SADE-Net: Strip-Aware Difference Enhancement and Multi-Dilation Fusion for Remote Sensing Change Detection

Zhenyu Wang, Guoxia Wang, and Gang Shi
College of Computer Science and Technology, Xinjiang University, Urumqi, China
Email: wangzhenyu1127@stu.xju.edu.cn, wangguoxia@stu.xju.edu.cn, shigang@xju.edu.cn

***Abstract*—Remote sensing change detection (CD) is a fundamental task in Earth observation, serving critical roles in urban planning, disaster assessment, and environmental monitoring. Despite the strong progress of deep learning-based CD methods, accurately detecting thin and elongated changes, such as roads, bridges, and pipelines, remains challenging. Existing methods often suffer from two limitations: 1) *feature fragmentation*, where isotropic pooling or attention operators fail to preserve the continuity of linear structures, and 2) *insufficient context aggregation*, where limited multi-scale interaction weakens the discrimination between real changes and pseudo-changes caused by illumination, seasonal variation, or sensor noise. To address these issues, we propose a Siamese encoder-decoder framework named SADE-Net. Specifically, we introduce a Strip-Aware Difference Enhancement (SADE) module that integrates horizontal and vertical strip pooling to explicitly model long-range anisotropic dependencies and preserve the structural connectivity of elongated changes. We further design a Multi-Dilation Spatial Fusion (MDSF) module at the bottleneck to aggregate multi-scale contextual information with parallel dilated convolutions in a lightweight manner. Compared with recurrent or global-attention alternatives, the proposed design retains spatial structure while substantially reducing computational cost. Experiments on LEVIR-CD and SYSU-CD demonstrate that SADE-Net achieves strong overall performance with only 1.63M parameters. On LEVIR-CD, SADE-Net reaches an F1-score of 90.85% and an IoU of 83.24%; on SYSU-CD, it achieves an F1-score of 81.85% and an IoU of 69.28%, while showing clear advantages in preserving the integrity of linear and boundary-sensitive changes.**

***Index Terms*—Change Detection, Strip Pooling, Linear Feature Extraction, Feature Fusion, Remote Sensing.**

I. Introduction

REMOTE sensing image change detection (CD) aims to identify and localize surface changes from a pair of co-registered images acquired at different times over the same geographical area [1]. It is a key technology for urban expansion monitoring, land-use analysis, infrastructure updating, and rapid disaster response [2], [26], [27]. With the increasing availability of high-resolution satellite and aerial imagery, deep learning, especially convolutional neural networks (CNNs), has become the dominant paradigm for CD due to its strong representation capability and end-to-end optimization [13], [14].

Despite these advances, accurately preserving the geometric integrity of changed objects in complex urban scenes remains difficult. **First**, a critical yet often under-emphasized issue is the detection of linear structures such as roads, elevated bridges, and narrow corridors. These targets usually exhibit extreme aspect ratios and occupy only a small fraction of the local receptive field. Conventional CNNs and many transformer-based CD methods rely on square kernels, square pooling, or isotropic window attention [6], [17]. Such operators introduce excessive surrounding background into the representation of thin targets, causing feature contamination and fragmented predictions [9], [16]. In practical mapping scenarios, fragmented outputs often break road connectivity and reduce the usability of the final map.

Second, pseudo-change suppression is still challenging. Seasonal appearance shifts, illumination differences, and sensor noise can produce strong spectral discrepancies even when no semantic change exists. Early Siamese CD models based on direct subtraction or concatenation [1], [3] often lack sufficient contextual interaction to separate genuine structural changes from background variation. Although attention-based methods alleviate this issue to some extent [4], [5], they may either incur high computational overhead or still treat spatial context in an isotropic manner.

To address these limitations, we propose **SADE-Net**, a lightweight framework tailored to the characteristics of linear and boundary-sensitive changes. The key idea is to inject an explicit geometric prior into bi-temporal fusion and combine it with efficient multi-scale context modeling. Our main contributions are summarized as follows:

1) We propose the **Strip-Aware Difference Enhancement (SADE)** module, which introduces horizontal and vertical strip pooling into bi-temporal feature fusion. Unlike standard isotropic context modeling, SADE explicitly aggregates long-range information along each axis, which is well aligned with the geometry of roads and other elongated structures.
2) We design the **Multi-Dilation Spatial Fusion (MDSF)** module as a lightweight bottleneck for multi-scale context aggregation. By using parallel dilated convolutions instead of recurrent units or global self-attention, MDSF enlarges the receptive field while preserving 2D spatial structure and maintaining low complexity.
3) We conduct comprehensive experiments on the LEVIR-CD and SYSU-CD benchmarks. Quantitative and qualitative results show that SADE-Net achieves competitive or superior performance while using only 1.63M

