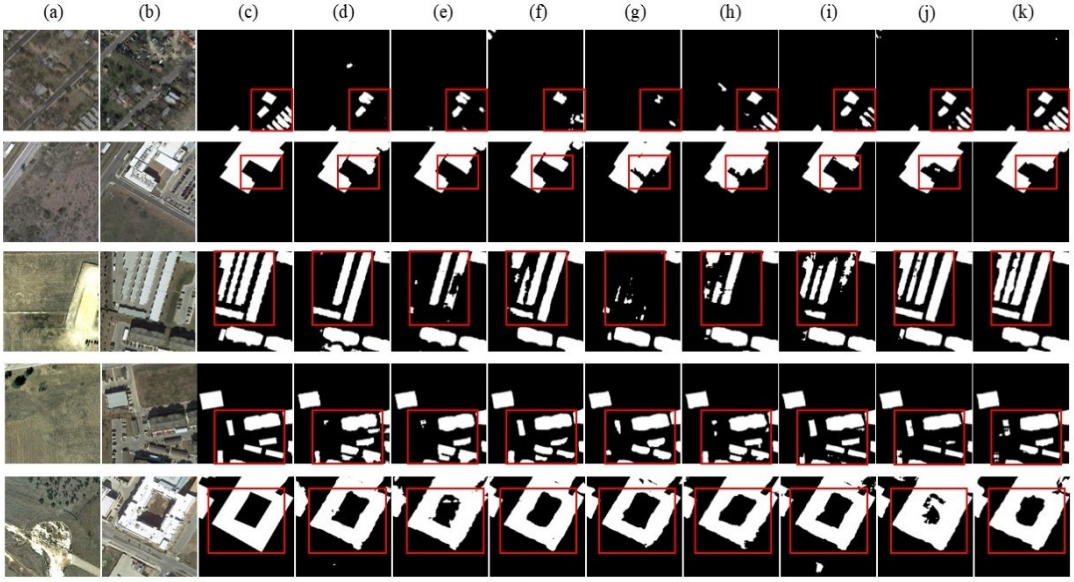
3.1 Datasets

The experiments were conducted on two publicly available CD datasets: CDD and LEVIR-CD. The CDD dataset contains 16,000 sets of high-resolution Google Earth images (256×256 pixels, 3–100 cm/pixel resolution) for real seasonal change detection, split into 10,000/3,000/3,000 sets for train/val/test. The LEVIR-CD dataset focuses on building CD, comprising



(a)pre-temporal images (b)post-temporal images (c)Ground Truth (d)FC_Siam_diff (e)SNUNet-32channel (f) AMT_Net
(g) STADE-CDNet (h) FTAN (i) HCGMNet (j) CGNet **(k) LEDGNet(ours)**

Figure 6. Visualization of Comparative Experiment Results on CDD Dataset.



(a)pre-temporal images (b)post-temporal images (c)Ground Truth (d)FC_Siam_diff (e)SNUNet-32channel (f) AMT_Net
(g) STADE-CDNet (h) FTAN (i) HCGMNet (j) CGNet **(k) LEDGNet(ours)**

Figure 7. Visualization of Comparative Experiment Results on LEVIR-CD Dataset.

637 high-resolution images (1024×1024 pixels, 0.5m resolution) collected from various cities in Texas, USA. We used its cropped version(256×256 pixels)with 31,333 change instances, split into 7,120/1,024/2,048 sets for train/val/test.

3.2 Parameter Setting

All experiments were implemented on the Ubuntu 20.04 platform based on the Pytorch 2.0.0 framework and the Python 3.10 programming language, using an RTX 3090 GPU (24GB VRAM)). In the experiments, batch size was set to 4, `num_workers` is set to 0, and the initial learning rate is set to 1×10^{-4}. The learning rate was adjusted by observing whether the F1 score of the validation set increased within 8 epochs, and the learning rate was reduced to half of the original rate if no increase was observed. We used the AdamW optimizer with the training epochs set to 100 epochs for both the CDD and LEVIR-CD datasets for the teacher model and 60 epochs for the student model. The tests were conducted with the best-performing weight files on the validation set.

