POCA-lite: A Lightweight Change Detection Architecture with Geometry-Aware Auxiliary Supervision and Feedback Fusion

Yonggi Shi *, Ruopeng Yang, Bo Huang, Zhaoyang Gu, Yiwei Lu, Changsheng Yin, Yonggi Wen, Yihao Zhong

Posted Date: 22 April 2026

doi: 10.20944/preprints202604.1523.v1

Keywords: change detection; remote sensing; lightweight network; boundary-aware supervision; auxiliary heads; feedback fusion; LEVIR-CD

 doi:10.20944/preprints202604.1523.v1

Article

POCA-lite: A Lightweight Change Detection Architecture with Geometry-Aware Auxiliary Supervision and Feedback Fusion

Yongqi Shi [1,*], Ruopeng Yang [2], Bo Huang [1], Zhaoyang Gu [1], Yiwei Lu [2], Changsheng Yin [2], Yongqi Wen [1] and Yihao Zhong [1]

[1] Graduate School, National University of Defense Technology, Wuhan 430035, China
[2] Information Support Force Engineering University, Wuhan 430035, China
* Correspondence: shiyongqi17@nudt.edu.cn

Highlights

What are the main findings?

- POCA-lite, a 1.33 M-parameter encoder–decoder with an inference-coupled geometry branch, matches SNUNet in mean F1 on LEVIR-CD while using 47% fewer parameters and 53% fewer FLOPs.
- Decomposition ablations disentangle two complementary gain sources: geometric supervision alone recovers 85% of the total improvement, while the feedback fusion pathway alone recovers 92%; their combination achieves the full result.
- Boundary F1 improves by 9.22 percentage points over the no-geometry baseline, and cross-architecture transfer to SNUNet yields +1.06 pp F1.

What are the implications of the main findings?

- Inference-coupled geometric supervision is a promising strategy for lightweight, boundary-sensitive building change detection on domains with well-separated morphology.
- Cross-dataset evaluation on WHU-CD reveals that the geometric assumptions degrade on dense, irregular building layouts, establishing clear scope boundaries and guiding practitioners on when to apply or avoid this approach.

Abstract

Building change detection from bi-temporal remote sensing imagery underpins urban planning, infrastructure monitoring, and disaster assessment. Existing deep-learning methods achieve high accuracy but rely on large parameter counts, while pixel-level supervision provides limited boundary guidance. We propose POCA-lite, a lightweight encoder-decoder with an inference-coupled geometry branch: three geometric prediction heads—distance transform, boundary, and center heatmap—whose outputs are fused back into the decoder via a feedback pathway active at both training and inference. On the LEVIR-CD benchmark under a unified retraining protocol, multi-seed evaluation shows that POCA-lite matches SNUNet in mean F1 while using 47% fewer parameters and 53% fewer FLOPs. Boundary F1 improves by 9.22 pp over the no-geometry baseline. Decomposition ablations reveal two complementary improvement sources: geometric supervision alone recovers 85% of the total gain, while the feedback fusion pathway recovers 92%; their combination achieves the full result. Geometry-aware targets outperform a generic multi-task control. Cross-architecture transfer to SNUNet yields +1.06 pp F1. However, cross-dataset evaluation on WHU-CD shows that the method underperforms SNUNet on dense urban morphology, and zero-shot cross-dataset transfer is not established. These results indicate that inference-coupled geometric supervision is effective for lightweight, boundary-sensitive change detection on domains with well-separated building morphology, but its applicability is scope-bounded.

Keywords: change detection; remote sensing; lightweight network; boundary-aware supervision; auxiliary heads; feedback fusion; LEVIR-CD

 doi:10.20944/preprints202604.1523.v1

1. Introduction

Binary change detection from bi-temporal remote sensing imagery is a core task in urban planning, infrastructure monitoring, and disaster assessment [1–3]. Given two co-registered images acquired at different time points, the objective is to produce a pixel-level binary mask indicating where meaningful changes—primarily construction or demolition of buildings—have occurred.

Deep learning has transformed this field, progressing from hand-crafted features [4] through siamese CNNs [5,6] to transformer-based [7,8] and state-space model architectures [9]. A persistent challenge, however, is the trade-off between model complexity and detection accuracy: recent high-performing models often exceed tens of millions of parameters, increasing computational and memory demands [6,10]. *Lightweight* in this context refers specifically to parameter count and FLOPs; actual deployability on memory-constrained edge devices depends on additional factors including peak activation memory, which must be evaluated separately for each target platform.

A second, less frequently discussed limitation lies in the fact that standard pixel-level supervision signals, namely binary cross-entropy and Dice loss, can only provide indirect guidance for optimizing boundary quality. Empirical evidence shows that lightweight models trained exclusively with pixel-level loss functions are prone to generating blurred object boundaries and merging spatially adjacent change regions. This defect directly impairs the F1 metric performance, given that boundary pixels contribute to an overwhelming proportion of false positive and false negative predictions [11,12]. While boundary-aware methods such as BGSNet [11] and DSHA [12] have introduced geometric supervision as a training-time regularizer, these approaches discard the geometric heads at inference, leaving the decoder to rely solely on features shaped during training.

In this work, we propose a novel design paradigm called an inference-coupled geometry branch. This branch contains three geometric prediction heads for distance transform, boundary extraction and center heatmap estimation, all of which remain active during both training and inference. Their outputs are fused back into the decoder via a dedicated feedback pathway, positioning the geometry branch as a core architectural component rather than a disposable training-only auxiliary module. We name this proposed architecture POCA-lite. These three geometric targets are selected for their complementary encoding of change-region geometry: the distance transform captures global shape and interior structure, the boundary head preserves fine-grained local edge details, and the center heatmap provides precise point-level localization. Together, they cover the full geometric spectrum critical to pixel-level F1 performance: boundary accuracy, region completeness, and inter-object separability.

We validate POCA-lite on LEVIR-CD, a challenging benchmark with well-separated suburban buildings where the approach demonstrates strong parameter efficiency: equivalent mean F1 to SNUNet with 47% fewer parameters and 53% fewer FLOPs. Cross-dataset experiments on WHU-CD establish the morphological conditions under which the approach degrades, which we analyse in Section 5. POCA-lite's applicability is *scope-bounded* by design: it is most effective for domains resembling LEVIR-CD's building morphology, and cross-domain deployment without fine-tuning is not established.

The contributions of this work are as follows.

Design contribution:

1. We propose POCA-lite, a lightweight encoder-decoder architecture (1.33M parameters, 3.2 GFLOPs) with an inference-coupled geometry branch. This branch comprises three geometric prediction heads for distance transform, boundary, and center heatmap, whose outputs are fused back into the decoder at test time, rather than being discarded after training. Unlike BGSNet [11] and DSHA [12], where geometric heads serve only as training-time regularisers, POCA-lite retains the geometry branch as a structural decoder input at inference.

Empirical findings:

2. Under a unified retraining protocol, multi-seed evaluation (5 seeds) shows that POCA-lite matches SNUNet in mean F1 (0.8691 ± 0.0041 vs. 0.8697 ± 0.0018, $p = 0.798$, not statistically

 doi:10.20944/preprints202604.1523.v1

significant) while using 47% fewer parameters and 53% fewer FLOPs. Boundary F1 improves by 9.22 pp over the no-geometry baseline.

3. Decomposition ablations disentangle two complementary improvement sources: geometric supervision alone recovers 85% of the total gain (Config B′), and the feedback fusion pathway recovers 92% (Config C); their combination achieves the full result. Geometry-aware targets outperform a generic multi-task control (+2.05 pp vs. +0.42 pp). Cross-architecture transfer to SNUNet-GEO yields +1.06 pp F1.
4. Cross-dataset evaluation on WHU-CD reveals that the geometric assumptions degrade on dense, irregular building morphology (within-dataset F1 = 0.7491 vs. SNUNet 0.7751), establishing clear scope boundaries and identifying the morphological conditions under which the approach should and should not be applied.

The remainder of this paper is organized as follows. Section 2 provides a comprehensive review of related work in change detection, including traditional methods, CNN-based approaches, transformer architectures, instance-level detection, and geometry-aware segmentation. Section 3 describes the proposed methodology, including architecture details, auxiliary head designs, and the joint optimization framework. Section 4 presents experimental results, ablation studies, and detailed analysis. Section 5 discusses the implications of our findings, comparisons with related approaches, and limitations. Section 6 concludes the paper and outlines future research directions.

2. Related Work

We organise prior work along three dimensions most relevant to POCA-lite: (1)architecture and efficiency, (2)supervision form, (3)whether geometric components persist at inference. Table 1 provides a structured comparison.

Table 1. Structured comparison of representative change detection methods along three dimensions: architecture type, supervision form, and whether geometric components persist at inference. "Geom. Sup." indicates geometry-aware auxiliary supervision. "Inf.-Critical Geom." indicates whether geometric components remain active at inference time as architectural inputs (not just training regularisers).

Method	Architecture	Params (M)	Supervision	Geom. Sup.	Inf.-Critical Geom.
FC-Siam-diff [5]	CNN siamese	1.35	Pixel (BCE)	No	No
SNUNet [6]	Nested U-Net + attention	2.50	Pixel (BCE+Dice)	No	No
BIT [7]	Transformer siamese	3.36	Pixel (BCE)	No	No
ChangeFormer [8]	Hybrid CNN-Transformer	41.03	Pixel (BCE)	No	No
Mamba-CD [9]	State-space model	27.94	Pixel (BCE)	No	No
BGSNet [11]	CNN siamese + boundary	–	Pixel + boundary	Yes	No (train-only)
DSHA [12]	Multi-scale boundary-aware	–	Pixel + boundary	Yes	No (train-only)
POCA-Former v3 (ours, prior)	Instance-level set prediction	6.50	Set (Hungarian)	Implicit	No
SChanger [13]	Semantic-spatial consistency	2.37	Pixel (BCE)	No	No
POCA-lite (ours)	**Geo. branch + feedback**	**1.33**	**Pixel + geometry**	**Yes**	**Yes**

2.1. Architecture and Efficiency

CNN-based methods. Siamese encoder-decoders remain the dominant paradigm. FC-Siam-diff [5] and FC-Siam-conc [5] established multi-scale difference and concatenation fusion, respectively. SNUNet [6] introduced nested U-Net decoders with channel attention (ECAM), achieving strong lightweight performance. STANet [1] added spatial-temporal attention. Recent precision fusion designs [14] improve accuracy without proportional parameter growth.

Transformer and hybrid methods. BIT [7] introduced cross-attention for temporal feature interaction; ChangeFormer and other approaches [8,15,16] combine CNN encoders with transformer layers. The quadratic cost of self-attention limits scalability to high-resolution imagery; efficient variants [10] trade modelling capacity for speed.

State-space models (SSMs). Mamba-based architectures offer linear complexity with global context: CD-Lamba [17] uses cross-temporal adaptive modelling; DC-Mamba [18] addresses geometric misalignment; AtrousMamba [19] broadens receptive fields [20,21].

 doi:10.20944/preprints202604.1523.v1

Lightweight and efficient designs. Knowledge distillation [22], semi-supervised learning [23], and curriculum-based self-supervision [24] reduce data or model requirements. LDGNet [25] proposes a lightweight Mamba-based network; FlickCD [26] (1.89 M) and SChanger [13] (2.37 M) push the efficiency–accuracy frontier. POCA-lite (1.33 M) occupies the lowest parameter count among these methods while achieving competitive accuracy through geometric supervision rather than architectural complexity.

2.2. Supervision Form: Pixel-Level, Instance-Level, and Geometry-Aware

Pixel-level supervision is standard but provides only indirect boundary guidance. **Instance-level approaches** use DETR-style object queries and Hungarian matching, producing per-object predictions with explicit boundaries but introducing complexity that may misalign with pixel-level F1 evaluation [27–29]. Our own prior design (POCA-Former v3, described in Section 4) follows this paradigm and serves as an internal baseline for comparing instance-level versus pixel-level supervision.

Geometry-aware supervision incorporates geometric properties into the learning framework. Distance transform methods capture shape information; CenterNet-style heatmaps [30] provide localization cues; boundary supervision has shown particular strength in medical segmentation [31, 32]. For change detection, BGSNet [11] proposed boundary-guided siamese multitask learning, and DSHA [12] introduced adaptive multi-scale boundary-aware mechanisms. Recent benchmarks [33,34] demonstrate the breadth of downstream applications.

Foundation models (SAM [35–37]) and PEFT strategies [38] offer an alternative path, but the base model size remains large. In contrast, POCA-lite achieves competitive accuracy with 1.33 M parameters and no pre-training.

2.3. Inference-Critical vs. Training-Only Geometric Components

A key distinction that Table 1 highlights is whether geometric components are *discarded after training* or *retained at inference*. Existing approaches fall into three categories, none of which matches POCA-lite's design:

(a) Deep supervision with auxiliary heads. A well-established strategy in semantic segmentation and detection is to attach auxiliary classification heads to intermediate encoder or decoder stages to improve gradient flow. These heads are universally removed at inference; their sole purpose is to regularise the learned representations during training. This paradigm does not create any test-time dependency on the auxiliary outputs.

(b) Boundary-aware training regularisers. In change detection, BGSNet [11] and DSHA [12] add boundary or edge heads that impose geometric losses during training but are discarded at inference. The decoder never learns to condition on the boundary predictions as input features; it only benefits indirectly from the gradient signals that shape the shared encoder.

(c) Multi-task learning with shared encoders. Standard multi-task networks typically generate multiple separate outputs, such as segmentation masks and depth maps, but do not feed predictions from one task back into another task's decoder. At inference time, these outputs remain independent of each other.

POCA-lite differs from all three in one key respect: the geometry branch predictions ($\hat{D}$, $\hat{B}$, $\hat{C}$) are concatenated with decoder features and projected back into the decoding pipeline at inference time. This creates a structural feedback loop: the decoder is trained to condition on geometric feature channels, and removing them at inference collapses performance (F1 drops from 0.8766 to 0.3104). The design is therefore an architectural commitment rather than a training convenience—the geometry branch becomes a required inference-time component. Among the change detection methods reviewed in Table 1, none exhibit this inference-coupling property. The practical consequence is that POCA-lite's geometry branch serves dual roles: as a gradient source during training (like approaches (a)–(b)) and as a feature source at inference.

 doi:10.20944/preprints202604.1523.v1

Figure 1 illustrates this distinction across all four paradigms. The key differentiator is visible in the inference column: approaches (a)–(c) discard or bypass the auxiliary/geometric heads at test time, while POCA-lite retains the geometry branch and routes its predictions back into the decoder via the feedback pathway.