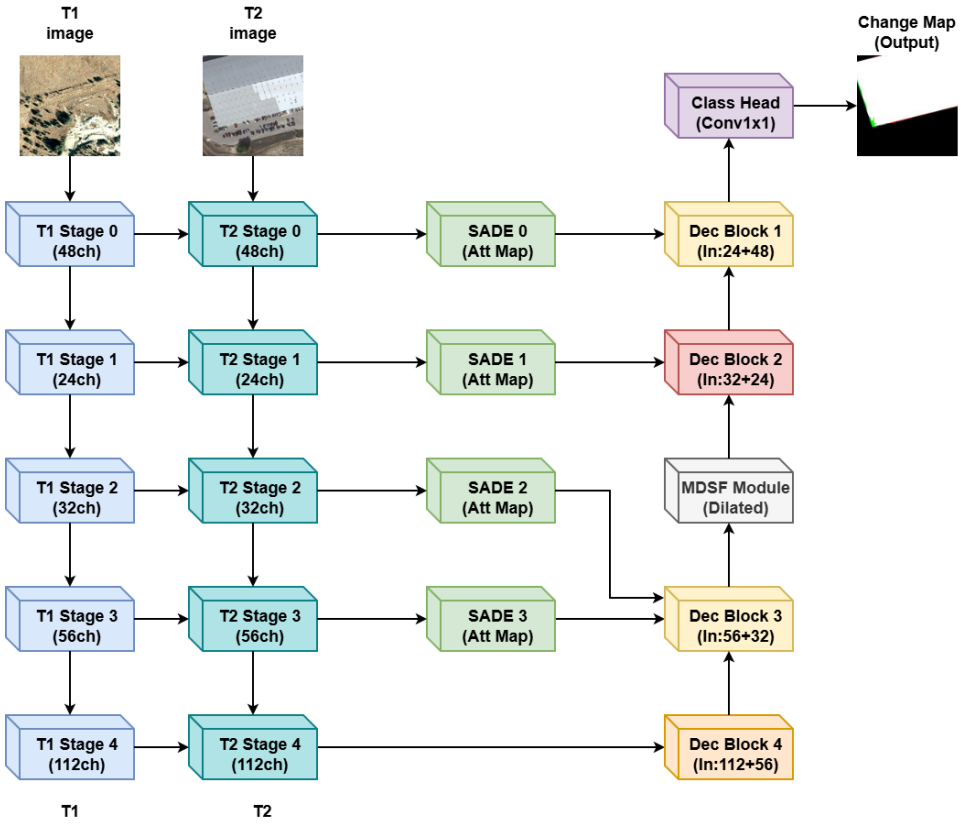

Fig. 1. Overall architecture of SADE-Net. A Siamese EfficientNet-B4 encoder extracts bi-temporal features. SADE modules are inserted into Stages 0–3 to enhance skip-level fusion, while the MDSF module is embedded at the bottleneck to aggregate multi-scale context before decoding.

parameters, and yields clearer structural continuity on challenging linear targets.

To improve reproducibility, all experiments are conducted on public splits with fixed training protocols, and the reported quantitative results are averaged over three independent runs. The code and implementation details will be released publicly.

II. Related Work

A. Deep Learning for Change Detection

The development of remote sensing CD has closely followed the evolution of deep neural architectures [15]. Early methods were dominated by fully convolutional Siamese networks. Daudt *et al.* [1] introduced FC-Siam-conc and FC-Siam-diff, demonstrating the effectiveness of weight-shared encoders for bi-temporal analysis. Subsequent methods such as improved UNet++ [3] and SNUNet-CD [4] adopted dense connections to enhance gradient flow and feature reuse.

With the rise of attention mechanisms, change detection models began to explicitly model salient temporal interactions. STA-Net [5] incorporated spatial-temporal attention, while BIT [6], ChangeFormer [17], and Changer [18] introduced transformer-style reasoning for long-range dependencies. More recent methods, such as RCTNet [7] and ChangeMamba [8], further improved performance through context transformers and state-space modeling.