3.3 Evaluation Metrics

(1)**Accuracy evaluation metrics**: The accuracy evaluation includes precision, recall, F1-score and IoU to evaluate the performance of our model. Additionally, referencing literature (Ma et al., 2024a) (Wang et al., 2023), we designed an object-level approach to evaluate small target detection accuracy, with metrics same as above. We filtered small targets via a 32×32 threshold and determined their detection accuracy by the IoU (threshold 0.5) of true and predicted target pixel areas, following generic remote sensing target detection literature (Wang et al., 2023) (Yu et al., 2023) (Yu and Ji, 2022).

(2)**Efficiency evaluation metrics**: Although the teacher net-

Ablation module				Overall accuracy evaluation				Only for small objects			
Baseline	EDM	StarLite	LiteDecoder	P(%)	R(%)	F1(%)	IoU(%)	P(%)	R(%)	F1(%)	IoU(%)
✓				95.62	93.31	94.45	89.48	80.29	79.02	79.64	66.18
✓	✓			95.52	94.13	94.82	90.15	84.69	76.71	80.50	67.37
✓	✓	✓		95.75	94.90	95.32	91.06	83.13	81.12	82.11	69.65
✓		✓	✓	**96.12**	94.64	95.37	91.16	84.79	81.00	82.85	70.72
✓	✓	✓	✓	96.08	**95.57**	**95.82**	**91.98**	**85.68**	**82.06**	**83.83**	**72.16**

Table 3. Ablation Studies on the CDD Dataset

Ablation module				Overall accuracy evaluation				Only for small objects			
Baseline	EDM	StarLite	LiteDecoder	P(%)	R(%)	F1(%)	IoU(%)	P(%)	R(%)	F1(%)	IoU(%)
✓				91.08	89.41	90.23	82.78	89.37	87.70	88.53	79.41
✓	✓			91.27	91.15	91.21	83.84	**91.51**	85.70	88.51	79.38
✓	✓	✓		91.67	91.14	91.40	84.16	90.33	89.44	89.88	81.62
✓		✓	✓	**92.20**	90.51	91.35	84.08	87.04	**91.50**	89.21	80.53
✓	✓	✓	✓	91.51	**91.33**	**91.41**	**84.19**	90.69	90.04	**90.36**	**82.42**

Table 4. Ablation Studies on the LEVIR-CD Dataset

work excels at fine-grained edge feature processing, its redundant architecture limits lightweight performance, and LEDGNet is a lightweight improvement based on it. Thus, we use three efficiency metrics—FLOPs, Params, and Inference Time—to measure the model's parameters and computational cost.

3.4 Comparison Methods

To rigorously assess LEDGNet, we benchmark it against 9 widely cited CD networks that span conventional designs through the latest CNN-Transformer hybrids, including FC_Siam_diff (Daudt et al., 2018), SNUNet(32 channel) (Fang et al., 2022), ChangeFormer (Bandara and Patel, 2022), BIT-CD (Chen et al., 2022), AMT_Net (Liu et al., 2023b), STADE-CDNet (Li et al., 2024), FTAN (Yu et al., 2024), HCGMNet (Han et al., 2023a) and CGNet (Han et al., 2023b).

3.5 Experimental Results

3.5.1 Comparison Experiments: The comparative results of the extensive experimental work performed on the two datasets are shown in Tables 1 and 2. On the CDD dataset, our method outperforms other methods in the overall assessment of all indicators and in small target detection. On the LEVIR-CD dataset, our method ranks second overall, with an FI score that is 0.58% lower than the top-ranked CGNet network. Compared to CGNet, which performed best, our method uses only 61% of the parameters and 43% of the computational cost.

