# Topology-Constrained Graph-Mamba with Logit Diffusion for Change Detection

Anonymous CVPR submission

Paper ID 7

## Abstract

*Remote sensing change detection requires distinguishing meaningful structural changes from appearance variations such as illumination or seasonal shifts. We propose **GMD-CD** (Graph-Mamba Diffusion for Change Detection), a framework that combines topology-constrained cross-temporal modeling with efficient refinement. Bi-temporal features are organized as a graph-structured bipartite representation, where regions from each time step are explicitly separated and jointly processed. Global interaction is performed using a bidirectional state-space model, enabling structured propagation with linear complexity. To improve robustness, we introduce orthogonal feature disentanglement that separates change-related and invariant components, reducing interference from appearance variations. Building on this representation, we design a conditional diffusion decoder in logit space that performs uncertainty-aware residual refinement, improving boundary quality while preserving global consistency. Experiments on multiple benchmarks show consistent gains in both accuracy and efficiency, yielding a strong accuracy–efficiency trade-off for high-resolution change detection.*

## 1. Introduction

Change detection (CD) in remote sensing aims to identify meaningful structural differences between images acquired at different times, and is fundamental to applications such as urban monitoring [4], disaster assessment [21], and environmental analysis. Despite significant progress, three challenges remain: **(i) pseudo-changes** caused by illumination, seasonal effects, or sensor variation, **(ii) boundary degradation** for thin and irregular regions, and **(iii) the need for global context** to distinguish true changes efficiently. Existing methods address these challenges with different trade-offs. CNN-based approaches [16, 19] are efficient but rely on local receptive fields, while transformer-based methods [2, 3, 7] capture long-range dependencies at quadratic cost. State-space models (SSMs) [20] provide global interaction with linear complexity, and recent Mamba-based

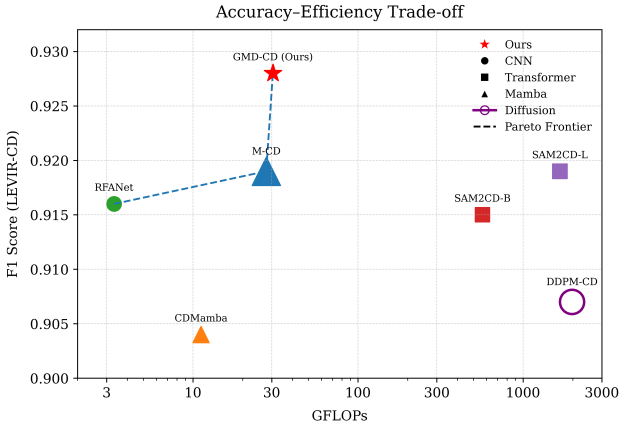


Figure 1. Accuracy–efficiency trade-off across representative CD methods.

CD methods [10, 34, 44] improve efficiency. However, they typically process features as a single flattened sequence, ignoring cross-temporal structure and leading to unstructured information mixing. Diffusion-based approaches such as DDPM-CD [1] improve prediction quality through stochastic refinement but incur high computational cost. As shown in Fig. 1, existing methods either favor efficiency (e.g., CD-Mamba [44]) or accuracy at high cost (e.g., DDPM-CD, SAM2-CD [35]). This motivates a formulation that enables structured cross-temporal reasoning with efficiency. We propose **GMD-CD**, a unified framework that models cross-temporal interaction as a *structured state-space propagation problem*. We impose a *bipartite structural constraint* over bi-temporal features, representing them as an ordered bipartite sequence and performing bidirectional state-space propagation. This induces topology-aware cross-temporal coupling and enables global interaction with linear complexity, avoiding unstructured mixing in prior Mamba-based approaches. Building on this, we introduce **Orthogonal Feature Disentanglement (OFD)** to separate change-sensitive and invariant components, improving robustness to pseudo-changes. To address boundary ambiguity, we design a **conditional diffusion decoder** operating in logit space, which

performs stochastic refinement to correct uncertain predictions while preserving efficiency.

**Contributions.**

- We introduce a **structured bipartite state-space formulation** that models cross-temporal interaction as topology-constrained propagation.
- We propose a **BGMO module** that performs partition-aware bidirectional state-space propagation with linear complexity.
- We design **Orthogonal Feature Disentanglement (OFD)** to decouple change and invariant components, improving robustness to pseudo-changes.
- We develop a **conditional diffusion decoder in logit space** for uncertainty-aware refinement with moderate overhead.
- We demonstrate consistent improvements across benchmarks while maintaining a favorable accuracy–efficiency trade-off.

## 2. Related Work

**CNN and transformer-based change detection.** Early CD methods use Siamese CNN architectures with feature differencing and encoder–decoder designs [16, 19]. Attention-based extensions improve spatial and temporal reasoning [4, 7]. Transformer models such as BIT [8] and ChangeFormer [3] capture long-range dependencies via self- and cross-attention, but incur quadratic complexity. Recent approaches scale performance with larger backbones [41], while foundation-based methods like SAM2-CD [35] achieve strong generalization at high computational cost. In contrast, we target efficient cross-temporal modeling with linear complexity.

**State-space models for vision and change detection.** Selective state-space models (SSMs), particularly Mamba [20], enable global interaction with linear complexity. Vision adaptations serialize spatial features into sequences [47], and CD variants such as ChangeMamba [10], CDMamba [44], and RSMamba [12] achieve strong efficiency. However, they treat bi-temporal features as a single unstructured sequence, limiting explicit cross-temporal modeling. We instead introduce a structured bipartite formulation with partition-aware state-space propagation.

**Graph and structured modeling.** Graph-based approaches model spatial relationships via explicit node interactions and message passing, but incur additional computational overhead. We adopt a lightweight alternative by organizing pooled regions into an ordered bipartite structure, where interactions are implicitly realized through state-space transitions without explicit graph construction.

**Diffusion-based change detection.** Diffusion models have been applied to CD as generative or refinement mechanisms. DDPM-CD [1] performs full diffusion-based generation with high cost, while other works explore generative modeling for representation learning [25]. In contrast, we use diffusion as a conditional decoder in logit space, enabling targeted refinement with a small number of DDIM steps and moderate overhead.