However, most of these methods still rely on isotropic spatial operators. Standard convolutions use square kernels, and transformer variants usually partition features into square windows or patches. This design is suboptimal for thin and elongated targets whose discriminative information mainly lies along one dominant spatial direction. As a result, linear changes are prone to fragmentation or discontinuity, especially in cluttered urban scenes.

B. Topology-Aware and Boundary-Sensitive Modeling

Several studies in remote sensing and segmentation have emphasized that preserving topology and boundaries is crucial for structured targets. For example, D-LinkNet [16] and strip pooling [10] demonstrated that elongated structures benefit from directional context aggregation rather than purely isotropic operations. In change detection, recent works have also explored lightweight or structure-sensitive designs [9], [29]. In addition, boundary-aware or detail-preserving strategies can improve the delineation of fine-scale changed regions, especially around man-made structures.

Our method is related to this line of work, but differs in two important aspects. First, instead of applying directional context only to single-image segmentation features, SADE-Net embeds strip-aware aggregation directly into *bi-temporal fusion*, allowing the network to compare two time points while preserving directional continuity. Second, our MDSF

module complements this topology-aware design with efficient multi-scale context aggregation, which helps suppress pseudo-changes without relying on computationally heavy recurrent or transformer blocks.

C. Feature Fusion and Context Aggregation

Feature fusion is the central operation in Siamese CD. Early methods mainly used subtraction or concatenation [12]. To improve discrimination, later approaches introduced non-local or attention-based fusion [28]. Some Siamese CD architectures also employed LSTM-style modules to model temporal relationships, but these require reshaping 2D feature maps into sequential representations and add notable complexity.

Global self-attention offers broad context but usually incurs quadratic complexity with respect to the number of tokens, which is unfriendly for high-resolution remote sensing images [19]. In contrast, dilated convolutions provide an efficient way to enlarge the receptive field while keeping the underlying 2D topology intact [22], [23]. Motivated by this trade-off, our MDSF module aggregates local, medium-range, and larger-scale spatial context using parallel dilated branches, achieving a favorable balance between effectiveness and efficiency.

III. Proposed Method

A. Network Overview

The overall architecture of SADE-Net is shown in Fig. 1. The framework follows a Siamese encoder-decoder structure with skip connections. Given a pair of co-registered images $T_1, T_2 \in \mathbb{R}^{H\times W\times 3}$, a weight-shared **EfficientNet-B4** backbone [11] extracts five stages of features, denoted by $F_i^{(1)}$ and $F_i^{(2)}$ for $i \in \{0,1,2,3,4\}$. The corresponding channel dimensions are $C \in [48, 24, 32, 56, 112]$.

To enhance bi-temporal interaction, SADE modules are inserted into the skip paths of Stages 0–3. These modules transform the paired encoder features into enhanced fusion representations D_i, which are then forwarded to the decoder. At the deepest stage, the MDSF module is employed before decoding to enrich global and multi-scale contextual information. This design separates the two key objectives of the network: directional preservation of linear structures in the skip connections, and efficient contextual aggregation at the bottleneck.

B. Strip-Aware Difference Enhancement (SADE)

The SADE module is designed to preserve the continuity of thin, elongated changes. Unlike conventional isotropic pooling, SADE aggregates context along horizontal and vertical directions independently, which better matches the geometry of roads and similar structures. Let the input features be $F^{(1)}, F^{(2)} \in \mathbb{R}^{C\times H\times W}$.

1) Bi-Temporal Fusion: We first concatenate the two temporal features along the channel dimension and then compress the channel number back to C using a 1×1 convolution:

$$X = \mathrm{Conv}_{1\times 1}(\mathrm{Concat}(F^{(1)}, F^{(2)})). \tag{1}$$

This step preserves richer complementary information than direct subtraction, while the 1×1 projection removes redundancy and keeps the module lightweight.

2) Parallel Context Aggregation: The fused feature X is processed by two complementary branches.

1) Strip-context branch. This branch captures long-range directional dependencies. Horizontal and vertical average pooling are defined as

$$y^h_{c,i} = \frac{1}{W}\sum_{0\le j<W} x_{c,i,j}, \tag{2}$$

$$y^v_{c,j} = \frac{1}{H}\sum_{0\le i<H} x_{c,i,j}. \tag{3}$$

The resulting 1D descriptors are transformed by 1×1 convolutions, upsampled to the original spatial size, concatenated, and fused as

$$M_{\mathrm{global}} = \mathrm{Conv}_{1\times 1}(\mathrm{Concat}(\mathrm{Upsample}(y^h), \mathrm{Upsample}(y^v))). \tag{4}$$

Intuitively, this branch tells each pixel how it relates to other pixels in the same row and column, which is particularly useful when a changed road segment is long but only a few pixels wide.