To visually compare LEDGNet with other methods in terms of CD performance, we visualized the results of these methods on the CDD (Figure 6) and LEVIR-CD datasets (Figure 7). They show the visualization results of the seven methods on the test set. As shown, LEDGNet identifies the boundaries and major structures of building changes more accurately and clearly than other methods. It also significantly reduces errors and omissions and effectively preserves edge and detail information. Additionally, LEDGNet performs well in identifying narrow roads and irregular, less obvious change areas. The visualization comparison confirms our method's superior performance in handling complex CD scenarios.

3.5.2 Ablation Study: This section aims to evaluate the effectiveness of the method proposed in this paper through ablation study. The key improvements and innovations of LEDGNet are focused on three core components: the EDM, the StarLite optimized by the StarBlock structure, and the LiteDecoder, which are analyzed in detail in this study according to the above order.Table 3 and 4 show the results of the ablation study on the CDD dataset and LEVIR-CD dataset.

As Table 3 demonstrates, the CD ability of the student model can be slightly improved or remain stable when these three modules are progressively incorporated.Specifically, the overall CD F1 were improved by 0.37%, 0.50%, and 0.50%, respectively, and for the small targets in the image, the F1 appeared to be improved more significantly, by 0.86%, 1.61%, and 1.72%, respectively. The above metrics show that for edge-complex CDD datasets, the edge learning and CD tasks can be implemented in a unified network, while the lightweight improvements we made did not affect the CD performance of the model. Similarly, ablation results on LEVIR-CD in Table 4 show that progressively incorporating the three modules yields consistent gains in both F1 score and IoU. Specifically, the overall CD F1 are improved by 0.98%, 0.19%, and 0.01%, respectively; for the small change buildings in the image, the F1 are also improved more obviously, with -0.02%, 1.37%, and 0.48%, respectively. To verify the necessity of the EDM, a non-distilled experiment was established. As shown in the results, integrating the EDM consistently improves performance across both benchmarks. Specifically, on LEVIR-CD, the EDM yields a 0.06% gain in overall F1 and a significant 1.15% boost for small targets. On CDD, it enhances the overall F1 by 0.45% and the small-target F1 by 0.98%. These gains, especially the marked improvement in small-target recognition, demonstrate that the EDM effectively transfers critical boundary knowledge to the student network. Therefore, the above experimental results also prove the effectiveness of our model in edge perception enhancement and the stability of the model detection performance when lightweight improvement is carried out.

3.5.3 Efficiency Evaluation: Table 5 contrasts the model size, computational cost, and inference latency of LEDGNet with the seven reference CD networks, using $3 \times 256 \times 256$ inputs for a consistent benchmark. LEDGNet requires 20.58 M parameters, placing it among the lightest architectures in the cohort, roughly half the size of FTAN and under two-thirds that of CGNet or HCGMNet. Its 35.18 G FLOPs confirm a modest arithmetic workload, markedly leaner than the frequency-heavy FTAN(211.06 G) and the hierarchically fused HCGMNet (318.42 G). While certain simpler architectures may offer lower latency, LEDGNet strikes an optimal balance between resource consumption and detection efficiency, ensuring high-fidelity

boundary reconstruction without the prohibitive costs of heavy Transformer-based models.

A single forward pass of LEDGNet executes in 255 ms, faster than transformer-based AMT_Net (611 ms) and difference-enhanced STADE-CDNet (321 ms), and only marginally slower than the minimalist FC_Siam_diff baseline (103 ms) that trades speed for lower accuracy. These metrics demonstrate that LEDGNet strikes a favorable balance: it approaches the agility of the most compact Siamese CNNs yet delivers performance competitive with substantially heavier, attention-rich models.