## 3. Method

### 3.1. Overall Architecture

Given bi-temporal images $(I^{t_1}, I^{t_2})$, we extract hierarchical features using a shared Swin-Tiny backbone [29], producing multi-scale representations $\{f_l^a, f_l^b\}_{l=1}^4$, as illustrated in Fig. 2. The model consists of three stages: *(i) multi-scale state encoding*, *(ii) structured state propagation*, and *(iii) logit refinement*. At shallow levels ($l = 1, 2$), we apply *LightFusion* to capture local differences using feature interaction and residual aggregation. These stages focus on fine-grained appearance variations. At deeper levels ($l = 3, 4$), we perform global cross-temporal modeling using the proposed *Graph-structured State-Space Operator (BGMO)*. Features are pooled into a bipartite node representation and arranged in a partitioned sequence, which preserves temporal identity. Bidirectional state-space propagation [20] then enables structured cross-temporal interaction with linear complexity. The resulting representation is refined using *Orthogonal Feature Disentanglement (OFD)*, which separates change-related and invariant components. The multi-scale features are fused using a lightweight top-down decoder to produce coarse change logits $L_0$. Finally, a *conditional diffusion decoder* operates in logit space and refines $L_0$ using a small number of denoising steps, improving boundary quality while preserving global consistency.

### 3.2. Multi-scale State Encoding

We extract hierarchical features from bi-temporal inputs $(I^{t_1}, I^{t_2})$ using a shared encoder, producing $\{f_l^a, f_l^b\}_{l=1}^4$ at multiple resolutions. These features capture spatial context at increasing receptive fields and serve as inputs for subsequent state modeling. At shallow levels ($l \in \{1, 2\}$), changes are mostly local. Following prior CD methods [16, 19], we encode both differences and agreements:

$$x_l = \mathrm{Conv}_{1\times1}\big([|f_l^a - f_l^b|;\ f_l^a \odot f_l^b]\big)\,, \tag{1}$$

where $|\cdot|$ captures change magnitude and $\odot$ captures shared responses. This provides a compact local representation sensitive to both variation and consistency. To improve spatial selectivity, we apply lightweight channel and spatial attention [38]:

$$S_l = x_l \cdot \sigma(\mathrm{Conv}_{7\times7}([\mathrm{AvgPool}(x_l);\ \mathrm{MaxPool}(x_l)]))\,, \tag{2}$$

which suppresses background noise and enhances change-relevant regions with minimal overhead. At deeper levels ($l \in \{3, 4\}$), features encode higher-level semantics. Instead of explicit differencing, we preserve their structure

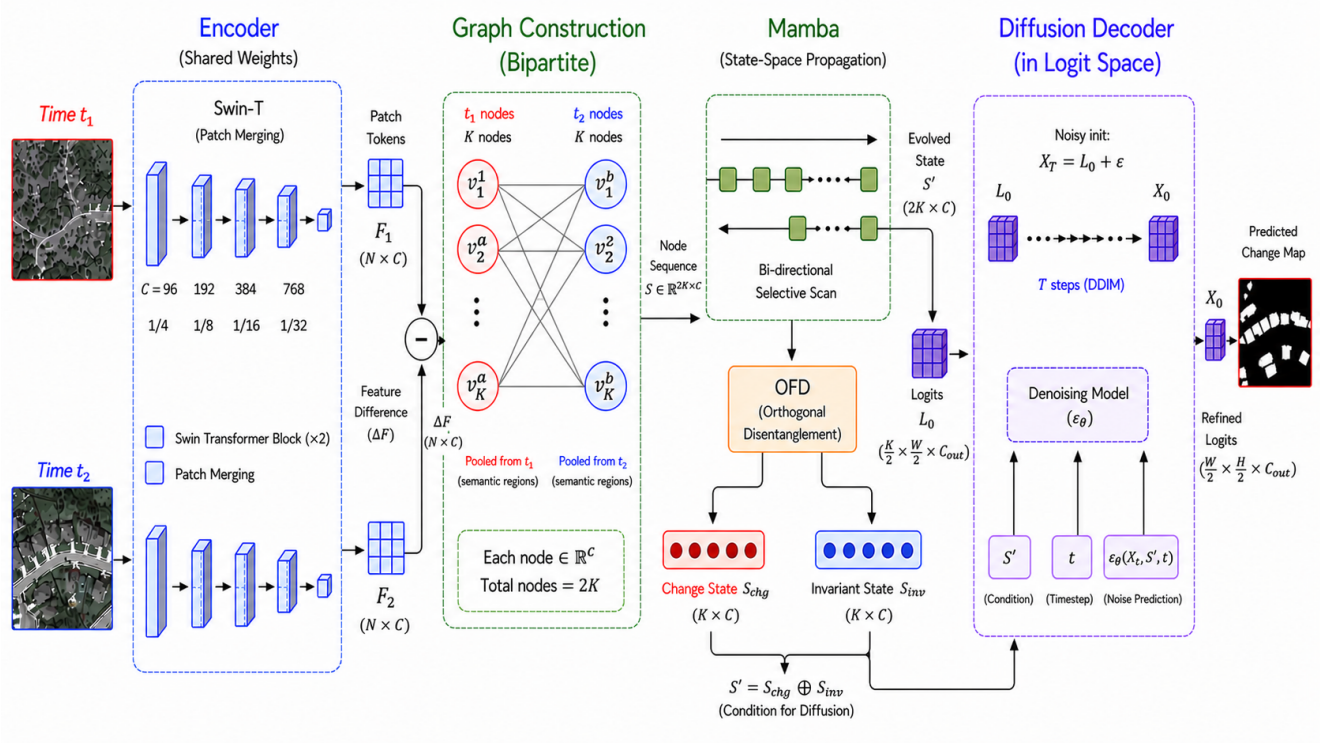


Figure 2. Overview of the proposed GMD-CD framework. Bi-temporal images are encoded into multi-scale features, followed by structured cross-temporal modeling using the Bipartite Graph-Mamba Operator (BGMO). The resulting representations are disentangled into change and invariant components via OFD and fused to produce coarse logits. A conditional diffusion decoder then refines the logits in logit space, improving boundary quality and reducing false detections.

and project them into a shared state-space:

$$S_l^a = W_l f_l^a, \quad S_l^b = W_l f_l^b, \tag{3}$$

where $W_l$ aligns feature dimensions across scales. This design yields multi-scale states $\{S_l\}$ that capture both local variations and global semantics, providing stable inputs for structured cross-temporal state propagation.