2) Local-texture branch. To retain local detail and boundary sensitivity, we apply a standard 3×3 convolution:

$$M_{\mathrm{local}} = \mathrm{Conv}_{3\times 3}(X). \tag{5}$$

This branch complements strip pooling by focusing on nearby texture and edge information.

3) Attention-Guided Enhancement: The two branches are fused to form a unified context representation:

$$M_{\mathrm{fuse}} = \mathrm{Conv}_{1\times 1}(\mathrm{Concat}(M_{\mathrm{global}}, M_{\mathrm{local}})), \tag{6}$$

followed by a sigmoid activation to obtain the attention map

$$\mathcal{A} = \sigma(M_{\mathrm{fuse}}). \tag{7}$$

The final output is defined as

$$\hat{X} = X \otimes \mathcal{A} + M_{\mathrm{fuse}} + X, \tag{8}$$

where $\otimes$ denotes element-wise multiplication. This residual formulation preserves the original fused representation, highlights spatially important responses, and explicitly injects the strip-aware context into the skip feature.

C. Multi-Dilation Spatial Fusion (MDSF)

The MDSF module is placed at the bottleneck to aggregate context at multiple scales without destroying the 2D structure of the feature map. Let $F_{\mathrm{in}} \in \mathbb{R}^{C\times H\times W}$ denote the bottleneck input. The module splits it into four parallel branches:

1) **Edge branch (B_0):** a 1×1 convolution that preserves compact structural information and controls complexity.
2) **Context branch (B_1):** a 3×3 convolution with dilation rate $d = 1$ for local context.
3) **Context branch (B_2):** a 3×3 convolution with dilation rate $d = 3$ for medium-range context.

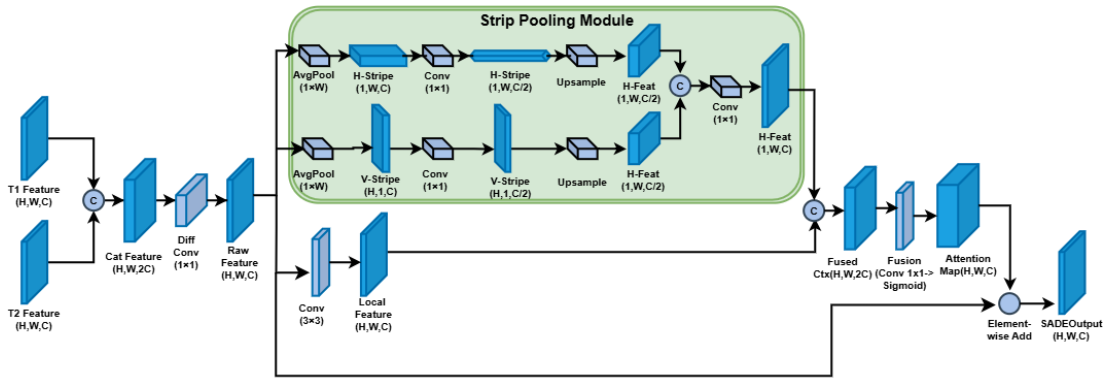

Fig. 2. Structure of the Strip-Aware Difference Enhancement (SADE) module. The module first concatenates bi-temporal features and reduces the channel dimension. It then uses a strip-context branch and a local-texture branch to construct an attention map for feature enhancement.