Method	Params(M)	FLOPs(G)	Inference Time(ms)
FC_Siam_diff	12.12	41.81	102.76
SNUNet-32channel	12.03	54.83	272.46
AMT_Net	24.67	21.56	611.48
STADE-CDNet	31.25	13.22	321.41
FTAN	42.27	211.06	3423.60
HCGMNet	47.32	318.42	232.24
CGNet	33.68	82.23	154.50
LEDGNet	20.58	35.18	255.06

Table 5. Statistics on Model Params, FLOPs and Inference Time

4. Conclusion

In this work, we introduced LEDGNet, an edge-knowledge-distillation-guided network that demonstrably reconciles computational efficiency with high-fidelity change detection. Through the synergistic use of a teacher–student training scheme, the Edge Distillation Module, the StarLite backbone, and the LiteDecoder, LEDGNet trims more than sixty percent of the FLOPs and nearly half of the inference latency of its full-capacity teacher while preserving F1 scores to within half a percentage point. Qualitative inspections confirm that the model outlines intricate building edges, narrow transportation corridors, and other fine-scale structures that lightweight baselines routinely miss, validating the value of edge-guided supervision for small-object recognition and boundary precision. Equally important, the network’s modular design makes each component individually replaceable, enabling straightforward adaptation to evolving hardware constraints or emerging low-rank, quantized, and mixed-precision operators.

Acknowledgment

This Research was Supported by Open Project Funds for the Joint Laboratory of Spatial Intelligent Perception and Large Model Application [Grant No. SIPLMA-2025-ZD-02]; the National Natural Science Foundation of China [grant number 42401500]; the Fundamental Research Funds for Chinese Academy of Surveying and Mapping [grant number AR2503].

References

Bandara, W. G. C., Patel, V. M., 2022. A transformer-based siamese network for change detection. *IGARSS 2022 - 2022 IEEE International Geoscience and Remote Sensing Symposium*, 207–210.

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S., 2020. End-to-end object detection with transformers. *COMPUTER VISION - ECCV 2020, PT I*, 12346, 213–229.

Chen, H., Qi, Z., Shi, Z., 2022. Remote Sensing Image Change Detection With Transformers. *IEEE Transactions on Geoscience and Remote Sensing*, 60, 1-14.

Chen, Y., Ning, X., Zhang, R., Zhang, H., Huang, X., He, Y., 2025. ESMII-Net: An edge-synergy and multidimensional information interaction network for remote sensing change detection. *International Journal of Applied Earth Observation and Geoinformation*, 139, 104507.

Daudt, R. C., Saux, B. L., Boulch, A., 2018. Fully Convolutional Siamese Networks for Change Detection. *CoRR*, abs/1810.08462.

Fang, S., Li, K., Shao, J., Li, Z., 2022. SNUNet-CD: A Densely Connected Siamese Network for Change Detection of VHR Images. *IEEE Geoscience and Remote Sensing Letters*, 19, 1-5.

Gou, J., Yu, B., Maybank, S. J., Tao, D., 2021. Knowledge Distillation: A Survey. *INTERNATIONAL JOURNAL OF COMPUTER VISION*, 129(6), 1789-1819.

Han, C., Wu, C., Du, B., 2023a. Hcgmnet: A hierarchical change guiding map network for change detection. *IGARSS 2023 - 2023 IEEE INTERNATIONAL GEOSCIENCE AND REMOTE SENSING SYMPOSIUM*, 5511–5514.

Han, C., Wu, C., Guo, H., Hu, M., Li, J., Chen, H., 2023b. Change Guiding Network: Incorporating Change Prior to Guide Change Detection in Remote Sensing Imagery. *IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATIONS AND REMOTE SENSING*, 16, 8395-8407.

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H., 2017. Mobilenets: Efficient convolutional neural networks for mobile vision applications.

Lei, T., Xu, Y., Ning, H., Lv, Z., Min, C., Jin, Y., Nandi, A. K., 2024. Lightweight Structure-Aware Transformer Network for Remote Sensing Image Change Detection. *IEEE Geoscience and Remote Sensing Letters*, 21, 1-5.

Li, Z., Cao, S., Deng, J., Wu, F., Wang, R., Luo, J., Peng, Z., 2024. STADE-CDNet: Spatial–Temporal Attention With Difference Enhancement-Based Network for Remote Sensing Image Change Detection. *IEEE Transactions on Geoscience and Remote Sensing*, 62, 1-17.