### 3.3. Bipartite Graph-Mamba Operator (BGMO)

As illustrated in Fig. 3, BGMO organizes bi-temporal features into a structured bipartite representation and performs bidirectional state-space propagation over the resulting node sequence. At deeper scales ($l \in \{3, 4\}$), we model cross-temporal interaction as a *structured state-space evolution over a bipartite representation*. Given projected features $q^a = W_a f_l^a$ and $q^b = W_b f_l^b$ (Alg. 1, l.1), we aggregate spatial regions into node embeddings $\mathbf{N}^a, \mathbf{N}^b \in \mathbb{R}^{B \times K \times C}$ via adaptive pooling (l.2–3). The two partitions correspond to $t_1$ and $t_2$, and are concatenated into an ordered sequence $\mathbf{S} = [\mathbf{N}^a; \mathbf{N}^b]$ (l.4), which explicitly preserves temporal identity while enabling joint processing. Interaction is performed using a bidirectional selective state-space model (Mamba) [20] (l.6–8). Unlike prior SSM-based CD methods that process flattened, unstructured sequences, the proposed bipartite ordering induces structured cross-temporal coupling: forward and backward scans repeatedly traverse both partitions, enabling each node to incorporate information from both temporal partitions under bidirectional traversal. This does not explicitly mask intra-

**Algorithm 1** Bipartite Graph-Mamba Operator (BGMO)

**Input:** $f_l^a, f_l^b \in \mathbb{R}^{C \times H_l \times W_l}$
**Output:** $S_l' \in \mathbb{R}^{C \times H_l \times W_l}$

1: $q^a \leftarrow W_a f_l^a, \quad q^b \leftarrow W_b f_l^b$
2: $\mathbf{N}^a \leftarrow \text{AdaptivePool}(q^a, \sqrt{K} \times \sqrt{K})$
3: $\mathbf{N}^b \leftarrow \text{AdaptivePool}(q^b, \sqrt{K} \times \sqrt{K})$
4: $\mathbf{S} \leftarrow [\mathbf{N}^a; \mathbf{N}^b]$
5: $\mathbf{Z} \leftarrow \text{Linear}(\text{SiLU}(\text{Linear}(\text{LayerNorm}(\mathbf{S}))))$
6: $\mathbf{Z}^{\rightarrow} \leftarrow \text{SSM}(\mathbf{Z})$
7: $\mathbf{Z}^{\leftarrow} \leftarrow \text{reverse}(\text{SSM}(\text{reverse}(\mathbf{Z})))$
8: $\mathbf{S}^{(1)} \leftarrow \mathbf{Z}^{\rightarrow} + \mathbf{Z}^{\leftarrow} + \mathbf{S}$
9: $\mathbf{R} \leftarrow \text{Linear}(\text{GELU}(\text{Linear}(\text{LayerNorm}(\mathbf{S}^{(1)}))))$
10: $\mathbf{S}^{(2)} \leftarrow \mathbf{R} + \mathbf{S}^{(1)}$
11: $(\mathbf{S}_{\text{chg}}, \mathbf{S}_{\text{inv}}) \leftarrow \text{ChannelSplit}(\mathbf{S}^{(2)})$
12: $\mathbf{U} \leftarrow \text{Upsample}(\text{LinearExpand}(\mathbf{S}_{\text{chg}}))$
13: $\mathbf{U} \leftarrow \text{SiLU}(\text{LayerNorm}(\text{Conv}_{3\times3}(\mathbf{U})))$
14: $S_l' \leftarrow f_l^a + f_l^b + \gamma \cdot \mathbf{U}$

partition interactions, but biases propagation toward cross-temporal mixing through repeated traversal under bidirectional dynamics. As a result, global interaction is achieved with linear complexity, without pairwise attention or explicit graph construction. The propagated sequence is refined through residual feed-forward transformation (l.9–10) and decomposed into complementary subspaces (l.11). The change component is projected back to the spatial domain (l.12–13) and fused with the original features via a gated residual (l.14), enabling stable optimization and progressive integration of propagated context. Overall, BGMO reformulates cross-temporal reasoning as structured state evolution over a bipartite sequence, providing a structured alternative to unstructured sequence modeling in SSM-based change detection while maintaining efficiency.

### 3.4. Orthogonal Feature Disentanglement (OFD)

The BGMO representation encodes both change-sensitive variations and invariant contextual structure, which can interfere during prediction. We introduce *Orthogonal Feature Disentanglement (OFD)* to explicitly separate these components at the node level, enabling stable state evolution and reducing pseudo-change leakage. Given $\mathbf{S} \in \mathbb{R}^{B\times 2K\times C}$, we partition channels into complementary subspaces $(\mathbf{S}_{\text{chg}}, \mathbf{S}_{\text{inv}})$ with a fixed channel partition. To enforce separation, we apply channel-wise $\ell_2$ normalization and impose an orthogonality constraint computed per node and averaged over the batch:

$$\mathcal{L}_{\text{orth}} = \left\|\mathbf{S}_{\text{chg}}^{\top}\mathbf{S}_{\text{inv}}\right\|_F^2, \tag{4}$$

which suppresses shared directions and reduces feature leakage between subspaces. To avoid degenerate solutions, we regularize the change component with $\mathcal{L}_{\text{var}} = \|\mathbf{S}_{\text{chg}}\|_F^2$, ensuring it remains informative. The subspaces are used asymmetrically: $\mathbf{S}_{\text{chg}}$ is projected back to the spatial domain for prediction, while $\mathbf{S}_{\text{inv}}$ is retained in the residual pathway to preserve context. This separation is necessary as BGMO propagates features across time; without disentanglement, invariant responses can contaminate change signals during propagation. By enforcing orthogonal subspaces on graph-evolved features, OFD stabilizes representation dynamics and improves robustness to appearance variations in subsequent refinement.

### 3.5. Hierarchical Fusion and Logit Projection

The graph-evolved features $\{S'_l\}_{l=1}^4$ form a multi-scale hierarchy, where deeper levels encode global context and shallower levels retain spatial detail. We adopt a top-down fusion scheme to propagate high-level information to higher-resolution features, similar to feature pyramid designs [27]. Fusion is performed recursively:

$$\tilde{S}_l = S'_l + \phi\Big(\text{Up}(\tilde{S}_{l+1})\Big), \quad l \in \{3,2,1\}, \tag{5}$$

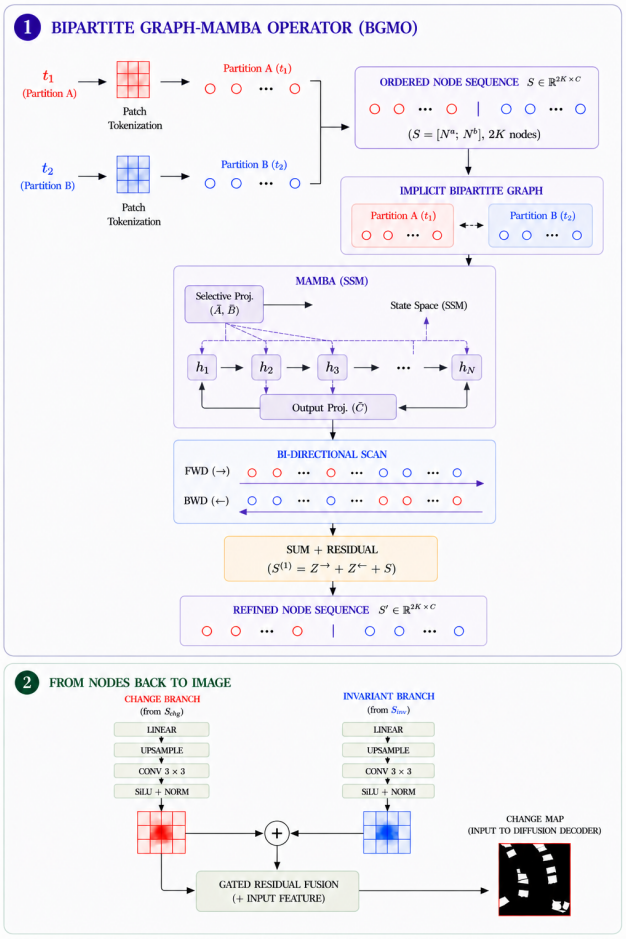


Figure 3. Architecture of Bipartite Graph-Mamba Operator (BGMO).