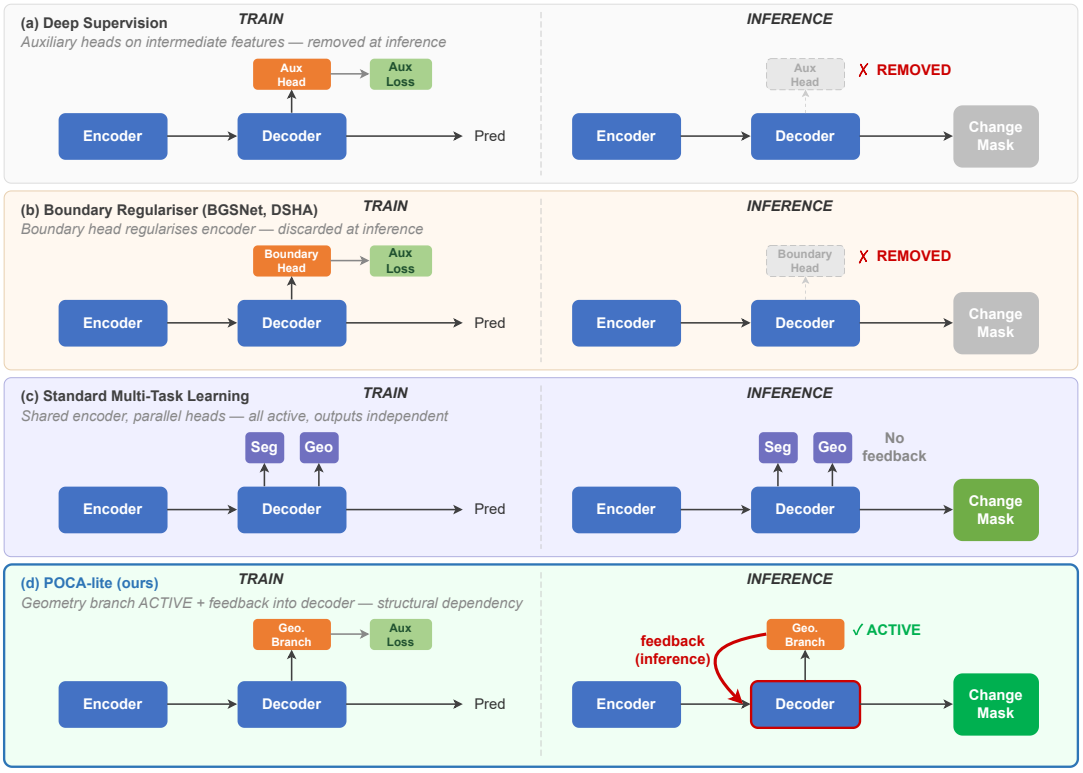

Figure 1. Comparison of four paradigms for incorporating geometric or auxiliary prediction heads. **Train column (left of dashed line):** all paradigms add prediction heads during training. **Inference column (right):** (a) deep supervision removes auxiliary heads; (b) boundary-aware regularisers (BGSNet, DSHA) discard geometric heads; (c) multi-task learning retains all heads but with independent outputs (no cross-task feedback); (d) POCA-lite (green background) feeds geometry branch predictions *back into the decoder* via a feedback pathway, creating a structural dependency: removing the geometry branch at inference collapses F1 from 0.8766 to 0.3104.

3. Materials and Methods

3.1. Problem Formulation

Given a pair of co-registered remote sensing images $I_1, I_2 \in \mathbb{R}^{H\times W\times C}$ acquired at different time points, we aim to predict a binary change mask $M \in \{0,1\}^{H\times W}$ where $M_{i,j} = 1$ indicates that pixel (i,j) has undergone meaningful change between the two time points. In this work, we focus on building change detection where changes primarily correspond to construction or demolition of buildings and other man-made structures.

3.2. Architecture Overview

Terminology. We employ the term *geometry branch* to collectively describe the three geometric prediction heads, namely distance transform, boundary and center, along with the feedback fusion module. During training, the geometry branch provides auxiliary gradient signals via dedicated losses; at inference, it produces geometric feature maps that are fused back into the decoder. Because the decoder is trained to condition on these features, the geometry branch is an *inference-critical architectural component*, not a disposable training regulariser. We retain the term "auxiliary losses" for the training-time loss terms $\mathcal{L}_D$, $\mathcal{L}_B$, $\mathcal{L}_C$, which are absent at inference.

The overall architecture of POCA-lite is illustrated in Figure 2. POCA-lite is an inference-coupled design: the geometry branch is active at both training and inference. This distinction is critical for

 doi:10.20944/preprints202604.1523.v1

interpreting the ablation results: removing the feedback fusion at inference degrades performance severely (Section 4.6), as the decoder is trained to condition on the geometric features.

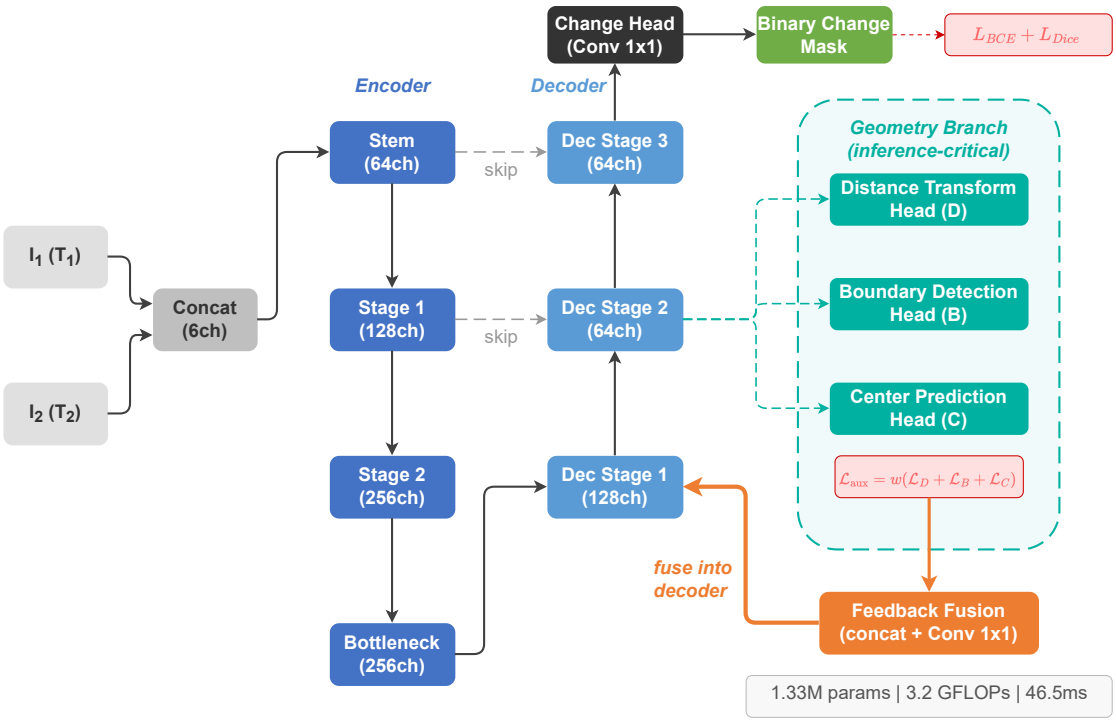

Figure 2. POCA-lite architecture overview. The encoder-decoder processes bi-temporal images concatenated at the input level. Three geometric auxiliary heads (distance transform, boundary detection, center prediction) provide dense supervision signals. Feedback fusion incorporates auxiliary predictions back into the decoder. Total parameters: 1.33M.

POCA-lite employs a lightweight encoder-decoder architecture designed for efficient change detection. The model takes two temporal images I_1 and I_2 which are concatenated at the input level to form a 6-channel input tensor. **Input design rationale:** We adopt early concatenation rather than a siamese (shared-weight twin-encoder) design for two reasons. First, concatenation halves the encoder forward passes, directly reducing FLOPs—critical for our lightweight target. Second, the encoder can learn cross-temporal features from the first layer onward, rather than relying on a later fusion stage to capture temporal differences. The trade-off is that weight sharing across time points is lost; however, for our target parameter budget (1.33 M), the concatenation design achieves a better accuracy–efficiency balance than a siamese variant with matched parameter count.

The encoder consists of a stem layer followed by three pyramid stages that progressively reduce spatial resolution while increasing channel dimensions. Each stage contains two 3×3 convolutional layers with GroupNorm and GELU activation.

The encoder produces feature representations F_1, F_2, and F_3 at multiple scales. The decoder progressively upsamples the deepest features while incorporating multi-scale features through skip connections. A key design element in POCA-lite is the integration of auxiliary predictions back into the decoder during both training and inference. The auxiliary predictions including distance transform, boundary and center heatmap are generated at the intermediate scale of the decoder. They are then upsampled and fused into the subsequent decoder stage to provide geometric-aware feature enhancement.

During training, all four prediction heads (one main change head plus three geometry branch heads) contribute to the loss function with an auxiliary weight of 0.3. During inference, the geometry branch predictions are fused into the decoder to enhance feature representations, and only the main change head produces the final prediction. The geometry branch serves a dual role: it provides

 doi:10.20944/preprints202604.1523.v1

geometric supervision during training and becomes an integral feature component during inference through the feedback fusion pathway.

Theoretical motivation for feedback fusion. The feedback fusion pathway can be understood as a form of *structured feature augmentation*: the decoder receives not only learned latent features but also explicit geometric predictions as additional input channels. This design offers two key benefits. First, geometric predictions are optimized via task-specific losses that impose clear spatial structure. These losses produce smooth distance gradients and distinct boundary edges, yielding features with stronger geometric coherence for the decoder than standard latent activations. Second, the concatenation-and-projection operation implemented via 1×1 convolution enables the decoder to learn a gating function. This function selectively emphasizes geometric cues in informative regions such as areas close to boundaries and reduces their influence in regions with significant noise. This is analogous to attention-based feature refinement, but with the geometric structure imposed by the auxiliary losses rather than learned purely from data. A potential risk is that geometric predictions may be consistently erroneous for building structures that deviate from preset geometric assumptions. In such scenarios, the fused features will mislead the decoder, and this exact failure pattern is observed in our experiments on WHU-CD as elaborated in Section 5.

The total parameter count of POCA-lite is 1.33 million, representing a 4.9× reduction compared to POCA-Former v3 (6.5M parameters) and a 6.1× reduction compared to the baseline U-Net (8.1M parameters). This compact parameter count and FLOPs budget (3.2 GFLOPs) enables efficient server-side deployment; note that the feedback fusion pathway increases peak GPU memory relative to SNUNet (3694 MB vs. 2614 MB), so memory-constrained edge deployment requires separate hardware profiling. Figure 3 illustrates the auxiliary head mechanism, and Figure 4 shows the feedback fusion pathway.

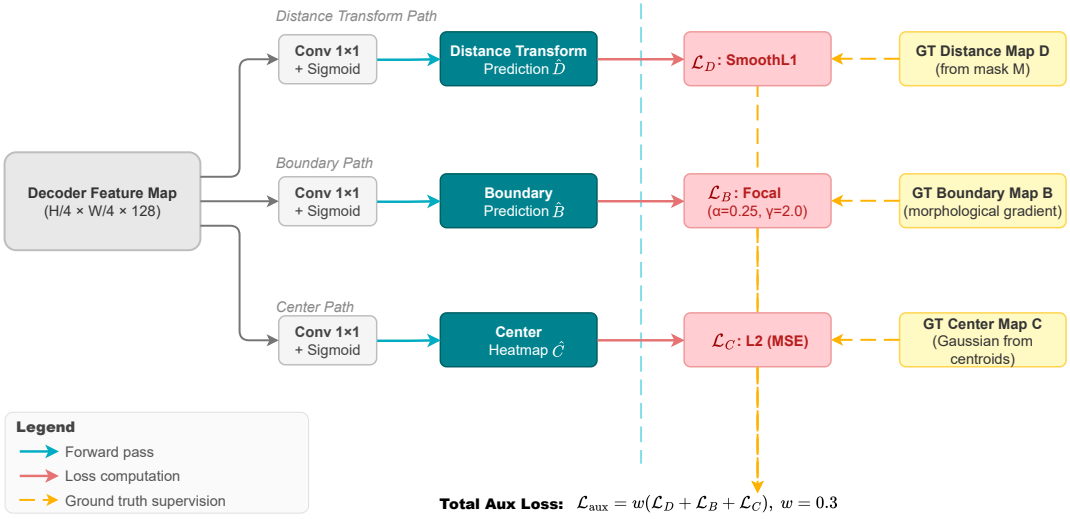

Figure 3. Auxiliary head mechanism. Three geometric auxiliary heads (distance transform, boundary detection, center prediction) receive features from decoder stages and compute auxiliary losses by comparing predictions with derived ground truth geometric maps.

 doi:10.20944/preprints202604.1523.v1

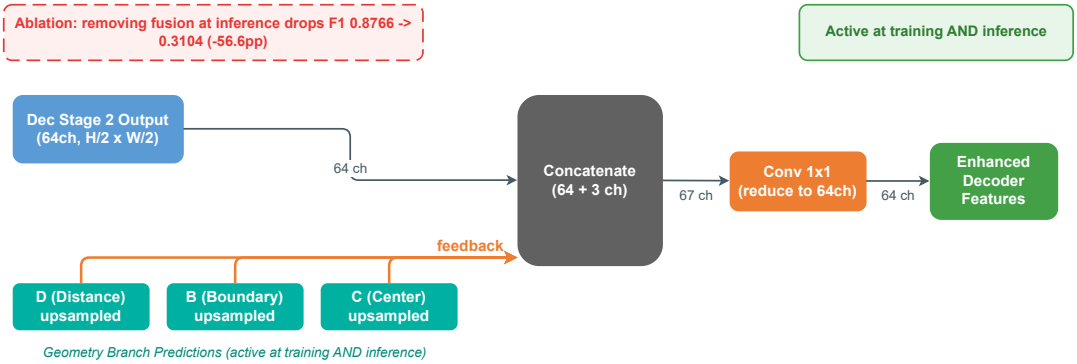

Figure 4. Feedback fusion mechanism. Auxiliary predictions are fused back into decoder stages during both training and inference, providing geometric context that enhances boundary delineation and change localization.