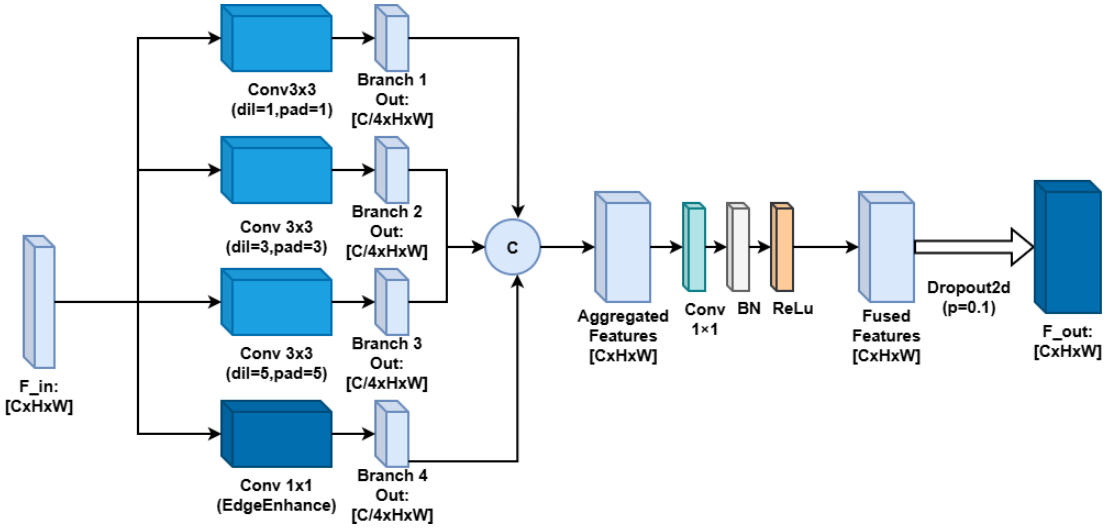

Fig. 3. Structure of the Multi-Dilation Spatial Fusion (MDSF) module. The module combines one lightweight 1×1 branch and three parallel dilated-convolution branches with rates 1, 3, and 5 to aggregate multi-scale spatial context.

4) **Context branch (B_3):** a 3×3 convolution with dilation rate $d = 5$ for larger-scale context.

All four branches output features with channel dimension $C/2$. Their concatenation is fused through a 1×1 convolution:

$$F_{\text{mdsf}} = \text{Conv}_{1\times1}(\text{Concat}(B_0, B_1, B_2, B_3)). \tag{9}$$

A spatial dropout layer with rate 0.1 is then applied for regularization.

The role of MDSF is intuitive: the small-dilation branch retains fine details, while the larger-dilation branches see broader neighborhoods and help suppress pseudo-changes that cannot be resolved from local evidence alone. Compared with LSTM-based or global-attention alternatives, MDSF keeps the computation entirely in the image plane and is therefore easier to optimize and more efficient at inference time.

D. Loss Function

Given the severe foreground-background imbalance in change detection, we employ a hybrid loss composed of binary cross-entropy (BCE) loss and Dice loss [24]:

$$\mathcal{L}_{\text{bce}} = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)\right], \tag{10}$$

$$\mathcal{L}_{\text{dice}} = 1 - \frac{2\sum_{i=1}^{N} y_i \hat{y}_i + \epsilon}{\sum_{i=1}^{N} y_i + \sum_{i=1}^{N} \hat{y}_i + \epsilon}, \tag{11}$$

where y_i and $\hat{y}_i$ denote the ground truth and prediction of the i-th pixel, N is the total number of pixels, and ϵ is a small constant for numerical stability. The overall loss is

$$\mathcal{L}_{\text{total}} = \lambda \mathcal{L}_{\text{bce}} + (1-\lambda)\mathcal{L}_{\text{dice}}. \tag{12}$$

In all experiments, we set $\lambda = 0.5$, which provides a stable compromise between per-pixel classification and overlap optimization [25].

IV. Experiments

A. Datasets

We evaluate SADE-Net on two widely used public datasets:

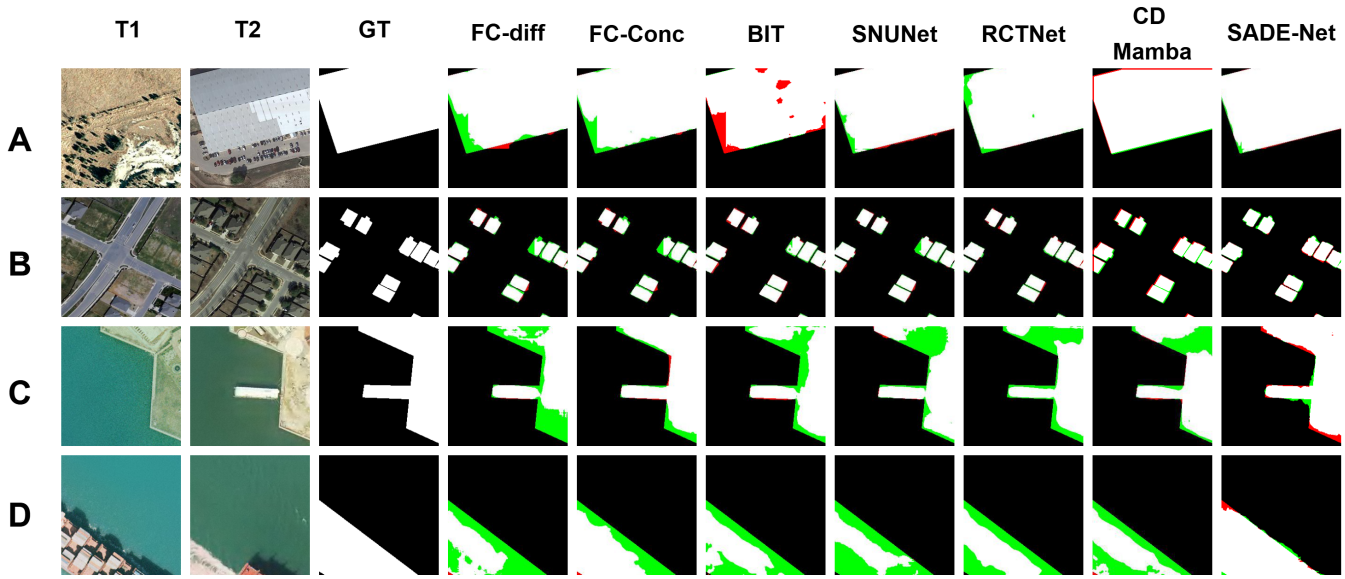

Fig. 4. Qualitative comparison of different change detection methods. Samples (a) and (b) are from LEVIR-CD and emphasize building changes, while samples (c) and (d) are from SYSU-CD and include more complex urban scenes with thin or elongated structures. White denotes true positives, red denotes false negatives, and green denotes false positives. SADE-Net produces more continuous predictions on structured changes.

- **LEVIR-CD** [5]: 637 pairs of high-resolution (1024×1024) Google Earth images with 0.5 m/pixel resolution, focusing primarily on building changes. We follow the standard split of 445 training pairs, 64 validation pairs, and 128 test pairs.
- **SYSU-CD** [12]: 20,000 pairs of aerial images of size 256×256 at 0.5 m/pixel. Compared with LEVIR-CD, SYSU-CD is more diverse and includes roads, vegetation, buildings, water, and other urban changes. We use the official split of 12,000/4,000/4,000 for train/validation/test.

B. Evaluation Metrics

We use Precision (P), Recall (R), F1-score ($F1$), and Intersection over Union (IoU), computed from true positives (TP), false positives (FP), and false negatives (FN):

$$P = \frac{TP}{TP + FP}, \qquad R = \frac{TP}{TP + FN}, \tag{13}$$

$$F1 = \frac{2PR}{P + R}, \qquad IoU = \frac{TP}{TP + FP + FN}. \tag{14}$$

Among them, $F1$ and IoU are the main indicators of overall performance. Unless otherwise stated, all reported results are averaged over three independent runs.

C. Implementation Details

The model is implemented in PyTorch and trained on a single NVIDIA RTX 3090 GPU (24 GB). We use AdamW with an initial learning rate of 1×10^{-4} and weight decay of 1×10^{-4}. A cosine annealing schedule decreases the learning rate to 1×10^{-6}. The batch size is 16, and each model is trained for 250 epochs. Data augmentation includes random flipping, rotation, and color jittering. For reproducibility, we use fixed dataset splits and report the average over three runs with the same training protocol. The parameter count is computed by summing the number of learnable parameters in the network, yielding approximately **1.63M** parameters.

D. Comparison with State-of-the-Art Methods

We compare SADE-Net with representative baselines, including classical Siamese models (FC-Siam-Diff and FC-Siam-Conc [1]), a transformer-based method (BIT [6]), a densely connected architecture (SNUNet [4]), and more recent advanced models (RCTNet [7] and ChangeMamba [8]).

Quantitative comparison. The results are summarized in Table I. On **LEVIR-CD**, SADE-Net achieves an F1-score of **90.85%** and an IoU of **83.24%**, outperforming several recent methods while using significantly fewer parameters. On **SYSU-CD**, SADE-Net reaches an F1-score of **81.85%** and an IoU of **69.28%**, slightly surpassing RCTNet in overall accuracy and showing better structural continuity in the qualitative examples of Fig. 4. These improvements indicate that explicitly encoding directional context is beneficial for complex urban scenes that contain elongated or boundary-sensitive changes.