with $\tilde{S}_4 = S'_4$. Here, $\text{Up}(\cdot)$ denotes bilinear upsampling, and $\phi(\cdot)$ is a $3\times 3$ convolution for channel alignment and local smoothing. This yields a consistent multi-scale representation where coarse features provide global priors for finer scales. The final fused feature $\tilde{S}_1 \in \mathbb{R}^{C\times H\times W}$ is projected to logit space:

$$L_0 = W_o\tilde{S}_1, \quad W_o \in \mathbb{R}^{1\times C\times 3\times 3}, \tag{6}$$

producing a coarse change logit map $L_0 \in \mathbb{R}^{1\times H\times W}$. This hierarchical fusion integrates global context with local detail, providing a stable and informative initialization for subsequent diffusion-based refinement.

### 3.6. Conditional Diffusion Decoder in Logit Space

As shown in Fig. 4, we formulate refinement as a conditional diffusion process operating directly in logit space. Given the fused representation $\tilde{S}_1$, we obtain a coarse logit map $L_0 \in \mathbb{R}^{1\times H\times W}$ that captures globally consistent change structure from BGMO. However, $L_0$ is deter-

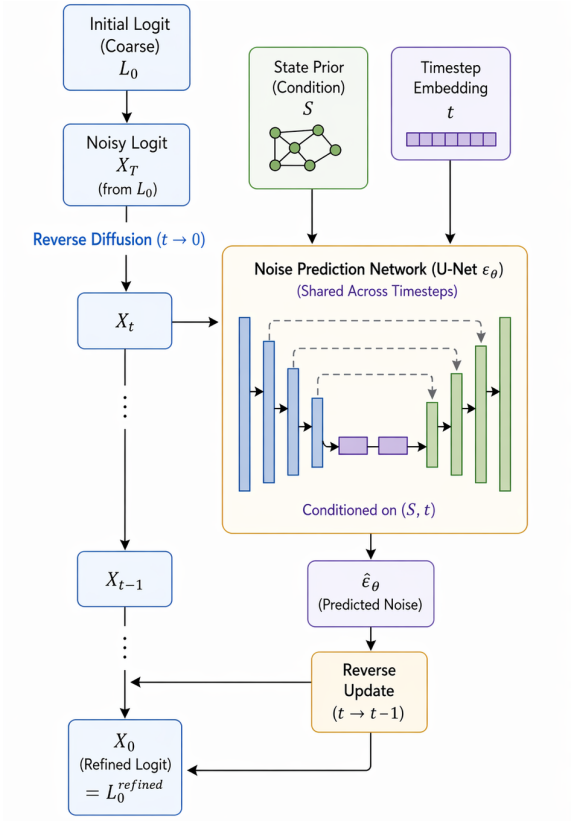


Figure 4. Architecture of Conditional Diffusion Decoding.

ministic and tends to be over-confident in ambiguous regions such as boundaries and small objects. To address this, we formulate refinement as a *conditional diffusion process* directly in logit space. We adopt a DDPM formulation [23, 33] conditioned on the graph-evolved state prior $S \in \mathbb{R}^{C\times H\times W}$. Unlike prior diffusion-based CD methods that perform full generative modeling [1], we restrict diffusion to a *residual correction regime* around $L_0$. During training, the model learns to predict noise from perturbed logits; at inference, we initialize from $x_T = L_0 + \epsilon$, ensuring the trajectory remains close to a meaningful solution. The denoiser $\epsilon_\theta(x_t, S, t)$ is implemented as a lightweight U-Net and conditioned via feature concatenation with a projected state prior, enabling each step to be guided by globally structured context. Training uses noisy logits derived from the same forward pipeline, reducing train–test mismatch when initialized from $L_0$. We employ a deterministic DDIM sampler [37] with a small number of steps ($T = 5$), resulting in efficient inference. In contrast to deterministic decoders that produce point estimates, diffusion captures uncertainty through stochastic perturbation and iterative denoising, enabling targeted correction in regions where predictions are unreliable. Operating in logit space constrains the process to refinement ($x_0 = L_0 + \Delta L$), where $x_0$ denotes the ground-truth logit obtained via inverse sigmoid of the binary mask. This helps preserve global semantics while correcting high-frequency errors. This design complements BGMO: structured state propagation ensures global consistency, while diffusion performs uncertainty-aware local refinement, improving boundary fidelity and reducing false positives without the cost of full generative diffusion.

# 4. Experiments

## 4.1. Datasets

We evaluate our method on four widely used remote sensing change detection benchmarks: *LEVIR-CD* [45], *DSIFN-CD* [39], *WHU-CD* [24], and *CDD* [6], which collectively cover building-level changes, multi-class land-cover variations, and seasonal transformations. LEVIR-CD consists of 637 high-resolution ($1024 \times 1024$, 0.5 m/pixel) Google Earth image pairs with annotated building changes, which are partitioned into non-overlapping $256 \times 256$ patches following standard practice (7120/1024/2048 for training/validation/testing). DSIFN-CD provides large-scale multi-class urban change data with $512 \times 512$ images, similarly divided into $256 \times 256$ patches using the official split (14,400/1,360/192). WHU-CD contains bi-temporal aerial imagery for building change detection, where we adopt the commonly used patch-based preprocessing and standard splits. CDD includes diverse change scenarios such as buildings, vehicles, and seasonal variations from Google Earth imagery, and is processed using the same $256 \times 256$ patch protocol with standard evaluation splits. This preprocessing ensures consistent training and fair comparison.