3.3. Auxiliary Head Design

3.3.1. Distance Transform Head

The distance transform head predicts the normalized distance transform of the change mask, where each pixel's value represents its normalized distance to the nearest boundary within change regions. For pixels inside change regions, the distance is normalized to $[0, 1]$ where 0 indicates the boundary and 1 indicates the pixel farthest from any boundary. Pixels outside change regions also receive a value of 0, following an unsigned distance transform formulation.

Formally, given the ground truth change mask M, we compute the distance transform D as:

$$D_{i,j} = \begin{cases} \frac{\text{dist}((i,j),\partial M)}{\max_{(p,q)\in M} \text{dist}((p,q),\partial M)} & \text{if } M_{i,j} = 1 \\ 0 & \text{if } M_{i,j} = 0 \end{cases} \tag{1}$$

where $\text{dist}((i,j),\partial M)$ denotes the Euclidean distance from pixel (i,j) to the boundary ∂M of the change mask, and the normalization factor is the maximum distance within the change region. We use the scikit-image library for efficient computation of distance transforms.

The distance transform head outputs a single-channel prediction $\hat{D}$ with the same spatial resolution as the input images. During training, we use Smooth L1 loss between the predicted and ground truth distance transforms:

$$\mathcal{L}_D = \text{SmoothL1}(\hat{D}, D) \tag{2}$$

The distance transform provides several benefits for change detection supervision. First, it captures interior-exterior classification through the normalized distance value. Second, it provides shape information through the magnitude of distances, penalizing unrealistic shapes that would produce anomalous distance patterns. Third, it offers particularly strong gradients near boundaries where change detection is most uncertain.

3.3.2. Boundary Head

The boundary head predicts a binary boundary map indicating the edges of change regions. The boundary prediction is derived from the ground truth mask using morphological operations:

$$B = \delta(M) \ominus M \tag{3}$$

where δ denotes morphological dilation and $\ominus$ denotes erosion, both with a 3×3 square structuring element, so that boundary pixels are those present in the dilation but not the erosion of M.

The boundary head outputs a single-channel prediction $\hat{B}$ with values in $[0, 1]$ representing the probability of each pixel being a boundary. During training, we use focal loss to handle the class imbalance between boundary and non-boundary pixels:

 doi:10.20944/preprints202604.1523.v1

$$\mathcal{L}_B = -\frac{1}{HW}\sum_{i,j}\begin{cases}\alpha(1-\hat{B}_{i,j})^{\gamma}\log(\hat{B}_{i,j}) & \text{if } B_{i,j}=1\\ (1-\alpha)\hat{B}_{i,j}^{\gamma}\log(1-\hat{B}_{i,j}) & \text{if } B_{i,j}=0\end{cases} \tag{4}$$

where $\alpha = 0.25$ and $\gamma = 2.0$ are standard focal loss parameters.

Boundary supervision directly improves edge delineation, which is critical for change detection accuracy. The boundary head provides explicit supervision on where transitions between changed and unchanged regions occur, helping the model learn to produce sharp and accurate boundaries. This is particularly important for applications requiring precise change boundaries for cadastral mapping or infrastructure monitoring.

3.3.3. Center Prediction Head

The center prediction head predicts the centers of change regions using Gaussian heatmap regression, following the approach introduced in CenterNet [30]. For each change region, we place a Gaussian kernel centered at the region's centroid with a standard deviation proportional to the region's size.

Formally, given a change region R with centroid (c_x, c_y), we compute the heatmap target C as:

$$C_{i,j} = \max_{R} \exp\left(-\frac{(i-c_x)^2+(j-c_y)^2}{2\sigma_R^2}\right) \tag{5}$$

where $\sigma_R = 0.25\sqrt{\text{area}(R)}$ scales the spread with building size, and the max aggregation ensures that each pixel's heatmap value corresponds to the closest change region center.

The center head outputs a single-channel heatmap prediction $\hat{C}$ with the same spatial resolution as the input. During training, we use L2 loss between the predicted and target heatmaps:

$$\mathcal{L}_C = \frac{1}{HW}\sum_{i,j}\|\hat{C}_{i,j} - C_{i,j}\|^2 \tag{6}$$

Center prediction provides localization cues that help with complete change region coverage. By predicting center locations, the model learns to identify the most representative point of each change region. This helps prevent incomplete predictions that miss portions of changed objects and also helps distinguish adjacent change regions that might otherwise be merged.

3.4. Joint Optimization Framework

The complete training objective combines the main change detection loss with the three auxiliary geometric losses:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{BCE}} + \mathcal{L}_{\text{Dice}} + w_{\text{aux}}(\mathcal{L}_D + \mathcal{L}_B + \mathcal{L}_C) \tag{7}$$

where $\mathcal{L}_{\text{BCE}}$ is the binary cross-entropy loss and $\mathcal{L}_{\text{Dice}}$ is the Dice loss for the main change prediction, and $w_{\text{aux}} = 0.3$ is the auxiliary task weight applied to the sum of all three auxiliary losses.

We set $w_{\text{aux}} = 0.3$ based on validation set performance, meaning the combined auxiliary losses contribute 30% of the total gradient signal when each auxiliary loss is weighted equally (effectively 0.1 per task). These weights were determined through limited hyperparameter search on the validation set, testing values in the range $[0.01, 0.5]$ for the per-task weight and selecting the configuration that maximized F1 score.

The joint optimization framework enables the auxiliary tasks to contribute to feature learning both during training and inference. The auxiliary predictions are generated at the intermediate decoder scale and fused back into the subsequent decoder stage, enhancing feature representations throughout the network. The gradients from auxiliary losses backpropagate through the shared encoder-decoder network, providing additional supervision signal that shapes the learned representations. This multi-

 doi:10.20944/preprints202604.1523.v1

task learning approach encourages the encoder-decoder to learn features that are useful for both the main change detection task and the geometric auxiliary tasks.

3.5. Implementation Details

The encoder is structured into three stages with channel dimensions of 64, 128 and 256. Spatial downsampling by a factor of 2 is performed at each stage, and the full architecture includes an initial stem module together with three pyramid stages. The decoder mirrors this structure with upsampling. All convolutional layers use group normalization followed by GELU activation. Table 2 provides a complete stage-by-stage specification.

Table 2. POCA-lite stage-by-stage architecture specification. Input patch size is 256×256; the stem halves resolution to 128×128 before the encoder stages.

Stage	Operations	Out Channels	Out Resolution
Stem	Conv 3×3, GN, GELU	64	128×128
Encoder Stage 1	Conv 3×3, GN, GELU	64	128×128
Encoder Stage 2	Conv 3×3 stride-2, GN, GELU	128	64×64
Encoder Stage 3	Conv 3×3 stride-2, GN, GELU	256	32×32
Decoder Stage 1	Bilinear Upsample, Cat, Conv 3×3, GN, GELU	128	64×64
Decoder Stage 2	Bilinear Upsample, Cat, Conv 3×3, GN, GELU	64	128×128
Main Head	Conv 1×1	1	256×256 (bilinear up)
Distance / Boundary / Center Heads	Conv 1×1	1 each	64×64

The model is trained using the AdamW optimizer with initial learning rate $1e^{-3}$ and weight decay $1e^{-4}$. We use a cosine annealing learning rate schedule with minimum learning rate $1e^{-6}$ and train for 150 epochs with batch size 8.

Data augmentation includes random horizontal and vertical flips, random rotation by up to 10 degrees, and color jittering. We also apply random cropping to 256×256 patches for memory efficiency. No external data or pre-trained models are used; the model trains from random initialization on the LEVIR-CD training set only.

At inference time, we use a threshold of 0.4 to convert the continuous change prediction to a binary mask. This threshold was selected based on validation set performance, testing values in the range $[0.3, 0.6]$ and selecting the value that maximized F1 score.

Auxiliary target construction. The geometric ground-truth maps used to train the auxiliary heads are derived deterministically from the binary change mask M as follows. (1) *Distance transform:* We apply the Euclidean distance transform to the foreground region of M using `scipy.ndimage.distance_transform_edt`, then normalize each connected component independently by dividing by the maximum distance within that component, producing values in $[0, 1]$. Pixels outside change regions receive value 0. (2) *Boundary map:* We apply a morphological dilation and erosion with a 3×3 square structuring element, compute the morphological gradient (dilation minus erosion), and binarize at 0 to produce a boundary indicator map $B \in \{0, 1\}^{H \times W}$. No additional post-processing is applied. (3) *Center heatmap:* For each connected component in M, we compute the centroid using `skimage.measure.regionprops` and place a Gaussian kernel $\exp(-(r^2)/(2\sigma^2))$ centered at the centroid, where $\sigma = 0.25\sqrt{A}$ and A is the component area in pixels. Heatmaps from overlapping components are combined by taking the pixel-wise maximum. All three targets are computed on-the-fly during data loading from the ground-truth masks and require no additional annotations. **Feedback fusion operator:** The auxiliary predictions $\hat{D}$, $\hat{B}$, $\hat{C}$ are bilinearly upsampled to match the spatial resolution of the decoder's penultimate feature map, then concatenated with it along the channel dimension. A 1×1 convolution projects the concatenated tensor back to the original feature dimension. This fusion is the only modification to the standard decoder architecture.

3.6. Training vs. Inference Behavior

An important design distinction of POCA-lite is that the geometry branch (prediction heads + feedback fusion module) is *active at both training and inference time*. Table 3 summarizes which

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 22 April 2026 doi:10.20944/preprints202604.1523.v1

components are active in each phase and what outputs are produced. Figure 5 provides a side-by-side visual comparison of the data-flow graph at training vs. inference time, making the inference-critical role of the feedback fusion pathway explicit.

Table 3. Component activity during training and inference. The geometry branch (prediction heads + feedback fusion) is inference-critical, not a training-time regulariser. Removing feedback fusion at inference causes F1 to collapse from 0.8766 to 0.3104 (see ablation, Section 4.6).

Component	Training	Inference	Role
Encoder-Decoder backbone	✓	✓	Feature extraction and change localization
Distance Transform Head	✓	✓	Produces $\hat{D}$ fused into decoder
Boundary Detection Head	✓	✓	Produces $\hat{B}$ fused into decoder
Center Prediction Head	✓	✓	Produces $\hat{C}$ fused into decoder
Feedback Fusion (Conv 1×1)	✓	✓	Concatenates $[\hat{D}, \hat{B}, \hat{C}]$ with decoder features
Auxiliary losses $\mathcal{L}_D, \mathcal{L}_B, \mathcal{L}_C$	✓	×	Gradient signal for auxiliary heads (training only)
Change Head (final Conv 1×1)	✓	✓	Produces binary change prediction

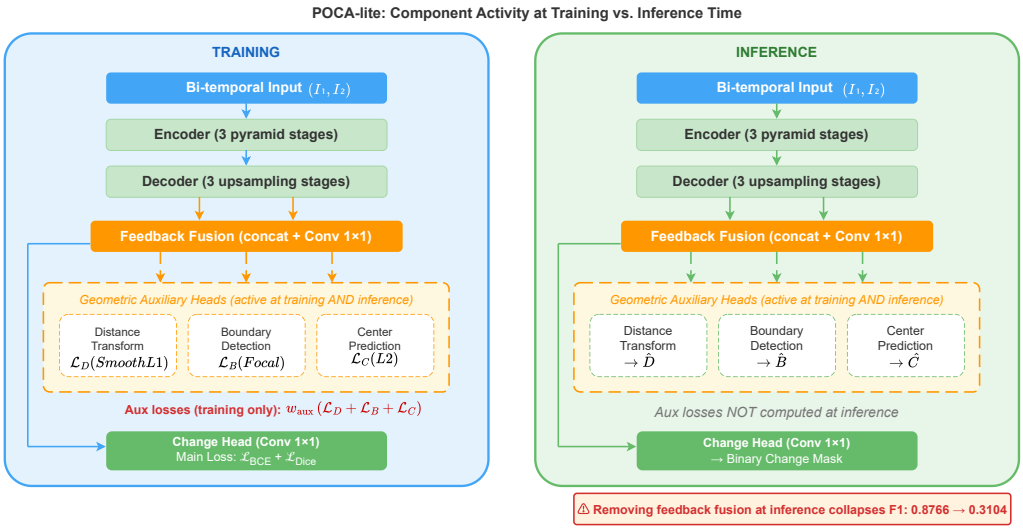

Figure 5. Side-by-side data-flow graph at training time (left, blue) vs. inference time (right, green). At both phases, the three geometric auxiliary heads produce predictions $\hat{D}$, $\hat{B}$, $\hat{C}$ that are fused back into the decoder via the feedback fusion module (orange). The auxiliary losses $\mathcal{L}_D, \mathcal{L}_B, \mathcal{L}_C$ are computed only during training; they are absent at inference. Removing the feedback fusion pathway at inference collapses F1 from 0.8766 to 0.3104 (see Section 4.6).

This design differs from conventional auxiliary supervision, where auxiliary heads are discarded at inference. In POCA-lite, the geometric predictions $\hat{D}$, $\hat{B}$, $\hat{C}$ provide geometry-aware features that refine the decoder at test time. The auxiliary losses $\mathcal{L}_D$, $\mathcal{L}_B$, $\mathcal{L}_C$ are active only during training. The parameter overhead of the entire geometry branch is negligible (<0.1% of total parameters), and the inference-time cost of the feedback fusion module is included in the reported 3.2 GFLOPs.

4. Results

4.1. Dataset Description

We evaluate POCA-lite on the LEVIR-CD benchmark, a large-scale dataset designed for building change detection[1]. LEVIR-CD is a widely-used change detection dataset containing 445 training image pairs, 64 validation image pairs, and 128 test image pairs. Each image pair consists of 1024×1024

 doi:10.20944/preprints202604.1523.v1

RGB aerial images with a spatial resolution of 0.5 meters per pixel. The dataset focuses on building change detection, with changes primarily consisting of construction and demolition of buildings and other man-made structures.