Li, Z., Tang, C., Liu, X., Zhang, W., Dou, J., Wang, L., Zomaya, A. Y., 2023. Lightweight Remote Sensing Change Detection With Progressive Feature Aggregation and Supervised Attention. *IEEE Transactions on Geoscience and Remote Sensing*, 61, 1-12.

Liu, D., Xie, B., Zhang, J., Ding, R., 2023a. An Extremely Lightweight Change Detection Algorithm Based on Light Global-Local Feature Enhancement Module. *IEEE Geoscience and Remote Sensing Letters*, 20, 1-5.

Liu, W., Li, J., Wang, H., Tan, R., Fu, Y., Tian, Q., 2025. LCD-Net: A Lightweight Remote Sensing Change Detection Network Combining Feature Fusion and Gating Mechanism. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 18(1), 7769-7780.

Liu, W., Lin, Y., Liu, W., Yu, Y., Li, J., 2023b. An attention-based multiscale transformer network for remote sensing image change detection. *ISPRS Journal of Photogrammetry and Remote Sensing*, 202, 599-609.

Ma, W., Wang, X., Zhu, H., Yang, X., Yi, X., Jiao, L., 2024a. Significant Feature Elimination and Sample Assessment for Remote Sensing Small Objects' Detection. *IEEE Transactions on Geoscience and Remote Sensing*, 62, 1-15.

Ma, X., Dai, X., Bai, Y., Wang, Y., Fu, Y., 2024b. Rewrite the stars. *2024 IEEE/CVF CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION, CVPR 2024*, 5694–5703.

Wang, G., Zhang, N., Wang, J., Liu, W., Xie, Y., Chen, H., 2024. Knowledge Distillation-Based Lightweight Change Detection in High-Resolution Remote Sensing Imagery for On-Board Processing. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 17, 3860-3877.

Wang, X., Wang, A., Yi, J., Song, Y., Chehri, A., 2023. Small Object Detection Based on Deep Learning for Remote Sensing: A Comprehensive Review. *Remote Sensing*, 15(13).

Yang, Y., Sun, X., Diao, W., Li, H., Wu, Y., Li, X., Fu, K., 2022. Adaptive Knowledge Distillation for Lightweight Remote Sensing Object Detectors Optimizing. *IEEE Transactions on Geoscience and Remote Sensing*, 60, 1-15.

You, Z.-H., Chen, S.-B., Wang, J.-X., Luo, B., 2024. Robust feature aggregation network for lightweight and effective remote sensing image change detection. *ISPRS Journal of Photogrammetry and Remote Sensing*, 215, 31-43.

Yu, C., Li, H., Hu, Y., Zhang, Q., Song, M., Wang, Y., 2024. Frequency-Temporal Attention Network for Remote Sensing Imagery Change Detection. *IEEE Geoscience and Remote Sensing Letters*, 21, 1-5.

Yu, D., Ji, S., 2022. A New Spatial-Oriented Object Detection Framework for Remote Sensing Images. *IEEE Transactions on Geoscience and Remote Sensing*, 60, 1-16.

Yu, N., Ren, H., Deng, T., Fan, X., 2023. Stepwise Locating Bi-directional Pyramid Network for Object Detection in Remote Sensing Imagery. *IEEE Geoscience and Remote Sensing Letters*, 20, 1-5.

Zhang, X., Zhou, X., Lin, M., Sun, R., 2018. Shufflenet: An extremely efficient convolutional neural network for mobile devices. *2018 IEEE/CVF CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION (CVPR)*, 6848–6856.

Zhu, Q., Guo, X., Deng, W., Shi, S., Guan, Q., Zhong, Y., Zhang, L., Li, D., 2022. Land-Use/Land-Cover change detection based on a Siamese global learning framework for high spatial resolution remote sensing imagery. *ISPRS JOURNAL OF PHOTOGRAMMETRY AND REMOTE SENSING*, 184, 63-78.