## 4.2. Implementation Details

We use a **Swin Transformer-Tiny (Swin-T)** backbone [29], pretrained on ImageNet-1K, as the shared multi-scale encoder to extract hierarchical features for graph-based state evolution. Optimization is performed using **AdamW** [30] with ($\beta_1 = 0.9$, $\beta_2 = 0.999$) and weight decay $1 \times 10^{-4}$. A **OneCycle learning rate schedule** is adopted with a maximum learning rate of $3 \times 10^{-4}$, 10% warmup, and cosine annealing. Training is conducted with a batch size of 8 using **automatic mixed precision (FP16)** for 125 epochs. The segmentation objective combines binary cross-entropy and Dice loss, $\mathcal{L}_{\text{seg}} = \mathcal{L}_{\text{BCE}} + \lambda_{\text{Dice}}\mathcal{L}_{\text{Dice}}$, and the overall objective is $\mathcal{L} = \mathcal{L}_{\text{seg}} + \lambda_{\text{orth}}\mathcal{L}_{\text{orth}} + \lambda_{\text{var}}\mathcal{L}_{\text{var}} + \lambda_{\text{diff}}\mathcal{L}_{\text{diff}}$. For the diffusion decoder, we follow the DDPM formulation [22, 33] with $T = 100$ forward steps and a cosine noise schedule, and perform efficient **5-step DDIM sampling** [36] during inference. All experiments are conducted on a single **NVIDIA RTX 4090 GPU (24GB)**. We report four standard evaluation metrics: F1-score, Intersection-over-Union (IoU), Overall Accuracy

Table 1. Quantitative comparison on WHU-CD, DSIFN-CD, LEVIR-CD and CDD datasets. Best results in **bold**, second-best underlined.

| Method | Extra data | WHU-CD | | | DSIFN-CD | | | LEVIR-CD | | | CDD | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | F1 | IoU | OA | F1 | IoU | OA | F1 | IoU | OA | F1 | IoU | OA |
| **CNN-based Methods** | | | | | | | | | | | | | |
| FC-Siam-conc [15] | None | 0.798 | 0.665 | 98.5 | 0.597 | 0.426 | 87.6 | 0.837 | 0.720 | 98.5 | 0.751 | 0.601 | 94.9 |
| SNUNet [18] | None | 0.835 | 0.717 | 98.7 | 0.662 | 0.495 | 87.3 | 0.882 | 0.788 | 98.8 | 0.839 | 0.721 | 96.2 |
| IFNet [40] | IN1k | 0.834 | 0.715 | 98.8 | 0.601 | 0.430 | 87.8 | 0.881 | 0.788 | 98.9 | 0.840 | 0.719 | 96.03 |
| **CNN + Attention Methods** | | | | | | | | | | | | | |
| DT-SCN [28] | IN1k | 0.914 | 0.842 | 99.3 | 0.706 | 0.545 | 82.9 | 0.877 | 0.781 | 98.8 | 0.921 | 0.853 | 98.2 |
| STANet [5] | IN1k | 0.823 | 0.700 | 98.5 | 0.645 | 0.478 | 88.5 | 0.873 | 0.774 | 98.7 | 0.841 | 0.722 | 96.1 |
| **Transformer-based Methods** | | | | | | | | | | | | | |
| BIT [8] | IN1k | 0.905 | 0.834 | 99.3 | 0.876 | 0.780 | 92.3 | 0.893 | 0.807 | 98.92 | 0.889 | 0.800 | 97.5 |
| ChangeFormer [3] | None | 0.886 | 0.795 | 99.12 | 0.947 | 0.887 | 93.2 | 0.904 | 0.825 | 99.0 | 0.946 | 0.898 | 98.7 |
| SAM2-CD [35] | SAM-pretrained | 0.926 | 0.889 | 99.77 | – | – | – | 0.919 | 0.855 | 99.19 | – | – | – |
| PeftCD [17] | IN1k | 0.959 | 0.9205 | 99.6 | – | – | – | 0.923 | 0.856 | 99.2 | 0.985 | 0.970 | 99.63 |
| **Self-supervised Pretraining** | | | | | | | | | | | | | |
| SiamSiam [42] | IN1k, IBSD, GoogleEarth | 0.847 | 0.734 | – | – | – | – | 0.880 | 0.786 | – | – | – | – |
| MoCo-v2 [13] | IN1k, IBSD, GoogleEarth | 0.882 | 0.789 | – | – | – | – | 0.879 | 0.784 | – | – | – | – |
| SeCo [32] | IN1k, IBSD, GoogleEarth | 0.883 | 0.790 | – | – | – | – | 0.881 | 0.787 | – | – | – | – |
| SaDL-CD [26] | IN1k, IBSD, GoogleEarth | 0.909 | 0.833 | – | – | – | – | 0.899 | 0.818 | – | – | – | – |
| **Diffusion-based Methods** | | | | | | | | | | | | | |
| DDPM-CD [1] | GoogleEarth | 0.927 | 0.863 | 99.4 | 0.967 | 0.913 | 97.1 | 0.909 | 0.833 | 99.1 | 0.956 | 0.916 | 99.0 |
| DiffRegCD [31] | GoogleEarth | 0.934 | 0.883 | 99.0 | 0.940 | 0.890 | 96.7 | 0.929 | 0.881 | 98.7 | – | – | – |
| **Mamba-based Methods** | | | | | | | | | | | | | |
| RSMamba [11] | IN1k | 0.927 | 0.865 | 99.4 | 0.965 | 0.913 | 97.0 | 0.897 | 0.814 | 98.9 | 0.943 | 0.902 | 98.8 |
| ChangeMamba [9] | IN1k | 0.925 | 0.861 | 99.4 | 0.875 | 0.778 | 95.8 | 0.902 | 0.821 | 99.0 | 0.944 | 0.920 | 99.0 |
| CDMamba [43] | IN1k | 0.937 | 0.882 | 99.5 | 0.966 | 0.914 | 97.0 | 0.907 | 0.831 | 99.0 | 0.960 | 0.919 | 99.1 |
| M-CD [34] | IN1k | 0.953 | 0.911 | 99.6 | 0.970 | 0.935 | 98.9 | 0.921 | 0.850 | 99.2 | 0.982 | 0.963 | 99.5 |
| **Boundary-focused Methods** | | | | | | | | | | | | | |
| LRNet [46] | IN1k | 0.925 | 0.861 | 99.47 | – | – | – | 0.911 | 0.836 | 99.10 | – | – | – |
| *Ours* | | | | | | | | | | | | | |
| **GMD-CD** | IN1k | **0.962** | **0.921** | **99.78** | **0.978** | **0.948** | **99.12** | **0.9375** | **0.882** | **99.35** | **0.990** | **0.975** | **99.64** |

(OA), and Boundary F1 (B-F1). B-F1 is computed using a 3-pixel tolerance following [14], measuring the alignment between predicted and ground-truth boundaries.