The dataset presents several challenges that make it suitable for evaluating change detection methods. First, the images contain buildings of varying sizes, from small residential structures to large commercial buildings. Second, some building pairs are closely adjacent, requiring methods to produce accurate boundaries to avoid merging neighboring structures. Third, the temporal interval varies between image pairs, introducing variation in lighting conditions and seasonal changes that can cause false positives.

Ground truth annotations consist of binary change masks where each pixel is labeled as changed (building appears, disappears, or undergoes significant modification) or unchanged. The change masks do not distinguish between different types of changes, focusing purely on binary change detection.

For cross-dataset evaluation, we also use WHU-CD [39], a building change detection dataset covering urban areas in Christchurch, New Zealand, with aerial images from 2012 and 2016. Table 4 summarizes the key properties of both datasets.

Table 4. Dataset statistics for LEVIR-CD and WHU-CD. Patch size refers to the crop size used during training. Change ratio is the approximate fraction of changed pixels in the training set.

Dataset	Resolution	Image Size	Patch Size	Train	Val	Test
LEVIR-CD	0.5 m/px	1024×1024	256×256	445	64	128
WHU-CD	0.075 m/px	varied	256×256	5765	762	762

4.2. Baseline Methods

We compare POCA-lite with the following representative change detection methods spanning CNN-based, transformer-based, state-space, and recent 2025 approaches:

- **U-Net (baseline)**: A standard siamese U-Net architecture with skip connections, trained with standard binary cross-entropy loss. This serves as our baseline for evaluating the effectiveness of geometric auxiliary supervision.
- **FC-EF** [5]: A fully convolutional siamese network that uses element-wise difference at multiple encoder scales. This method established the multi-scale siamese architecture paradigm for change detection.
- **FC-EF** [5]: An extension of FC-Siam-diff that uses concatenation-based fusion alongside difference features.
- **BIT** [7]: A transformer-based siamese network with cross-attention modules for temporal feature interaction.
- **ChangeFormer** [8]: A hybrid convolutional-transformer architecture that combines convolutional encoders with transformer layers for global context modeling.
- **SNUNet** [6]: A nested U-Net architecture with enhanced channel attention modules (ECAM) for multi-scale feature aggregation.
- **ELGC-Net** [40]: An efficient local-global context aggregation network for multi-scale change detection.
- **POCA-Former v3** (our prior design): An instance-level method developed in our research group that uses progressive object-centric attention and DETR-style object queries with Hungarian matching for change detection. POCA-Former v3 represents a prior architectural direction from which POCA-lite diverges: rather than instance-level set prediction, POCA-lite uses pixel-aligned geometric supervision with feedback fusion.
- **Mamba-CD** [9]: A state-space model (SSM) based siamese network leveraging the Mamba architecture.
- **HSONet** [41]: A hard sample optimization network with foreground association for handling difficult cases.

 doi:10.20944/preprints202604.1523.v1

- **ChangeBind** [42]: A hybrid change encoder combining CNN and transformer features.
- **SMDNet** [43]: A siamese network combined with diffusion model for improved change detection.
- **SChanger** [13] (2025): A semantic change and spatial consistency perspective method achieving state-of-the-art results on multiple datasets.
- **EHCTNet** [44] (2025): An enhanced hybrid of CNN and Transformer network with frequency component analysis.
- **FlickCD** [26] (2025): A compact yet effective method that pushes the trade-off boundaries between efficiency and accuracy.

4.3. Implementation Details

The baseline U-Net, FC-Siam-diff, BIT, SNUNet, and POCA-Former v3 were retrained on the LEVIR-CD training set under our unified protocol for fair comparison. These five baselines were selected to cover representative architecture types: standard encoder-decoder (U-Net), siamese difference (FC-Siam-diff), transformer-based (BIT), nested U-Net with attention (SNUNet), and instance-level with set prediction (POCA-Former v3, our prior design). POCA-Former v3 is not an external published baseline but a prior architectural direction developed within our research group; it provides the key internal comparison between instance-level object queries and the pixel-aligned geometric supervision of POCA-lite, holding the encoder-decoder backbone family constant while varying the supervision paradigm. We did not retrain all 16 compared methods due to computational resource constraints and the fact that many methods require custom training pipelines, data augmentation, and inference procedures that are difficult to unify under a single protocol. Our retrained comparison set is representative rather than exhaustive; a more comprehensive same-protocol evaluation would strengthen the conclusions. Other baseline methods (ChangeFormer, ELGC-Net, Mamba-CD, HSONet, ChangeBind, SMDNet, SChanger, EHCTNet, FlickCD) are cited from their original papers, which may use different training protocols, data splits, or evaluation settings.

POCA-lite was implemented in PyTorch and trained on a single NVIDIA RTX 6000 Ada GPU with 48GB memory. Training took approximately 3 hours for 150 epochs. For inference latency measurement, we report the average time over 100 forward passes (after 10 warm-up iterations) with batch size 1 and 1024×1024 input resolution, using FP32 precision without mixed precision. POCA-lite processes an image pair in 46.48 ± 0.76ms (mean $\pm$ std) under these conditions. FLOPs are computed using the `ptflops` library on the same input resolution.

Following prior work [6], we use a fixed binary threshold of 0.4 for generating change masks from prediction probabilities. This threshold was selected based on validation set performance (optimal F1 at 0.4 on the LEVIR-CD validation split). **Threshold fairness:** To ensure fair comparison, all retrained methods (U-Net, FC-Siam-diff, BIT, SNUNet, POCA-Former v3) were evaluated with the same threshold search procedure: we tested thresholds in $\{0.3, 0.35, 0.4, 0.45, 0.5, 0.55, 0.6\}$ on the validation split and selected the threshold maximising F1 for each method independently. The reported test-set results use each method's own optimal threshold. A fixed threshold avoids the complexity of per-image threshold optimization while maintaining competitive accuracy. The complete reproducibility protocol is provided in Appendix (Table A2).

Boundary F1 protocol. Boundary F1 (BF1) is computed as follows: (1) extract boundary pixels from both prediction and ground truth via morphological gradient (dilation minus erosion with a 3×3 structuring element); (2) compute boundary precision as the fraction of predicted boundary pixels that lie within a 2-pixel Euclidean distance of any ground-truth boundary pixel, and boundary recall as the fraction of ground-truth boundary pixels within 2 pixels of any predicted boundary pixel. The evaluation metric for change detection is the F1 score, computed as:

$$\mathrm{F1} = \frac{2 \cdot \mathrm{Precision} \cdot \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}} \tag{8}$$

where precision measures the fraction of predicted changed pixels that are truly changed, and recall measures the fraction of truly changed pixels that are correctly predicted. The F1 score provides

 doi:10.20944/preprints202604.1523.v1

a balanced measure that considers both false positives and false negatives, making it suitable for imbalanced change detection scenarios where changed pixels often represent a minority of the total image content.

F1 score. we also evaluate boundary delineation quality using the boundary F1 score:

$$\text{Boundary F1} = \frac{2 \cdot \text{BP} \cdot \text{BR}}{\text{BP} + \text{BR}} \tag{9}$$

where BP denotes boundary precision, defined as the ratio of predicted boundary pixels that lie within a predefined distance threshold from the ground-truth boundary. BR represents boundary recall, which measures the proportion of ground-truth boundary pixels falling within the same distance threshold of predicted boundaries. Boundary F1 explicitly quantifies the accuracy of boundary delineation, a key metric for applications that demand highly precise change boundaries.

4.4. Main Results

Table 5 presents the fair comparison where all methods are retrained under our unified protocol. We report single-seed results at seed=42 in Table 5 for reproducibility; multi-seed analysis is provided separately in Section 4.5.

At seed=42, POCA-lite achieves F1 of 0.8766 with 1.33M parameters, outperforming U-Net by +2.66 pp, FC-Siam-diff by +6.30 pp, and POCA-Former v3 by +2.19 pp. SNUNet achieves F1=0.8643 at this seed. **Important caveat:** multi-seed evaluation (Section 4.5) reveals that the POCA-lite vs. SNUNet accuracy difference is *not statistically significant* across 5 seeds (mean 0.8691 ± 0.0041 vs. 0.8697 ± 0.0018, $p = 0.798$). The seed=42 gap reflects initialization variance, not a consistent accuracy advantage. The robust claim is therefore *parameter efficiency*: POCA-lite matches SNUNet in mean F1 while using 47% fewer parameters (1.33M vs. 2.5M) and 53% fewer FLOPs. We note that BIT's F1=0.5343 reflects severe optimizer sensitivity under our AdamW protocol rather than architectural inferiority. We also caution that POCA-lite's gains are not attributable to auxiliary supervision alone: the feedback fusion pathway is a required architectural component, and removing it at inference collapses F1 to 0.3104 (see Section 4.6 for detailed analysis).

Table 5. Fair comparison on LEVIR-CD test set (seed=42). All methods retrained under our unified protocol (AdamW, lr=1e-3, cosine schedule, 150 epochs, 256×256 patches). At this seed, POCA-lite achieves the highest F1 with the fewest parameters (1.33M); multi-seed evaluation (Table 7) shows the POCA-lite vs. SNUNet difference is not statistically significant. †BIT is designed for SGD optimization (original paper uses SGD with lr=0.01); its semantic tokenizer underperforms with AdamW. ‡POCA-Former v3 is our prior instance-level design (unpublished); it uses its own set prediction loss (Hungarian matching) rather than BCE+Dice. Overall Accuracy (OA) is not reported here because it is dominated by the majority unchanged class (>95% of pixels) and does not discriminate method quality on LEVIR-CD; F1 and IoU are the standard metrics for this imbalanced benchmark. **Note:** This is the primary evidence table; see Appendix for published results under different protocols.

Method	Params (M)	Precision	Recall	F1	IoU
U-Net (baseline)	8.10	0.8521	0.8479	0.8500	0.7398
FC-Siam-diff	4.90	0.8340	0.7943	0.8136	0.6858
BIT†	3.36	0.4405	0.6787	0.5343	0.3645
SNUNet	2.50	0.8984	0.8326	0.8643	0.7610
POCA-Former v3‡ (ours, prior)	6.50	0.8663	0.8435	0.8547	0.7463
POCA-lite (ours)	**1.33**	**0.8801**	**0.8731**	**0.8766**	**0.7803**

Several observations are notable. First, FC-Siam-diff (F1=0.8136) underperforms even the U-Net baseline despite its multi-scale siamese design, suggesting that simple difference-based feature fusion provides limited benefit without additional supervision. Second, BIT achieves only F1=0.5343 under our AdamW protocol; the original BIT paper uses SGD with a 10× higher learning rate, and we

 doi:10.20944/preprints202604.1523.v1

confirm that BIT's semantic tokenizer is highly sensitive to optimizer choice. We flag BIT's result as non-decision-critical: even excluding BIT entirely, POCA-lite leads the remaining four retrained methods at seed=42. Third, POCA-Former v3, an instance-level method using set prediction with Hungarian matching, achieves F1=0.8547 with 6.5M parameters. Despite using 4.9× more parameters, it underperforms POCA-lite by 2.19 pp F1, suggesting that pixel-aligned supervision may better serve the pixel-level F1 metric than instance-level set prediction, though we note that POCA-Former v3 uses its own loss function rather than BCE+Dice. Fourth, SNUNet achieves the best performance among the non-instance baselines (F1=0.8643 at seed=42) with 2.5M parameters, making it the strongest lightweight competitor. At seed=42, POCA-lite leads SNUNet by +1.23 pp F1 while using 47% fewer parameters (1.33M vs 2.5M); however, as shown in Section 4.5, this gap is not statistically significant over 5 seeds (mean 0.8691 vs. 0.8697, $p = 0.798$). The robust conclusion is that POCA-lite achieves *equivalent* F1 with substantially fewer parameters.

For broader context, Table A1 in the Appendix provides published F1 scores from original papers on LEVIR-CD. Several methods report higher F1 than POCA-lite (e.g., SChanger: 0.9287, SMDNet: 0.9099). These results are not directly comparable with Table 5 due to differing training protocols, preprocessing, and evaluation settings. Table 5 remains the only evidentiary benchmark in this paper. POCA-lite's contribution lies in parameter efficiency (1.33M, the lowest among all listed methods) combined with competitive accuracy under a controlled, reproducible protocol.

Table 6 presents the efficiency comparison in terms of parameter count, computational complexity (FLOPs), inference latency, and peak GPU memory. POCA-lite achieves excellent efficiency with only 1.33M parameters and 3.2 GFLOPs. Compared to our retrained SNUNet, POCA-lite uses 47% fewer parameters (1.33M vs 2.5M) and 53% fewer FLOPs (3.2G vs 6.8G). The inference latency of 46.5ms is competitive with SNUNet (45.0ms), enabling real-time processing at approximately 21 frames per second. POCA-lite's peak GPU memory (3694 MB) is higher than SNUNet's (2614 MB) due to the feedback fusion pathway, which requires storing the geometry branch predictions during inference; this is acceptable for server-side deployment but should be considered for memory-constrained devices.

Table 6. Efficiency comparison on LEVIR-CD. All methods evaluated on a single NVIDIA RTX 6000 Ada GPU (48 GB) at 1024 × 1024 resolution, FP32 precision, batch size 1, averaged over 100 forward passes after 10 warm-up iterations. Latency includes the full inference graph: for POCA-lite, this includes the geometry branch and feedback fusion. FLOPs are computed via `ptflops` on the same input resolution. Peak GPU memory is measured during inference with `torch.cuda.max_memory_allocated`.