Precision-recall behavior. SADE-Net consistently achieves higher recall than precision on both datasets. This is consistent with the design objective of preserving the completeness of changed structures: strip-aware aggregation helps recover connected linear regions, but it can also activate ambiguous background pixels under severe appearance variations. In applications such as infrastructure updating, this trade-off is often preferable because disconnected detections are more harmful than a limited number of extra pixels that can be filtered in post-processing.

Efficiency comparison. In addition to accuracy, SADE-Net is highly efficient. It uses only **1.63M** parameters and

TABLE I
QUANTITATIVE COMPARISON ON LEVIR-CD AND SYSU-CD. BEST RESULTS ARE SHOWN IN **BOLD**; SECOND BEST RESULTS ARE UNDERLINED.

Method	Year	Params (M)	FLOPs (G)	LEVIR-CD				SYSU-CD			
				Pre (%)	Rec (%)	F1 (%)	IoU (%)	Pre (%)	Rec (%)	F1 (%)	IoU (%)
FC-Siam-Diff [1]	2018	1.54	5.30	89.53	83.31	86.31	75.92	89.13	61.21	72.58	56.96
FC-Siam-Conc [1]	2018	1.35	4.70	**91.99**	76.77	83.69	71.96	82.54	71.03	76.35	61.75
BIT [6]	2021	3.55	67.80	89.24	89.37	89.37	80.68	81.14	76.48	78.74	64.94
SNUNet [4]	2021	12.03	54.83	89.18	87.17	88.16	78.83	78.26	76.30	77.27	62.96
RCTNet [7]	2024	23.37	20.71	88.48	73.89	80.53	67.41	**84.90**	78.30	81.49	68.76
CDMamba [8]	2025	11.91	21.25	87.96	89.08	88.52	79.40	82.28	69.06	75.09	60.12
SADE-Net (Ours)	–	**1.63**	**8.31**	87.87	**94.05**	**90.85**	**83.24**	77.70	**86.46**	**81.85**	**69.28**

8.31G FLOPs, which is much smaller than RCTNet (23.37M) and ChangeMamba (11.91M), and also significantly lighter than transformer or dense-connection designs such as BIT and SNUNet. This favorable balance is mainly due to the use of strip pooling and dilated convolutions instead of heavy global reasoning modules.

E. Ablation Study and Discussion

We further analyze the contributions of SADE and MDSF on SYSU-CD. To make the comparison more interpretable, the baseline keeps the same encoder-decoder framework and replaces SADE with direct concatenation and MDSF with a single 1×1 convolution. The results are reported in Table II.

Several observations can be made. First, adding **SADE only** improves both F1 and IoU over the baseline, indicating that directional context aggregation is the dominant factor for structured change modeling. Second, adding **MDSF only** also improves performance, mainly by increasing recall through richer multi-scale context. Third, the **full model** yields the best overall score in the main benchmark comparison of Table I and achieves the most balanced precision–recall trade-off among the tested variants.

It is worth noting that the SADE-only variant attains a slightly higher F1 in Table II, which suggests that strip-aware fusion is highly effective by itself. Nevertheless, the full model is retained as the final design because MDSF provides complementary contextual support, improves the overall benchmark result reported in Table I, and produces better qualitative completeness on difficult examples, as shown in Fig. 4. In other words, SADE captures *where* elongated structures should remain connected, while MDSF helps determine *whether* the surrounding large-scale context supports the predicted change.

F. Qualitative Analysis

Figure 4 provides visual comparisons on both datasets. On LEVIR-CD, SADE-Net produces cleaner building boundaries with fewer broken edges. On SYSU-CD, its advantage is more apparent on roads and elongated urban structures, where competing methods either miss narrow segments or generate disconnected responses. The visual results further suggest that the proposed design is particularly useful in scenarios where preserving structural continuity is more important than maximizing raw precision.

TABLE II
ABLATION STUDY OF SADE AND MDSF ON THE SYSU-CD DATASET.

MDSF	SADE	F1 (%)	IoU (%)	Rec (%)	Pre (%)
×	×	81.35	68.57	88.17	75.52
×	✓	82.12	69.66	85.31	79.16
✓	×	82.01	69.50	**88.38**	76.50
✓	✓	**81.85**	69.28	86.46	**77.70**

V. CONCLUSION

We presented SADE-Net, a lightweight Siamese framework for remote sensing change detection. The proposed SADE module introduces strip-aware directional context into bi-temporal fusion, which improves the continuity of thin and elongated changed structures. The MDSF module complements this design by aggregating multi-scale context at the bottleneck with low computational overhead. Experiments on LEVIR-CD and SYSU-CD show that SADE-Net achieves strong quantitative performance and clear qualitative advantages on structured urban changes.