### 4.3. Comparison with State-of-the-Art

Tab. 1 reports quantitative comparisons across four benchmarks. CNN-based methods (e.g., FC-Siam-conc, SNUNet [15, 18]) are limited by local receptive fields and underperform on complex scenes. Attention-based models (STANet [5], BIT [8], ChangeFormer [3]) improve global context modeling but incur higher complexity. Recent approaches explore complementary directions: foundation-based models such as SAM2-CD [35] and adaptation-based methods such as PeftCD [17] leverage large pretrained priors, while state-space models (RSMamba, ChangeMamba, CDMamba, M-CD [10, 12, 34, 44]) provide efficient long-range modeling. Diffusion-based methods (DDPM-CD, DiffRegCD [1, 31]) improve refinement quality, and LR-Net [46] focuses on boundary modeling. However, these approaches typically improve either global interaction or refinement in isolation. In contrast, GMD-CD jointly models structured cross-temporal interaction and uncertainty-aware refinement. The BGMO module induces structured propagation across temporal partitions with linear complexity, while the diffusion decoder performs logit-space correction guided by global context. This combination improves both region accuracy and boundary quality. As shown in Tab. 1, GMD-CD achieves the best performance across all datasets, improving both F1 and IoU over CNN and transformer baselines and outperforming recent state-space and diffusion-based methods. Compared to strong baselines such as M-CD and PeftCD, the proposed model yields consistent gains, indicating that structured propagation and logit-space refinement are complementary. Qualitative results in Fig. 5 further support these findings: compared to M-CD and DDPM-CD, GMD-CD produces more coherent regions with sharper boundaries and fewer false positives, particularly in densely structured scenes.

### 4.4. Ablation and Analysis

Table 2 quantifies the role of each component on LEVIR-CD and WHU-CD. The Mamba-only baseline captures global context (88.30/93.85 F1) but exhibits poor boundary quality (66.50/80.10 B-F1), indicating that unstructured sequence propagation is insufficient for precise localization. Introducing structured interaction (BGMO) im-

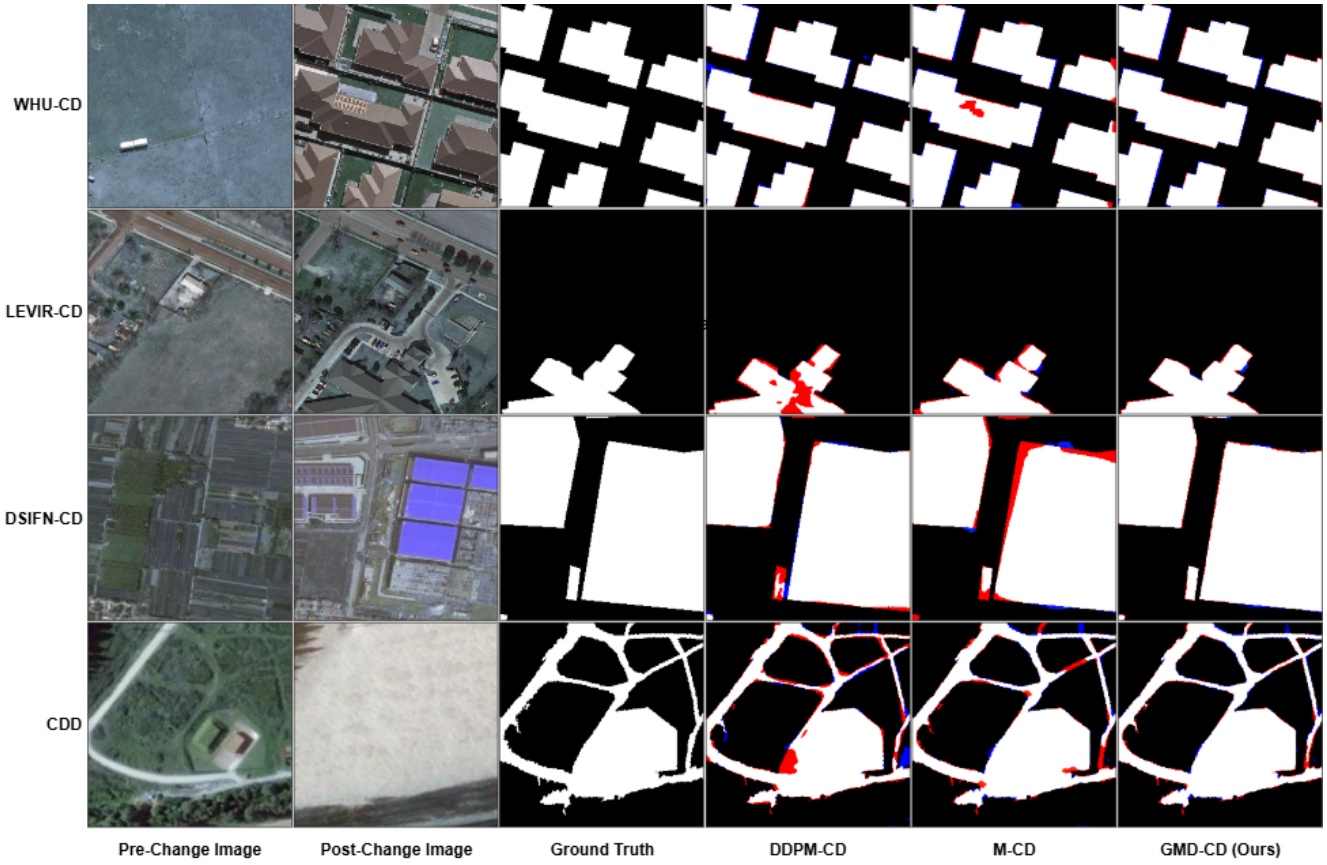


Figure 5. Change map comparison. The six columns show: Time 1 image, Time 2 image, Ground Truth, DDPM-CD, M-CD, and GMD-CD (Ours). In the error overlay, white denotes true positives (TP), black denotes true negatives (TN), red denotes false positives (FP), and blue denotes false negatives (FN).

proves recall (88.10 → 90.95 on LEVIR; 93.50 → 95.80 on WHU) and increases F1 to 90.04/95.45, showing that partition-aware ordering improves coverage of spatially distributed changes. Diffusion, in contrast, yields large gains in boundary quality (66.50 → 84.10 on LEVIR; 80.10 → 88.60 on WHU) with smaller F1 improvements, indicating uncertainty-aware refinement rather than primary prediction. Combining both leads to consistent gains (92.34/96.40 F1 and 85.90/89.40 B-F1), demonstrating complementary behavior. Finally, adding OFD further improves all metrics (93.750/96.20 F1 and 87.40/91.20 B-F1), confirming that disentangling change and invariant features reduces interference during propagation. Overall, structured propagation improves semantic completeness, diffusion refines uncertain regions, and OFD stabilizes feature dynamics, leading to joint gains in region accuracy and boundary quality.