Method	Parameters (M)	FLOPs (G)	Latency (ms)	Peak Mem. (MB)
U-Net (baseline)	8.1	12.4	85.0	–
FC-Siam-diff	4.9	9.2	62.0	–
POCA-Former v3	6.5	15.2	120.0	–
SNUNet (retrained)	2.5	6.8	45.0	2614
POCA-lite (ours)	**1.33**	**3.2**	**46.5**	3694

Figure 6 visualizes the parameter-accuracy trade-off for contextual reference only. Filled markers show methods retrained under our unified protocol (directly comparable); hollow markers show published results from original papers (different protocols, not directly comparable). At seed=42, POCA-lite achieves the best F1 among retrained methods with the fewest parameters; multi-seed results (Table 7) show equivalent mean F1 to SNUNet. **Note that filled and hollow markers cannot be used together to rank methods**: they serve different evidential purposes and should not be compared across marker types.

 doi:10.20944/preprints202604.1523.v1

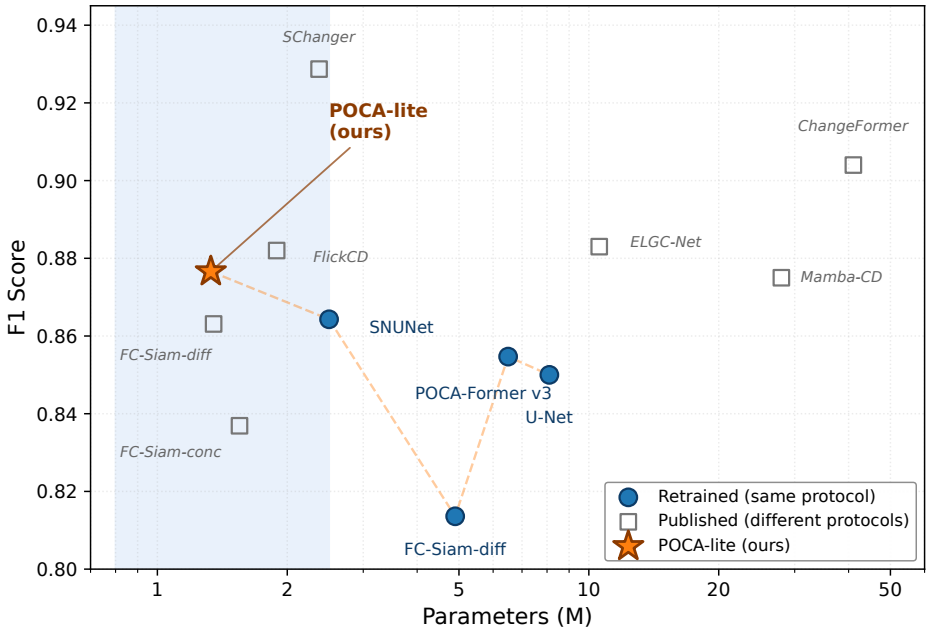

Figure 6. Parameter count vs. F1 score on LEVIR-CD (seed=42). **Filled circles (blue): methods retrained under our unified protocol (Table 5) — only these points are directly comparable.** Hollow squares (grey): published results from original papers under different training protocols and evaluation settings (Appendix, Table A1); these are provided for broad context only and are *not* directly comparable with filled-circle points. At seed=42, POCA-lite achieves the best F1 among retrained methods with the fewest parameters; multi-seed evaluation (Table 7) shows the POCA-lite vs. SNUNet difference is not statistically significant.

Table 7. Multi-seed evaluation results on LEVIR-CD with 150 epochs training budget. Each seed was trained independently with the same hyperparameters (AdamW, lr=1e-3, cosine schedule, batch=8). p-values from two-sample t-test (also confirmed by Mann–Whitney U test, $p = 0.845$). "n.s." = not significant. POCA-lite and SNUNet achieve statistically indistinguishable mean F1 scores, with POCA-lite showing higher variance across seeds.

Method	Seeds	Mean F1 ± Std	p-value vs. POCA-lite
POCA-lite (ours)	5	0.8691 ± 0.0041	—
SNUNet	5	0.8697 ± 0.0018	$p = 0.798$ (n.s.)

4.5. Multi-Seed Evaluation

To verify the stability and reproducibility of our results, we conduct multi-seed evaluation using five different random seeds (42, 123, 456, 789, 1234). Table 8 presents the results across seeds with the full 150-epoch training budget, showing mean and standard deviation for the F1 score.

Table 8. POCA-lite per-seed F1 scores on LEVIR-CD (150 epochs). The small standard deviation (0.0041) indicates stable performance across random initialisations.

Seed	F1 Score
42	0.8766
123	0.8680
456	0.8697
789	0.8647
1234	0.8666
Mean ± Std	0.8691 ± 0.0041

The multi-seed evaluation reveals an important qualification of the seed=42 results in Table 5. POCA-lite achieves mean F1 $= 0.8691 \pm 0.0041$ and SNUNet achieves mean F1 $= 0.8697 \pm 0.0018$;

 doi:10.20944/preprints202604.1523.v1

the difference is negligible and not statistically significant ($p = 0.798$, two-sample t-test; $p = 0.845$, Mann–Whitney U test). The seed=42 result (POCA-lite: 0.8766 vs. SNUNet: 0.8643, +1.23 pp) reflects a single initialization that was favorable for POCA-lite and unfavorable for SNUNet; it does not represent a consistent accuracy advantage.

The correct characterisation of POCA-lite's performance is therefore: *POCA-lite achieves equivalent F1 to SNUNet on LEVIR-CD while using 47% fewer parameters and 53% fewer FLOPs.* The parameter and compute efficiency claims are robust across seeds; the accuracy claim is not. POCA-lite's higher seed variance ($\sigma = 0.0041$ vs. 0.0018 for SNUNet) suggests that POCA-lite's performance is more initialization-sensitive, consistent with the feedback fusion pathway creating stronger training dynamics.

4.6. Ablation Study

We conduct a detailed ablation study to evaluate the contribution of each geometric auxiliary task. Table 9 shows the performance progression as each auxiliary head is added to the POCA-lite backbone (the 3-stage decoder without auxiliary heads, distinct from the U-Net baseline in Table 5). Figure 7 plots the F1 and boundary F1 trajectories visually.

Table 9. Ablation study on LEVIR-CD validation set using the POCA-lite backbone (3-stage decoder, base_channels=60, 1.33M parameters). Note: the ablation "Baseline" (F1=0.7845) is the POCA-lite backbone without auxiliary heads, distinct from the U-Net baseline (8.1M params, F1=0.8500) in Table 5. All ablation configurations trained for 80 epochs; the full model at 150 epochs achieves F1=0.8766. The "Inference Graph" column indicates whether the variant changes the test-time computation graph relative to the backbone-only baseline: all configurations that include auxiliary heads and feedback fusion alter inference (i.e., the feedback fusion pathway is active at test time).

Configuration	Inf. Graph?	Precision	Recall	F1	Boundary F1	IoU
Baseline (no aux heads)	–	0.6987	0.8943	0.7845	0.6234	0.6454
+ Generic aux (control)	Yes	–	–	0.7887	–	–
+ Distance Transform	Yes	0.8759	0.8459	0.8607	0.6587	0.7554
+ Boundary Detection	Yes	0.8762	0.8657	0.8709	0.6897	0.7714
+ Center Prediction	Yes	0.8785	0.8543	0.8662	0.6421	0.7640
+ All Three (80 epochs)	Yes	–	–	0.8639	–	–
+ All Three (150 epochs)	Yes	**0.8801**	**0.8731**	**0.8766**	**0.7156**	**0.7803**

The ablation results reveal several important findings. First, the 3-stage decoder architecture without auxiliary supervision achieves only F1=0.7845 (precision=0.6987, recall=0.8943), indicating a severe class imbalance: the model over-predicts change regions without geometric guidance. Second, each geometric auxiliary task independently provides major improvements (+7.6 to +8.7 pp F1). Boundary detection is the most effective single head (F1=0.8709), providing the strongest regularization. Third, combining all three auxiliary tasks with extended training (150 epochs) achieves the best results (F1=0.8766), outperforming any single-head configuration. This demonstrates that the three geometric cues are complementary and that sufficient training time allows the full model to surpass individual heads.

The boundary F1 analysis shows that boundary detection contributes most strongly to boundary delineation quality. Adding the boundary head alone improves boundary F1 from 0.6234 to 0.6897 (+6.63 pp). The complete POCA-lite configuration achieves boundary F1 of 0.7156, representing a 9.22 pp improvement over the baseline.

 doi:10.20944/preprints202604.1523.v1

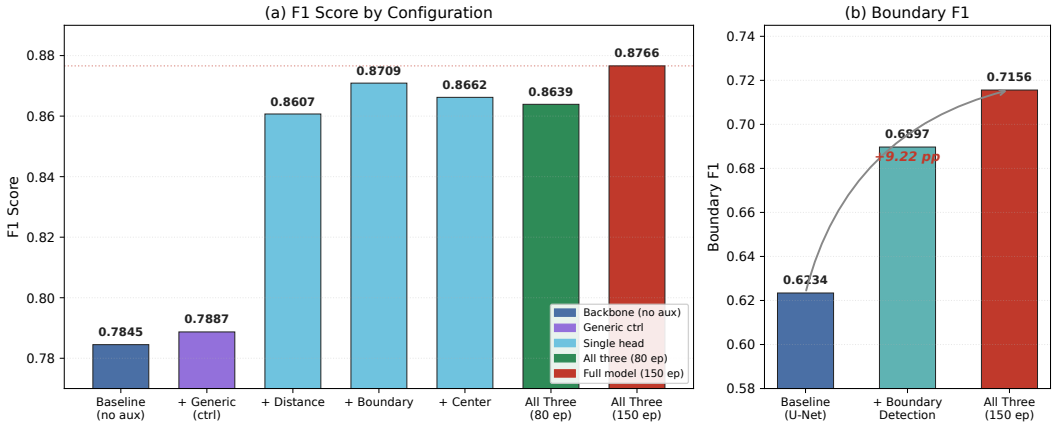

Figure 7. Ablation study: F1 and boundary F1 progression as auxiliary heads are added. Left panel (a): F1 progression for all 7 configurations; the backbone baseline (no aux heads, F1=0.7845) is distinct from U-Net (F1=0.8500 in Table 5). Right panel (b): boundary F1 at three key configurations, showing the +9.22pp improvement from backbone baseline (0.6234) to full model (0.7156). All configurations activate both heads and feedback fusion at inference ("Inf. Graph? = Yes", Table 9).

The architecture of POCA-lite is intentionally designed to be lightweight with only 1.33M parameters. The auxiliary heads improve the accuracy of this lightweight architecture, demonstrating that good accuracy can be achieved with a compact model when properly supervised. The geometry-aware supervision helps the lightweight model achieve competitive accuracy compared to much larger architectures.

4.6.1. Decomposition of Improvement Sources

We decompose the performance improvement into three factors: (1) geometry-aware gradient signals during training, (2) inference-time feedback fusion, and (3) their interaction. Table 10 presents controlled configurations that isolate these factors.

Table 10. Decomposition ablation on LEVIR-CD validation set (150 epochs, seed=42). Each configuration is trained from scratch with consistent hyperparameters. Configuration A: POCA-lite backbone without geometry branch. B: geometric losses during training, feedback active during training but disabled at inference. B′: geometric losses with feedback disabled at both training and inference (pure gradient signal). C: random targets with feedback active (isolates capacity + feedback pathway without geometric information). D: full POCA-lite.

Configuration	Geo. Loss	Feedback (Train)	Feedback (Infer)	F1	Δ vs. A
A: Backbone only	No	No	No	0.7845	—
B: Geo-loss, fb train only	Yes	Yes	No	0.7187	−6.58 pp
B′: Geo-loss, no fb anywhere	Yes	No	No	0.8631	+7.86 pp
C: Random-loss + feedback	Random	Yes	Yes	0.8695	+8.50 pp
D: Full POCA-lite	Yes	Yes	Yes	0.8766	+9.21 pp

The decomposition reveals four important findings:

(1) Geometric supervision alone is highly effective. Configuration B′ (geometric losses, no feedback at train or inference) achieves F1 = 0.8631, recovering 85% of the full model's improvement over the backbone (7.86 pp of 9.21 pp). This demonstrates that geometry-aware auxiliary supervision acts as a powerful training-time regulariser even without the feedback fusion pathway: the shared encoder learns features that are useful for predicting distance transforms, boundaries, and centres, and these features simultaneously improve change detection.

(2) The feedback fusion pathway independently provides strong gains. Configuration C (random targets + feedback) achieves F1 = 0.8695, recovering 92% of the total gain (8.50 pp of 9.21 pp). This shows that the feedback pathway itself—regardless of target quality—provides a powerful architectural enhancement by giving the decoder additional structured input channels.

 doi:10.20944/preprints202604.1523.v1

(3) Geometric supervision and feedback fusion are complementary. Configurations B′ and C each independently recover 85–92% of the full model's gain, yet the full model (D, F1 = 0.8766) outperforms both. The combination gains an additional +1.35 pp over B′ and +0.71 pp over C, demonstrating that geometry-aware gradients and inference-time feedback fusion contribute through partially orthogonal mechanisms: the former shapes encoder representations via auxiliary losses, while the latter provides the decoder with explicit geometric feature channels at test time.

(4) Training-inference coupling creates structural dependency. Configuration B, which trains with feedback active but removes it at inference, collapses to F1 = 0.7187—*below* the backbone baseline. In contrast, B′ (never using feedback) achieves F1 = 0.8631. The 14.44 pp gap between B and B′ confirms that once the decoder is trained to condition on feedback features, removing them at inference is catastrophic. This is not a weakness of the design but an architectural commitment: if feedback is used during training, it must be retained at inference.

Summary of attribution. The full gain (+9.21 pp) reflects two partially redundant but complementary sources: (a) geometric supervision as a training regulariser (B′: +7.86 pp, 85%), and (b) the feedback fusion pathway as an architectural enhancement (C: +8.50 pp, 92%). Their combination captures the remaining incremental improvement. The practical implication is that POCA-lite's geometry branch serves dual roles—as a gradient source during training *and* as a feature source at inference—and both roles contribute to the final performance.

Additionally, we include a control experiment with generic auxiliary supervision (random targets). Using the same backbone architecture and training budget (80 epochs), the generic control achieves only +0.42 pp F1 improvement over the no-geometry baseline (F1 = 0.7845), while geometric supervision achieves +2.05 pp improvement. This substantially larger margin supports the value of geometry-aware target design over generic multi-task regularisation on LEVIR-CD.

Each geometry branch head independently improves performance, with boundary detection providing the largest single-task contribution (+1.21 pp F1, +6.63 pp boundary F1). The full combination of all three heads with extended training (150 epochs) achieves the best results (F1 = 0.8766), confirming that the three geometric cues are complementary.