This study also suggests a useful design principle for remote sensing CD: incorporating simple geometric priors can be both more effective and more efficient than relying solely on heavier global reasoning modules. In future work, we plan to explore stronger boundary supervision, more systematic sensitivity analysis of module hyperparameters, and deployment on edge-oriented platforms.

ACKNOWLEDGMENT

This work was supported by the Key Research and Development Project in Xinjiang Uygur Autonomous Region (No. 2022B01006).

REFERENCES

[1] R. C. Daudt, B. Le Saux, and A. Boulch, "Fully convolutional siamese networks for change detection," in *Proc. IEEE ICIP*, 2018, pp. 4063–4067.

[2] H. Chen and Z. Shi, "A spatial-temporal attention-based method and a new dataset for remote sensing image change detection," *Remote Sens.*, vol. 12, no. 10, p. 1662, 2020.

[3] D. Peng, Y. Zhang, and H. Guan, "End-to-end change detection for high resolution satellite images using improved UNet++," *Remote Sens.*, vol. 11, no. 11, p. 1382, 2019.

[4] S. Fang, K. Li, J. Shao, and Z. Li, "SNUNet-CD: A densely connected Siamese network for change detection of VHR images," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2021.

[5] J. Chen et al., "Spatial-temporal attention-based method for change detection in high-resolution remote sensing images," *Remote Sens.*, vol. 12, no. 1, 2020.

[6] H. Chen, Z. Qi, and Z. Shi, "Remote sensing image change detection with transformers," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–14, 2021.

[7] L. Wang et al., "RCTNet: Region-aware Context Transformer Network for Remote Sensing Image Change Detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, pp. 1–16, 2024.

[8] H. Chen et al., "ChangeMamba: Remote Sensing Change Detection with Spatio-Temporal State Space Model," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, pp. 1–14, 2024.

[9] E. Zhang, Y. Li, and H. Lin, "A lightweight remote-sensing image-change detection algorithm based on strip pooling," *Remote Sens.*, vol. 16, no. 13, p. 2226, 2024.

[10] Q. Hou, L. Zhang, M.-M. Cheng, and J. Feng, "Strip pooling: Rethinking spatial pooling for scene parsing," in *Proc. IEEE/CVF CVPR*, 2020, pp. 4003–4012.

[11] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proc. ICML*, 2019, pp. 6105–6114.

[12] Q. Shi et al., "A deeply supervised attention metric-based network and an open aerial image dataset for remote sensing change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–16, 2021.

[13] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE CVPR*, 2016, pp. 770–778.

[14] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE CVPR*, 2015, pp. 3431–3440.

[15] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. MICCAI*, 2015, pp. 234–241.

[16] L. Zhou, C. Zhang, and M. Wu, "D-LinkNet: LinkNet with pretrained encoder and dilated convolution for high resolution satellite imagery road extraction," in *Proc. IEEE CVPR Workshops*, 2018, pp. 182–186.

[17] W. G. C. Bandara and V. M. Patel, "A transformer-based siamese network for change detection," in *Proc. IEEE IGARSS*, 2022, pp. 207–210.

[18] S. Fang et al., "Changer: Feature interaction is what you need for change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, pp. 1–11, 2023.

[19] Z. Liu et al., "Swin Transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF ICCV*, 2021, pp. 10012–10022.

[20] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE CVPR*, 2018, pp. 7132–7141.

[21] S. Woo, J. Park, J. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. ECCV*, 2018, pp. 3–19.

[22] 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.

[23] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *Proc. ICLR*, 2016.

[24] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-Net: Fully convolutional neural networks for volumetric medical image segmentation," in *Proc. 3DV*, 2016, pp. 565–571.

[25] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE ICCV*, 2017, pp. 2980–2988.

[26] S. Ji, S. Wei, and M. Lu, "Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 1, pp. 574–586, 2018.

[27] C. Zhang et al., "A deeply supervised image fusion network for change detection in high resolution bi-temporal remote sensing images," *ISPRS J. Photogramm. Remote Sens.*, vol. 166, pp. 183–200, 2020.

[28] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proc. IEEE CVPR*, 2018, pp. 7794–7803.

[29] A. Codegoni, G. Lombardi, and A. Ferrari, "TinyCD: A (not so) deep learning model for change detection," *Neural Process. Lett.*, vol. 55, pp. 1–18, 2022.