### 4.5. Graph & Diffusion Analysis

Table 3 analyzes the sensitivity of the structured state-space module and diffusion refinement. Performance peaks at $K = 64$, indicating a balance between spatial abstraction and representation capacity; smaller $K$ reduces spatial coverage, while larger $K$ approaches dense tokenization with diminishing returns. Average pooling consistently outperforms max pooling, suggesting that stable region aggregation is preferable to sparse activation selection. The $[a; b]$ ordering improves performance over interleaved and random layouts, confirming that preserving temporal partitioning influences cross-temporal interaction within the state-space formulation. Bidirectional scanning further improves results over forward-only scanning, indicating that symmetric traversal enhances information exchange across partitions. For diffusion, increasing DDIM steps from 1 to 5

Table 2. Component-wise ablation on LEVIR-CD and WHU-CD.

| Mamba | Graph | Diff | OFD | LEVIR-CD | | | | WHU-CD | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | Prec. | Rec. | F1 | B-F1 | Prec. | Rec. | F1 | B-F1 |
| ✓ | × | × | × | 88.50 | 88.10 | 88.30 | 66.50 | 94.20 | 93.50 | 93.85 | 80.10 |
| ✓ | ✓ | × | × | 89.15 | 90.95 | 90.04 | 69.10 | 95.10 | 95.80 | 95.45 | 82.30 |
| ✓ | × | ✓ | × | 90.50 | 90.10 | 90.30 | 84.10 | 95.80 | 94.70 | 95.25 | 88.60 |
| ✓ | ✓ | ✓ | × | 90.98 | 93.750 | 92.34 | 85.90 | 94.10 | 94.70 | 95.40 | 89.40 |
| ✓ | ✓ | ✓ | ✓ | **93.41** | **94.04** | **93.75** | **87.40** | **95.80** | **96.17** | **96.20** | **91.20** |

Table 3. BGMO sensitivity analysis on LEVIR-CD. Default: $K = 64$, avg pooling, $[a; b]$, bidirectional scan, 5-step DDIM.

| Configuration | F1 | Δ F1 |
|---|---|---|
| *Number of graph nodes K* | | |
| $K = 16$ | 92.35 | −0.75 |
| $K = 64$ [default] | **93.750** | — |
| $K = 256$ | 92.88 | −0.22 |
| *Node construction* | | |
| Max pooling | 92.60 | −0.50 |
| Average pooling [default] | **93.750** | — |
| *Sequence ordering* | | |
| Interleaved | 92.45 | −0.65 |
| $[a; b]$ [default] | **93.750** | — |
| Random | 91.80 | −1.30 |
| *Scan direction* | | |
| Forward | 92.30 | −0.80 |
| Bidirectional [default] | **93.750** | — |
| *Diffusion steps (DDIM)* | | |
| 1 step | 92.62 | −0.48 |
| 3 steps | 92.95 | −0.15 |
| 5 steps [default] | **93.750** | — |
| 10 steps | 93.754 | +0.04 |

improves F1, while further increases yield marginal gains. This behavior indicates that a small number of denoising iterations is sufficient for correcting high-frequency errors. We therefore use 5 steps as an effective trade-off between accuracy and efficiency. Overall, the results show that structured state propagation governs global consistency, while diffusion provides targeted refinement, and OFD stabilizes feature interactions under varying configurations. Unlike deterministic refiners, diffusion corrects predictions conditioned on uncertainty, which is reflected in larger gains in B-F1 compared to F1.

### 4.6. Efficiency Analysis

We evaluate computational efficiency using three metrics: number of trainable parameters, GFLOPs, and average inference time per image pair. Results on LEVIR-CD and WHU-CD are reported in Table 4. All measurements are conducted on a single NVIDIA RTX 4090 GPU with input resolution $256 \times 256$. For methods without reported latency, inference time is estimated from GFLOPs under the same hardware setting, and may vary across implementations. The proposed model without diffusion requires 13.2 GFLOPs and 22 ms inference time, achieving 0.913 F1 on LEVIR-CD. Incorporating diffusion increases computation to 32.09 GFLOPs and 47 ms, while improving performance to 0.9375 (LEVIR-CD) and 0.962 (WHU-CD). GFLOPs include all components, including the 5-step DDIM refinement. Compared to ChangeFormer, the proposed model achieves higher accuracy with substantially lower computational cost and reduced latency. Lightweight models such as CDMamba and PeftCD achieve lower FLOPs but exhibit reduced accuracy, indicating limited modeling capacity. In contrast, M-CD attains competitive performance at significantly higher latency (160 ms), reflecting an unfavorable efficiency trade-off. High-capacity approaches such as SAM2-CD and DDPM-CD incur substantially higher computational cost due to large-scale pretrained backbones or full diffusion processes. Overall, GMD-CD operates at moderate computational cost while achieving the best accuracy across datasets. The gains from diffusion are obtained at the cost of moderate additional latency, consistent with its role as a lightweight refinement stage rather than a full generative process. This results in a favorable accuracy–efficiency trade-off, combining structured state-space interaction with efficient logit-space refinement.

Table 4. Computational complexity comparison on LEVIR-CD and WHU-CD. Models are ordered by increasing GFLOPs.

| Method | Params(M) | GFLOPs | Avg. Inf. Time | $F1_L$ | $F1_W$ |
|---|---|---|---|---|---|
| CDMamba [43] | 11.90 | 10.4 | 15 ms | 0.907 | 0.937 |
| PeftCD [17] | 10.15 | 22.0 | 25 ms | 0.923 | 0.959 |
| M-CD [34] | 69.80 | 29.58 | 160 ms | 0.921 | 0.953 |
| LRNet [46] | 48.71 | 92.23 | 60 ms | 0.911 | 0.925 |
| ChangeFormer [3] | 41.02 | 254.8 | 85 ms | 0.904 | 0.886 |
| SAM2-CD [35] | 2.59 | 1193.83 | 95 ms | 0.919 | 0.926 |
| DDPM-CD [1] | 46.41 | 2175.46 | 88.10 ms | 0.909 | 0.927 |
| **Ours (w/o Diff)** | 30.7 | 13.2 | 22 ms | 0.913 | 0.9589 |
| **Ours (full)** | **33.55** | 32.09 | 47 ms | **0.9375** | **0.962** |

## 5. Conclusion

We introduced **GMD-CD**, a unified framework for remote sensing change detection that combines structured state-space interaction with logit-space diffusion refinement. The proposed BGMO module models cross-temporal relationships through a bipartite structured sequence and bidirectional state-space propagation, enabling global interaction with linear complexity. Complementing this, the conditional diffusion decoder refines coarse predictions directly in logit space, improving boundary quality without the overhead of full generative diffusion. Extensive experiments across four benchmarks demonstrate consistent gains over CNN, transformer, and recent state-space and diffusion-based methods. Ablation results further show that structured state propagation improves semantic completeness, while diffusion provides targeted boundary refinement, indicating complementary behavior. Overall, the results suggest that combining structured state-space modeling with lightweight refinement is an effective direction for high-resolution change detection. Future work will explore extensions to multi-class settings, adaptive node construction, and stronger pretraining strategies for state representations.