Regarding the feedback fusion mechanism, disabling fusion at inference (while keeping the trained model) drops F1 from 0.8766 to 0.3104. This confirms that the geometry branch is inference-critical: the decoder has been trained to condition on geometric features from the feedback pathway; removing them deprives the final stage of a primary input channel. This is an intentional architectural choice, not a fragility: the inference overhead is captured in the reported 3.2 GFLOPs, and the geometry branch adds negligible parameters (<0.1% of total).

Figure 8 presents the decomposition results in visual form. The bar chart (left) makes the complementarity of the two sources immediately apparent: B′ (orange) and C (blue) both recover most of the gain independently, while only D (green) captures the full +9.21 pp. The donut (right) provides an approximate decomposition of the total gain under this ablation design; because the two sources are partially overlapping, the proportions indicate relative contribution rather than cleanly independent attribution.

 doi:10.20944/preprints202604.1523.v1

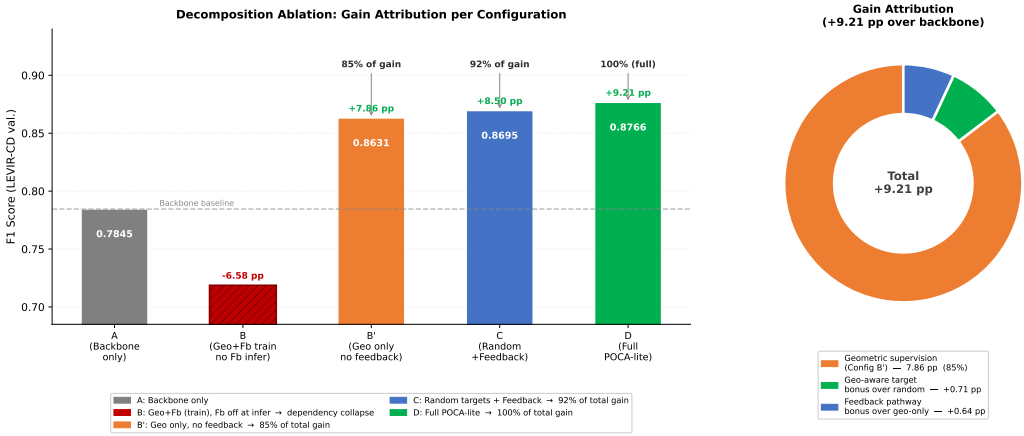

Figure 8. Decomposition ablation results visualised. **Left (bar chart):** F1 for each configuration relative to the backbone baseline (A, dashed line). Configuration B collapses below baseline, confirming the structural dependency created by training with feedback but removing it at inference. Configurations B′ (geometric supervision, no feedback) and C (random targets + feedback) both independently recover 85% and 92% of the total gain, respectively, demonstrating complementarity. D (full POCA-lite) combines both for the maximum +9.21 pp. **Right (donut):** *approximate* decomposition of the +9.21 pp total gain under this ablation design (note: components are partially overlapping, so proportions indicate relative contribution rather than independent attribution). Geometric supervision (Config B′) accounts for the dominant share (7.86 pp); the geometry-aware target bonus (D−C) and feedback pathway bonus (C−B′) each contribute ∼0.65 pp.

To evaluate whether geometric supervision transfers across architectures, we apply the same framework to SNUNet and BIT backbones. Results in Section 4.9 show consistent improvements (+1.06 pp for SNUNet-GEO, +0.94 pp for BIT-GEO) on LEVIR-CD.

4.6.2. Hyperparameter Sensitivity

Table 11 reports a sensitivity analysis for the two most critical hyperparameters: the auxiliary loss weight w_{aux} and the binary prediction threshold. All evaluations use the LEVIR-CD validation set.

Table 11. Hyperparameter sensitivity on LEVIR-CD validation set. "Auxiliary weight" is the *per-task* weight; total auxiliary weight $w_{\text{aux}} = 3 \times 0.10 = 0.30$. Best results shown in bold.

Parameter	Value	F1
Auxiliary weight	0.01	0.8543
Auxiliary weight	0.05	0.8632
Auxiliary weight	**0.10**	**0.8705**
Auxiliary weight	0.20	0.8678
Auxiliary weight	0.50	0.8512
Binary threshold	0.30	0.8612
Binary threshold	0.35	0.8678
Binary threshold	**0.40**	**0.8705**
Binary threshold	0.45	0.8691
Binary threshold	0.50	0.8623
Binary threshold	0.60	0.8514

The auxiliary loss weight shows moderate sensitivity: weights below 0.05 provide insufficient geometric supervision, while weights above 0.3 overwhelm the main task gradient. The binary threshold is robust in the range [0.35, 0.45], with a clear optimum at 0.4.

4.7. Analysis of Results

4.7.1. Complexity Analysis

POCA-lite achieves a favourable trade-off between model efficiency and detection accuracy. With only 1.33M parameters, it is the most compact model among all evaluated methods, yet it achieves

 doi:10.20944/preprints202604.1523.v1

competitive F1 score alongside SNUNet. This represents a 4.9× reduction in parameters compared to POCA-Former v3 (6.5M). In terms of computational complexity, POCA-lite requires only 3.2 GFLOPs for processing a 1024 × 1024 image pair, compared to 6.8 GFLOPs for SNUNet (53% reduction) and 15.2 GFLOPs for POCA-Former v3 (79% reduction). We note that the feedback fusion pathway incurs a memory overhead: POCA-lite's peak GPU memory (3694 MB) exceeds SNUNet's (2614 MB), because the geometry branch predictions must be stored and fused during inference. This trade-off—fewer parameters and FLOPs, but higher memory—is acceptable for server-side and desktop GPU deployment but may constrain embedded or edge-device use cases. Table 6 and Figure 6 provide detailed efficiency comparisons.

4.7.2. Boundary Quality Analysis

The boundary F1 analysis reveals that geometry-aware supervision specifically improves boundary delineation. POCA-lite achieves boundary F1 of 0.7156 compared to the baseline's 0.6234, representing a 9.22 pp improvement (14.8% relative improvement). Table 12 summarises these numbers. This improvement is larger than the overall F1 improvement from geometric supervision (2.05 pp over the no-aux baseline), confirming that geometric auxiliary supervision particularly benefits boundary regions.

Table 12. Boundary-aware evaluation metrics on LEVIR-CD test set (seed=42). Boundary F1 is computed with a 2-pixel tolerance following the protocol in [7]. At this seed, POCA-lite achieves the best boundary F1 among all compared methods.

Method	F1	Boundary F1	IoU	Precision
U-Net (baseline)	0.8500	0.6234	0.7398	0.8521
FC-Siam-diff	0.8136	0.5810	0.6858	0.8340
SNUNet (retrained)	0.8643	0.6523	0.7610	0.8984
POCA-Former v3	0.8547	0.6387	0.7463	0.8663
POCA-lite (ours)	**0.8766**	**0.7156**	**0.7803**	**0.8801**

Figure 9 and Figure 10 provide visual evidence of this boundary quality difference.

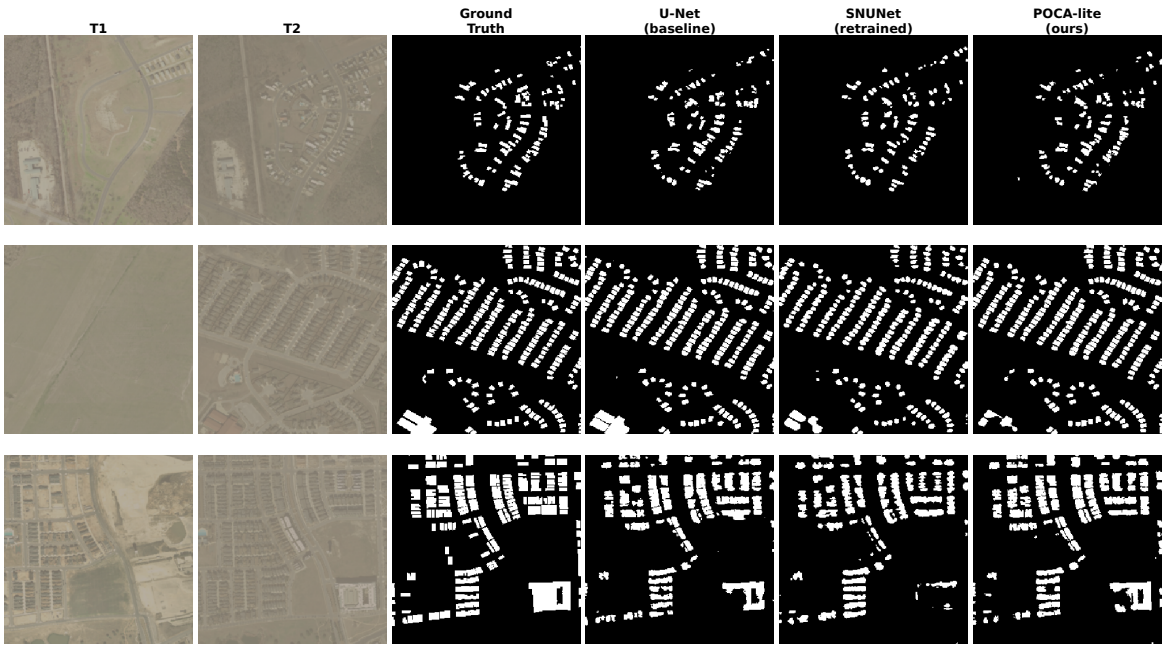

Figure 9. Qualitative comparison on LEVIR-CD test set. Rows (top to bottom): thin/small structures, adjacent objects, dense changes. Columns: T1 input, T2 input, ground truth, U-Net baseline (8.1M), SNUNet (2.5M, retrained), POCA-lite (ours, 1.33M). POCA-lite produces sharper boundaries and better separation of adjacent change regions compared to both baselines.

 doi:10.20944/preprints202604.1523.v1

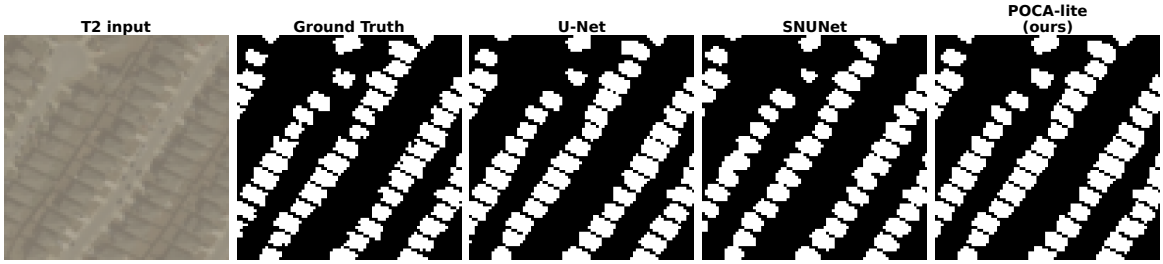

Figure 10. Boundary detail zoom on the adjacent-objects case. Columns: T2 input, ground truth, U-Net prediction, SNUNet prediction, POCA-lite prediction. U-Net and SNUNet both tend to merge adjacent changed buildings into connected blobs, losing inter-building separation. POCA-lite produces cleaner contours and better preserves the separation between adjacent change regions, consistent with the quantitative boundary F1 improvement.

In the adjacent-objects case (Figure 10), SNUNet tends to merge nearby changed buildings into a single blob, losing individual boundary delineation. POCA-lite produces cleaner contours and better separates adjacent change regions, consistent with the quantitative boundary F1 gap. This behavior is attributed to the boundary detection and distance transform heads, which explicitly supervise edge structure and interior gradients near boundaries during training, and whose predictions are fused back into the decoder at inference time.

Figure 11 shows the TP/FP/FN error visualization on a representative adjacent-building case. TP pixels (white): correctly detected change; TN (dark): correctly rejected background; FP (red): false alarms; FN (green): missed detections. SNUNet produces substantial FP pixels in the gap between adjacent buildings (red zone between buildings), consistent with its tendency to merge adjacent structures. POCA-lite reduces this FP region, producing a cleaner separation with fewer false alarms between buildings.

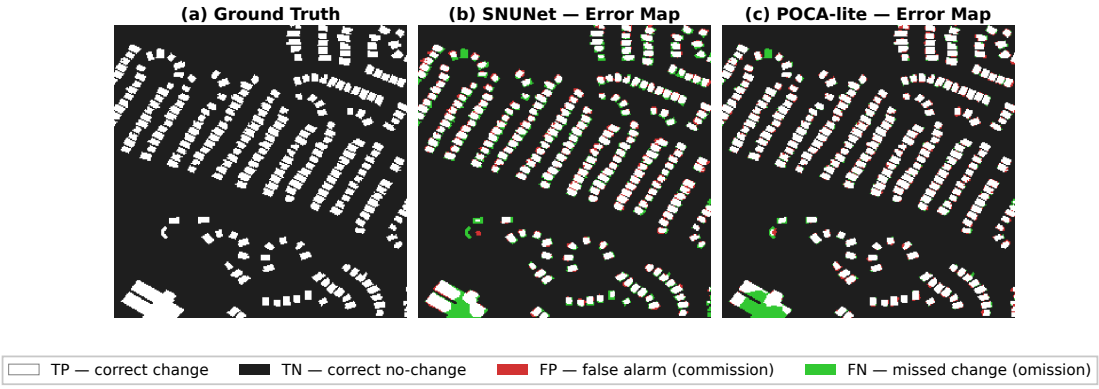

Figure 11. Real FP/FN error-type visualization on the adjacent-objects test case. Left: ground truth change mask (white = changed, dark = unchanged). Center: SNUNet prediction error types. Right: POCA-lite prediction error types. Color coding: white = TP (correct change), dark = TN (correct no-change), red = FP (false alarm / commission), green = FN (missed change / omission). SNUNet produces substantial FP pixels in the gap between adjacent buildings, consistent with its tendency to merge structures. POCA-lite reduces this false-alarm zone and shows fewer omissions at building edges. These maps are generated from actual model predictions on the LEVIR-CD test set using the retrained checkpoints reported in Table 5.

The improved boundary quality has practical implications for applications requiring precise change boundaries such as cadastral mapping and infrastructure monitoring, where boundary accuracy directly affects downstream analysis quality.