## References

[1] Chaminda Bandara and Vishal Patel. Ddpm-cd: Diffusion-based change detection. In *CVPR*, 2022. 1, 2, 5, 6, 8

[2] Wele Gedara Chaminda Bandara and Vishal M. Patel. A transformer-based siamese network for change detection. In *ICIP*, 2022. 1

[3] Wele Gedara Chaminda Bandara and Vishal M. Patel. Changeformer. In *CVPR*, 2022. 1, 2, 6, 8

[4] Hao Chen and Zhenwei Shi. A spatial-temporal attention-based method and a new dataset for remote sensing image change detection. In *CVPR*, 2020. 1, 2

[5] Hao Chen and Zhenwei Shi. Stanet: Spatial-temporal attention network. In *CVPR*, 2020. 6

[6] Hao Chen, Zhi Zhang, et al. Change detection dataset for remote sensing images. *Remote Sensing*, 2020. 5

[7] Hao Chen, Zipeng Qi, and Zhenwei Shi. Remote sensing image change detection with transformers. In *ICCV*, 2021. 1, 2

[8] Hao Chen, Zipeng Qi, and Zhenwei Shi. Bit: Bitemporal image transformer. In *CVPR*, 2022. 2, 6

[9] Hongruixuan Chen, Jian Song, Chengxi Han, Junshi Xia, and Naoto Yokoya. Changemamba. In *CVPR*, 2024. 6

[10] Hongruixuan Chen, Jian Song, Chengxi Han, Junshi Xia, and Naoto Yokoya. Changemamba: State space models for change detection. In *CVPR*, 2024. 1, 2, 6

[11] Keyan Chen, Bowen Chen, Chenyang Liu, Wenyuan Li, Zhengxia Zou, and Zhenwei Shi. Rsmamba. In *CVPR*, 2024. 6

[12] Keyan Chen, Bowen Chen, Chenyang Liu, Wenyuan Li, Zhengxia Zou, and Zhenwei Shi. Rsmamba: Multi-directional state space models for remote sensing. In *CVPR*, 2024. 2, 6

[13] Xinlei Chen. Moco v2. In *CVPR*, 2020. 6

[14] Gabriela Csurka, Diane Larlus, Florent Perronnin, and Fabien Meylan. What is a good evaluation measure for semantic segmentation? In *BMVC*, 2013. 6

[15] Rodrigo Daudt, Bertrand Le Saux, and Alexandre Boulch. Fully convolutional siamese networks for change detection. In *ICIP*, 2018. 6

[16] Rodrigo Caye Daudt, Bertrand Le Saux, Alexandre Boulch, and Yann Gousseau. Fully convolutional siamese networks for change detection. In *ICIP*, 2018. 1, 2

[17] Sijun Dong, Yuxuan Hu, Libo Wang Wang, Geng Chen, and Xiaoliang Meng. Peftcd: Leveraging vision foundation models with parameter-efficient fine-tuning for remote sensing change detection. *TGRS*, 2026. 6, 8

[18] Shuai Fang and Kun Fu. Snunet-cd: A densely connected siamese network for change detection. In *IEEE TIP*, 2021. 6

[19] Shuai Fang, Kaiyu Li, Jinyuan Shao, and Zhe Li. Snunet-cd: A densely connected siamese network for change detection. *IEEE Geoscience and Remote Sensing Letters*, 2022. 1, 2

[20] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. In *ICLR*, 2024. 1, 2, 3

[21] Rishabh Gupta, Manish Shah, et al. Creating xbd: A dataset for assessing building damage. *CVPR*, 2019. 1

[22] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 5

[23] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 5

[24] Shunping Ji, Shuang Jiang, and Yubing Lu. Fully convolutional networks for multisource building change detection using very high resolution images. *Pattern Recognition*, 2019. 5

[25] Yilmaz Korkmaz and Vishal M. Patel. Remotevar: Autoregressive visual modeling for remote sensing. In *CVPR*, 2024. 2

[26] Yunsong Li, Chao Zhang, Kai Chen, and Aojie Li. Sadl-cd. In *CVPR*, 2023. 6

[27] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017. 4

[28] Jing Liu, Lei Liu, and Yu Zheng. Dt-scn: Deep temporal siamese network. In *CVPR*, 2020. 6

[29] Ze Liu, Yutong Lin, Yue Cao, et al. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 2021. 2, 5

[30] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2017. 5

[31] Seyedehanita Madani, Rama Chellappa, and Vishal M. Patel. Diffregcd: Integrated registration and change detection with diffusion features. In *CVPR*, 2024. 6

[32] Oğuzhan Fatih Manas, Jean-Francois Todorovic, Shreyas Kulkarni, and Julien Cornebise. Seco: Seco self-supervised learning. In *CVPR*, 2021. 6

[33] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *ICML*, 2021. 5

[34] Jay N. Paranjape, Celso M. de Melo, and Vishal M. Patel. M-cd. In *WACV*, 2026. 1, 6, 8

[35] Yuan Qin, Chaoting Wang, Yuanyuan Fan, and Chanling Pan. Sam2-cd: Remote sensing image change detection with sam2. *TGRS*, 2025. 1, 2, 6, 8

[36] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*, 2021. 5

[37] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*, 2021. 5

[38] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *ECCV*, 2018. 2

[39] Chen Wu, Bin Du, and Liangpei Zhang. A benchmark dataset for change detection in remote sensing images. *TGRS*, 2021. 5

[40] Jian Zhang, Qingyun Li, and Guohua Cao. Ifnet for change detection. In *CVPR*, 2023. 6

[41] Jian Zhang, Qingyun Li, and Guohua Cao. Next2former-cd: Convnext meets mask2former for change detection. In *CVPR*, 2024. 2

[42] Ming Zhang, Xiang He, and Zhaohui Li. Siamsiam self-supervised change detection. In *CVPR*, 2022. 6

[43] Y. Zhang and et al. Cdmamba. In *CVPR*, 2024. 6, 8

[44] Y. Zhang and et al. Cdmamba: Bidirectional state space models for change detection. In *CVPR*, 2024. 1, 2, 6

[45] Zhengxin Zhang, Qingjie Liu, and Yunhong Wang. Levir-cd: A large-scale remote sensing dataset for building change detection. In *CVPR*, 2020. 5

[46] Huan Zhong, Chen Wu, and Ziqi Xiao. Lrnet: Change detection in high-resolution remote sensing imagery via a localization-then-refinement strategy. In *Remote Sensing*, 2023. 6, 8

[47] Lianghui Zhu, Biao Wang, Tang Tang, Xiaofan Zhu, and et al. Vision mamba: Efficient vision backbone with state space model. *arXiv*, 2024. 2