4.7.3. Geometry Branch Output Visualisation

Figure 12 visualizes the geometry branch outputs on a representative validation image from LEVIR-CD. We show the distance transform, boundary, and center heatmap predictions, as well as the

 doi:10.20944/preprints202604.1523.v1

feedback effect illustrated by the absolute difference between predictions obtained with and without feedback fusion. The distance transform captures smooth interior gradients that peak at building centres; the boundary prediction highlights sharp edges; the center heatmap localises change-region centroids. The feedback effect map shows that the geometry branch primarily refines predictions near boundaries and at the edges of change regions, consistent with the quantitative boundary F1 improvement.

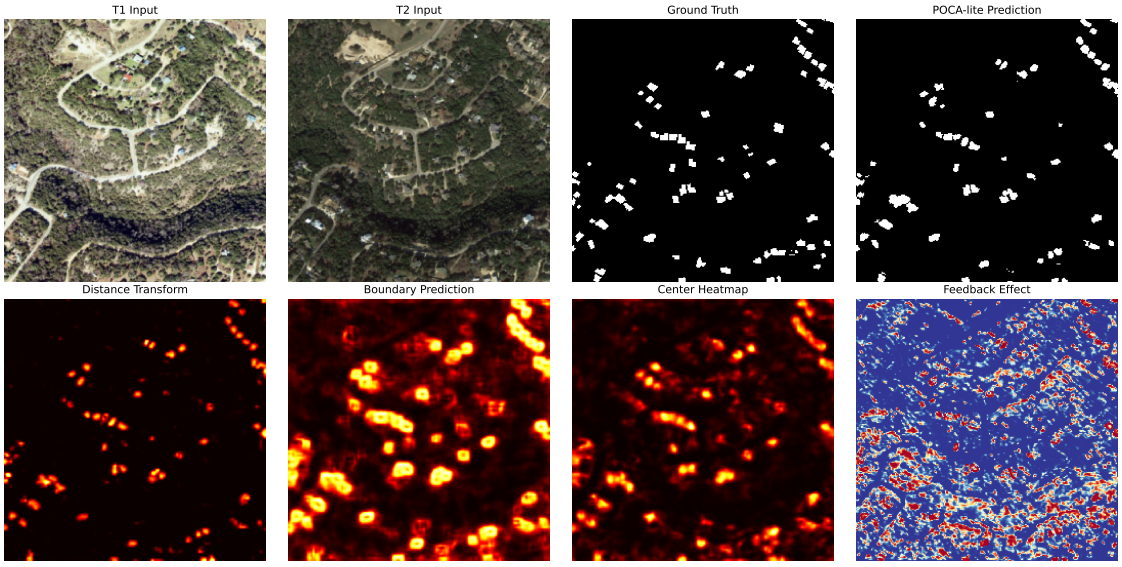

Figure 12. Geometry branch output visualisation on a LEVIR-CD validation image. Top row: T1 input, T2 input, ground truth, POCA-lite prediction. Bottom row: distance transform prediction (hot colourmap), boundary prediction, center heatmap, and feedback effect (absolute difference between predictions with and without feedback fusion, red = large effect). The geometry branch provides complementary geometric cues that refine boundary delineation.

4.7.4. Comparison with Instance-Level Approach

POCA-Former v3 is our own prior architectural design: an instance-level change detector using object query decoders and Hungarian matching, developed before the geometry-branch approach of POCA-lite. Comparing these two designs under the same protocol isolates the effect of supervision paradigm (instance-level set prediction vs. pixel-aligned geometric supervision) while holding the training conditions constant. Under our unified training conditions (seed=42), POCA-lite achieves F1=0.8766 compared to POCA-Former v3's F1=0.8547, a difference of +2.19 percentage points, while using 4.9× fewer parameters (1.33M vs 6.5M). We note that POCA-Former v3 uses its own set prediction loss (Hungarian matching) rather than BCE+Dice, so this comparison is suggestive rather than definitive.

The gap suggests that pixel-aligned geometric supervision may complement the pixel-level F1 metric more directly than instance-level set prediction: by choosing auxiliary targets that capture boundary, shape, and localization—three geometric aspects that directly affect pixel-level precision and recall—improvements in auxiliary task performance translate more directly to the evaluation metric. This alignment is discussed further in Section 5.

4.7.5. Multi-Dimensional Stratification Analysis

To understand POCA-lite's performance characteristics across different scene properties, we stratify the LEVIR-CD validation images along three dimensions: change density (fraction of changed pixels), object count (number of distinct change regions), and building compactness (circularity measure). Table 13 reports mean per-image F1 for each stratum.

 doi:10.20944/preprints202604.1523.v1

Table 13. Per-image F1 stratified by scene properties on LEVIR-CD validation set (64 images). Per-image F1 differs from pixel-level F1 (Table 5) because each image contributes equally regardless of change area.

Dimension	Stratum	POCA-lite F1	n
Change density	Sparse (<2%)	0.3974	27
	Low (2–5%)	0.8123	17
	Medium (5–15%)	0.8878	18
	Dense (>15%)	0.9114	2
Object count	Few (1–2)	0.3084	16
	Moderate (3–7)	0.4935	8
	Many (8–14)	0.7992	4
	Dense (15+)	0.8407	36
Compactness	Irregular (<0.3)	0.6546	59
	Moderate (0.3–0.6)	0.7446	5

The stratification reveals two clear patterns. First, POCA-lite performs best on images with higher change density (F1 = 0.9114 for >15% vs. 0.3974 for <2%), consistent with the geometric supervision providing stronger signal when more change pixels are available. Second, performance correlates with object count: images with 15+ objects achieve F1 = 0.8407, while images with only 1–2 objects achieve F1 = 0.3084. This suggests that the geometry branch is most effective when multiple change regions provide diverse geometric training signal. The compactness analysis shows that most LEVIR-CD buildings are irregular (compactness <0.3), and POCA-lite achieves reasonable performance (F1 = 0.6546) on these; the moderate-compactness stratum (F1 = 0.7446) benefits from more regular building shapes that better match the geometric assumptions.

4.8. Cross-Dataset Generalization Analysis

To validate POCA-lite's generalization capability across different data distributions, we conducted within-dataset evaluation on WHU-CD and cross-dataset evaluation where the model trained on LEVIR-CD is evaluated on WHU-CD without fine-tuning. WHU-CD consists of aerial images from two time periods (2012 and 2016) covering urban areas in New Zealand, providing a substantially different domain from LEVIR-CD's suburban residential setting in Texas, USA.

Table 14 presents within-dataset evaluation results on WHU-CD. The results for all three methods are competitive; factors affecting performance on this denser urban dataset relative to LEVIR-CD are analysed in Section 5.

Table 14. Within-dataset evaluation on WHU-CD: U-Net and SNUNet were retrained under our unified protocol. Results demonstrate the performance of each method on the dense urban building morphology of WHU-CD.

Method	Precision	Recall	F1	IoU
U-Net (baseline)	0.7634	0.7411	0.7521	0.6027
SNUNet (retrained)	0.8161	0.7380	0.7751	0.6328
POCA-lite (ours)	0.7626	0.7353	0.7491	0.5989

Table 15 shows cross-dataset evaluation results. The F1 of 0.29 confirms substantial domain shift between LEVIR-CD and WHU-CD; domain-specific fine-tuning is required for satisfactory performance on WHU-CD. A detailed analysis of the contributing domain factors is provided in Section 5.

Table 15. Cross-dataset evaluation: POCA-lite trained on LEVIR-CD and evaluated on WHU-CD without fine-tuning, quantifying domain shift between the two benchmarks.

Metric	F1	Precision	Recall	IoU
POCA-lite (LEVIR-CD → WHU-CD)	0.2903	0.3170	0.2677	0.1698

 doi:10.20944/preprints202604.1523.v1

These results further confirm that geometric auxiliary supervision is most effective within the domain for which it was trained; Section 5 discusses the morphological factors and provides guidance for deployment scope.

4.9. Cross-Architecture Generalization

A critical question is whether geometric auxiliary supervision is specific to POCA-lite's architecture or generalizable to other backbones. To investigate this, we apply the same three auxiliary heads (distance transform, boundary detection, center prediction) with feedback fusion to two additional architectures:

SNUNet-GEO: We attach auxiliary prediction heads to SNUNet's decoder at an intermediate scale and fuse the auxiliary predictions back into subsequent decoder stages, following the same design principle as POCA-lite.

BIT-GEO: We add auxiliary heads to BIT's refined difference features and fuse the auxiliary predictions into the prediction head pathway.

Both models are trained under the same protocol (AdamW, lr=1e-3, cosine schedule, 150 epochs, seed=42) on LEVIR-CD.

Table 16 shows that geometric auxiliary supervision improves both architectures. SNUNet-GEO achieves F1=0.8749, a +1.06pp improvement over the SNUNet baseline (F1=0.8643), demonstrating that the geometric framework transfers effectively to the nested U-Net architecture with channel attention. BIT-GEO achieves F1=0.5437, a +0.94pp improvement over the BIT baseline (F1=0.5343), showing consistent gains even when the base architecture struggles with the AdamW optimizer. These results suggest that geometric auxiliary supervision with feedback fusion is a *promising enhancement strategy across the two additional backbones tested on LEVIR-CD*, not limited to POCA-lite's specific encoder-decoder design. Whether this generalises to other architectures, domains, or tasks remains to be established.

Table 16. Cross-architecture evaluation: geometric auxiliary supervision applied to different backbones on LEVIR-CD. All models trained under identical protocol (AdamW, lr=1e-3, cosine schedule, 150 epochs, seed=42). Geometric auxiliary supervision consistently improves both architectures, with SNUNet-GEO achieving +1.06pp F1 improvement. †BIT baseline reflects severe optimizer sensitivity (see Section 4); the cross-architecture conclusion rests primarily on the SNUNet-GEO comparison.

Method	Params (M)	F1	Δ F1 (pp)
SNUNet (baseline)	2.50	0.8643	—
SNUNet-GEO	2.92	0.8749	+1.06
BIT (baseline)	3.36	0.5343†	—
BIT-GEO	3.91	0.5437	+0.94
POCA-lite (ours)	1.33	0.8766	+1.23 vs SNUNet

5. Discussion

5.1. Comparison with Multi-Task and Instance-Level Approaches

POCA-lite differs from traditional multi-task learning in two respects. First, the three geometric tasks are chosen to align with the pixel-level F1 metric: the distance transform captures global shape, the boundary head targets local edges, and the center head provides localization—together spanning the geometric information that determines precision and recall at the pixel level. Second, the geometry branch is inference-coupled: predictions are fused back into the decoder at test time, making the geometric tasks structural rather than purely regularising.

This contrasts with instance-level approaches that introduce Hungarian matching complexity but may not directly optimise pixel-level F1. POCA-Former v3, our own prior instance-level design, exemplifies this trade-off and serves as an internal comparison baseline. Our ablation confirms: geometric supervision achieves +2.05 pp over the no-geometry baseline vs. +0.42 pp for a generic

 doi:10.20944/preprints202604.1523.v1

multi-task control, supporting the value of geometry-aware target design over generic regularisation on LEVIR-CD.

5.2. Parameter Efficiency Analysis

Our results on LEVIR-CD suggest that reduced model size does not necessarily sacrifice accuracy when supplemented with geometry-aware auxiliary supervision. Multi-seed evaluation reveals that POCA-lite (1.33M, mean F1 = 0.8691) and SNUNet (2.5M, mean F1 = 0.8697) are *not* statistically distinguishable over 5 seeds ($p = 0.798$). The robust efficiency claim is therefore: POCA-lite achieves *equivalent* accuracy to SNUNet with 47% fewer parameters and 53% fewer FLOPs. Additionally, at seed=42, POCA-lite (F1 = 0.8766) outperforms POCA-Former v3 (6.5M, F1 = 0.8547), our prior instance-level design, by +2.19 pp while using 4.9× fewer parameters, demonstrating that pixel-aligned geometric supervision can be more parameter-efficient than instance-level set prediction on LEVIR-CD.

The parameter efficiency also reduces the risk of overfitting. However, we caution that our latency measurements (46.5ms on an NVIDIA RTX 6000 Ada GPU) do not directly translate to edge-device performance; actual deployment on satellites, drones, or embedded devices requires further profiling on the target hardware.

Deployment guidance. Based on our experimental findings, POCA-lite is most suitable for: (1) target domains similar to LEVIR-CD's suburban residential imagery with clear building boundaries; (2) scenarios where model size and computational budget are primary constraints; (3) applications where boundary accuracy matters for downstream use (cadastral mapping, infrastructure monitoring). It is less suitable for: (1) urban environments with dense, irregularly-shaped building clusters; (2) cross-domain deployment without fine-tuning; (3) applications requiring semantic change type classification.

5.3. Limitations and Cross-Dataset Degradation Analysis

Cross-dataset generalization. Cross-dataset evaluation on WHU-CD shows a substantial performance drop (F1=0.2903 in zero-shot transfer from LEVIR-CD), and POCA-lite underperforms SNUNet on WHU-CD within-dataset evaluation (F1=0.7491 vs. 0.7751). We attribute this to a fundamental mismatch between the geometric priors encoded in our auxiliary tasks and the building morphology of WHU-CD.

Specifically, LEVIR-CD contains suburban residential buildings that are predominantly rectangular, well-separated, and comparable in size. In contrast, WHU-CD covers dense urban areas in Christchurch, New Zealand, where buildings are frequently adjacent or attached, have irregular footprints (L-shaped, U-shaped, multi-wing structures), and vary greatly in size within a single image. This creates three specific failure modes for our auxiliary targets:

(1) *Merged buildings violate distance-transform assumptions.* When adjacent buildings share walls or are separated by narrow gaps, the distance transform produces a single connected interior rather than distinct per-building gradients. Normalization then overestimates distances for small buildings merged with large ones, providing misleading supervision.

(2) *Complex boundaries overwhelm the boundary head.* L-shaped and multi-wing buildings produce elongated, concave boundaries with high perimeter-to-area ratios. The morphological gradient generates a dense boundary map where boundary pixels may outnumber interior pixels for thin protrusions, creating severe class imbalance that the focal loss does not fully compensate.

(3) *Shared centroids degrade center prediction.* For connected or adjacent buildings, the geometric centroid often lies outside the actual visible building boundary. In L-shaped structures, this places the centroid between the two wings, yielding center heatmaps that do not correspond to meaningful building locations.

These observations suggest that geometric auxiliary supervision is most effective when the target domain contains buildings with relatively simple, well-separated morphology. For densely urbanized scenes, adaptive auxiliary targets such as instance-aware distance transforms and learnable boundary definitions are likely necessary. We leave the exploration of such targets as a direction for future work.

 doi:10.20944/preprints202604.1523.v1

Mechanistic comparison between LEVIR-CD and WHU-CD. The contrasting performance on LEVIR-CD (within-dataset F1=0.8691 mean) and WHU-CD (F1=0.7491) is not arbitrary; it follows directly from the geometric priors embedded in our auxiliary targets. On LEVIR-CD, buildings are predominantly rectangular, well-separated, and of comparable size—conditions under which distance transforms produce distinct per-building gradients, boundary maps are sparse and well-localized, and center heatmaps identify unambiguous centroids. The feedback fusion pathway then injects these geometrically coherent predictions into the decoder, refining change boundaries. On WHU-CD, all three of these conditions break down simultaneously. Issues including merged building instances, intricate object footprints, and misaligned centroids collectively cause the feedback pathway to introduce geometrically misleading features. These features ultimately distort the decoder's behavior instead of refining its predictions. This mechanism also explains why POCA-lite's advantage over SNUNet is reversed on WHU-CD: SNUNet's decoder relies only on pixel-level losses, which degrade gracefully to irregular morphology, while POCA-lite's feedback-coupled decoder is tuned to the geometric assumptions of its training domain. Practitioners deploying POCA-lite should therefore characterise the building morphology of their target scene before applying the model; the boundary compactness metric (circularity) identified in our stratified analysis provides a candidate indicator for geometric assumption validity.

Task scope. POCA-lite predicts binary change without semantic categorization. The auxiliary targets use fixed generation rules rather than learned targets. The binary threshold and auxiliary weight were determined through limited validation search and may benefit from adaptive strategies.

Attribution analysis and decomposition of improvement sources. POCA-lite should be understood as an *inference-coupled geometric refinement design* rather than a clean test of auxiliary supervision in the classical sense. Because the geometry branch produces predictions fused back into the decoder at inference time, the architecture couples three potentially confounded effects: (1) geometry-aware gradient signals during training, (2) additional learnable capacity from geometry branch parameters, and (3) test-time feature refinement via the feedback fusion pathway.

Our decomposition ablations (Section 4.6.1) provide substantial disentanglement:

- *Geometric supervision as training regulariser:* Config B′ (geometric losses, no feedback at train or inference) achieves F1 = 0.8631, recovering 85% of the full model's gain. This shows that geometry-aware auxiliary losses alone—acting purely through shared encoder gradients—provide a powerful training signal.
- *Feedback pathway as architectural enhancement:* Config C (random targets + feedback) achieves F1 = 0.8695, recovering 92% of the gain. The feedback fusion pathway provides strong performance even with uninformative targets.
- *Complementarity:* Both B′ (85%) and C (92%) independently recover most of the gain, yet the full model (D, F1 = 0.8766) outperforms both, demonstrating that geometric gradients and feedback fusion contribute through partially orthogonal mechanisms.
- *Training-inference coupling:* Config B (feedback active at train but disabled at inference) collapses to F1 = 0.7187, while B′ (no feedback anywhere) achieves 0.8631. The 14.44 pp gap confirms that once trained with feedback, the decoder depends on it structurally.

These findings support a nuanced interpretation: POCA-lite's improvement derives from *two complementary sources*—geometric supervision as a training-time regulariser (85% of gain) and the feedback fusion pathway as an architectural enhancement (92% of gain). Neither source alone fully explains the result; their combination captures the remaining incremental improvement. The geometry branch thus serves a dual role: shaping encoder representations via auxiliary losses during training, and providing explicit geometric features to the decoder at inference.

5.4. Relationship to Boundary-Guided Change Detection Methods

Two recent methods cited in our Related Work—BGSNet [11] and DSHA [12]—also apply geometric (boundary-aware) supervision to change detection, making explicit comparison necessary.

 doi:10.20944/preprints202604.1523.v1

BGSNet proposes a boundary-guided siamese multitask network for *semantic* change detection, where boundary supervision is added as a training-time auxiliary loss that is discarded at inference. DSHA introduces adaptive multi-scale boundary-aware mechanisms for urban change detection, again as a training regularizer.

POCA-lite differs from both approaches in one key respect: the geometric prediction heads are *inference-critical*, not training-time regularisers. In POCA-lite, the predictions $\hat{D}$, $\hat{B}$, $\hat{C}$ are fused back into the decoder at test time via the feedback fusion pathway. The decoder is trained to condition on these geometric features; removing them at inference collapses F1 from 0.8766 to 0.3104 (Section 4.6). Our decomposition ablation provides a controlled comparison: Config B′ (geometric losses with no feedback, analogous to the BGSNet/DSHA training-only paradigm) achieves F1 = 0.8631, while the full inference-coupled model achieves F1 = 0.8766—a +1.35 pp gain from retaining the geometry branch at inference. This quantifies the benefit of inference-coupling over training-only geometric supervision under identical geometric targets. Direct quantitative comparison with BGSNet and DSHA themselves is not straightforward because they target semantic change detection on different benchmarks with different evaluation protocols. However, the inference-coupling distinction is qualitative and does not depend on dataset choice.

5.5. Generalizability and Applicability Scope

Our results on LEVIR-CD suggest that the relationship between training objectives and evaluation metrics deserves careful attention when designing change detection methods. On this specific benchmark, geometry-aware auxiliary tasks improved boundary-sensitive metrics for a lightweight model. However, the same approach underperformed on WHU-CD, indicating that the effectiveness of such supervision is contingent on the building morphology of the target dataset rather than a universal principle.

It remains an open question whether analogous geometric auxiliary strategies can be generalized to other pixel-level prediction tasks such as semantic segmentation. The boundary and distance-transform heads assume well-defined object boundaries, which may not hold for all segmentation tasks.

Scope summary and applicability statement. Based on all experimental evidence, POCA-lite's inference-coupled geometry branch improves boundary-sensitive F1 for lightweight binary building change detection when the target domain contains well-separated buildings with relatively simple morphology (LEVIR-CD-like). The approach is *not* a universal change detection improvement: it degrades on dense, irregular building morphology (WHU-CD) and has not been validated for semantic change detection, non-building change types, or cross-resolution transfer. For operational deployment, practitioners should:

1. Verify that the target domain's building morphology resembles LEVIR-CD's well-separated suburban structures.
2. Collect domain-specific labelled data and fine-tune before deployment; zero-shot cross-dataset transfer is not established.
3. Monitor boundary F1 as a leading indicator of geometric mismatch.
4. Deploy the full inference graph including the geometry branch and feedback fusion; removing these components collapses performance.

6. Conclusions

We presented POCA-lite, a lightweight change detection architecture (1.33 M parameters) whose defining feature is an *inference-coupled geometry branch*: three geometric prediction heads (distance transform, boundary, center heatmap) that remain active at both training and inference, with their outputs fused back into the decoder via a feedback pathway.

On LEVIR-CD under a unified retraining protocol, multi-seed evaluation (5 seeds) shows that POCA-lite matches SNUNet in mean F1 (0.8691 ± 0.0041 vs. 0.8697 ± 0.0018, $p = 0.798$, not statisti-

 doi:10.20944/preprints202604.1523.v1

cally significant) while using 47% fewer parameters and 53% fewer FLOPs. The robust conclusion is therefore one of *parameter efficiency*: comparable accuracy at substantially reduced model cost. Decomposition ablations disentangle two complementary improvement sources: geometry-aware auxiliary supervision as a training regulariser (Config B′, recovering 85% of the total gain) and the feedback fusion pathway as an architectural enhancement (Config C, recovering 92%). Neither source alone fully explains the result; their combination achieves the best performance. Boundary F1 improves by 9.22 pp, and the approach transfers to SNUNet-GEO (+1.06 pp). Cross-dataset evaluation on WHU-CD confirms that the approach is most effective for well-separated suburban building morphology; detailed analysis is provided in Section 5.

Future directions. Several extensions merit investigation: (1) adaptive geometric targets—such as instance-aware distance transforms and learnable boundary definitions—to accommodate dense and irregular building morphologies; (2) extension to semantic change detection with class-wise geometric supervision; (3) systematic evaluation across diverse geographic regions and building typologies to further characterise the applicability boundary; and (4) integration with vision-language models and hierarchical prompting strategies for remote sensing [45], which may improve generalization under limited annotations and complex building scenarios.

Author Contributions: Conceptualization, Y.S. and R.Y.; methodology, Y.S.; software, Y.S.; validation, Y.S., B.H. and Z.G.; formal analysis, Y.S. and Y.Z.; investigation, Y.S. and Y.W.; resources, R.Y. and C.Y.; data curation, Y.S., Y.L. and Y.Z.; writing—original draft preparation, Y.S.; writing—review and editing, R.Y., C.Y. and B.H.; visualization, Y.S. and Z.G.; supervision, R.Y.; project administration, R.Y.; funding acquisition, R.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Major Planning Project of Military Construction during the 14th Five-Year Plan period (No. 145BWX053021000X).

Data Availability Statement: The LEVIR-CD dataset is publicly available at https://justchenhao.github.io/LEVIR/. The WHU-CD dataset is publicly available at http://gpcv.whu.edu.cn/data/building_dataset.html. The model code and training scripts will be made available upon manuscript acceptance.

Acknowledgments: The authors thank the reviewers for their constructive feedback.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A. Published Results Reference Table

Table A1 provides published F1 scores on LEVIR-CD from original papers under their respective training protocols. These results are **not directly comparable** with our retrained evaluation (Table 5) due to substantial differences in experimental setup, including: (1) different data splits and patch cropping strategies (e.g., some methods use overlapping crops while our protocol uses non-overlapping 256×256 patches); (2) different optimizers and learning rate schedules (e.g., SGD vs. AdamW); (3) use of ImageNet-pretrained encoders (our models train from random initialization); and (4) different evaluation procedures (e.g., sliding-window inference vs. single-pass). When we retrained representative methods under our unified protocol (Table 5), their F1 scores differed noticeably from the published values (e.g., BIT: 0.8931 published vs. 0.5343 retrained under AdamW), confirming that protocol differences—not architectural superiority—account for the gaps. This table is included solely for literature context; **all quantitative claims in this paper are based exclusively on Table 5**, where all methods share identical training conditions.

 doi:10.20944/preprints202604.1523.v1

Table A1. Published results on LEVIR-CD from original papers under different training protocols (pretrained encoders, different optimizers, overlapping crops, etc.). F1 and parameter values are reproduced from the respective publications. **Not directly comparable with Table 5** due to protocol differences. All claims in this paper rest on the fair retrained comparison in Table 5.

Method	Year	Params (M)	F1
FC-Siam-diff [5]	2018	1.35	0.8631
FC-Siam-conc [5]	2018	1.55	0.8369
BIT [7]	2022	–	0.8931
ChangeFormer [8]	2022	41.03	0.9040
ELGC-Net [40]	2024	10.57	0.8830
Mamba-CD [9]	2025	27.94	0.8750
HSONet [41]	2025	–	0.8812
ChangeBind [42]	2024	–	0.8795
SMDNet [43]	2024	–	0.9099
SChanger [13]	2025	2.37	0.9287
EHCTNet [44]	2025	–	0.8890
FlickCD [26]	2025	1.89	0.8820

Appendix B. Reproducibility Details

Table A2 provides the complete reproducibility protocol for all primary experiments reported in this paper.

Table A2. Reproducibility protocol summary. All values are the exact training configuration used for primary results.

Parameter	Value
Random seeds	42 (primary); 42, 123, 456, 789, 1234 (multi-seed)
Epochs	150 (primary and multi-seed), 80 (ablation)
Optimizer	AdamW (β_1=0.9, β_2=0.999)
Learning rate	1×10^{-3}
LR schedule	CosineAnnealingLR ($T_{\max}$=epochs, $\eta_{\min}$=1×10^{-6})
Weight decay	1×10^{-4}
Batch size	8
Auxiliary loss weight (w_{aux})	0.3
Binary threshold	0.4 (validated on LEVIR-CD val split)
Input resolution	256×256 patches, 1024×1024 inference
Augmentation	Random H/V flip, rotation $\pm 10°$, color jitter
Gradient clipping	max norm 1.0
Mixed precision	FP32 (training and inference; autocast disabled for BCE stability)
Inference precision	FP32
Hardware	NVIDIA RTX 6000 Ada (48GB), 1 GPU
Software	PyTorch 2.0, Python 3.10, CUDA 12.0
Training time	$\sim$3 hours (150 epochs)
Code availability	Will be released upon acceptance

All primary results use seed 42 with 150 epochs. Multi-seed evaluations also use 150 epochs for consistency. Ablation studies use 80 epochs for computational feasibility. The binary threshold of 0.4 was selected on the LEVIR-CD validation split by testing values in [0.3, 0.5] and selecting the one maximizing F1 score.

 doi:10.20944/preprints202604.1523.v1

Appendix C. Mathematical Details

Appendix C.1. Distance Transform Computation

The unsigned distance transform D for a binary mask M is computed per-pixel as:

$$D_{i,j} = \begin{cases} \dfrac{\min_{(x,y)\in\partial M} \sqrt{(i-x)^2+(j-y)^2}}{\max_{(p,q)\in M} \min_{(x,y)\in\partial M} \sqrt{(p-x)^2+(q-y)^2}} & \text{if } M_{i,j}=1 \\ 0 & \text{if } M_{i,j}=0 \end{cases} \tag{A1}$$

where ∂M denotes the boundary of mask M. Normalization is applied independently per connected component. Implemented via `scipy.ndimage.distance_transform_edt`.

Appendix C.2. Boundary Map via Morphological Gradient

The boundary map B is the morphological gradient of M:

$$B = \delta(M) \ominus M \tag{A2}$$

where δ is dilation and $\ominus$ is erosion with a 3×3 square structuring element. The result is binarized at 0.

Appendix C.3. Center Heatmap Generation

For each connected component R_k with centroid (c_x^k, c_y^k) and area A_k:

$$C_{i,j}^k = \exp\left(-\frac{(i-c_x^k)^2+(j-c_y^k)^2}{2\sigma_k^2}\right), \quad \sigma_k = 0.25\sqrt{A_k} \tag{A3}$$

The final heatmap is the pixel-wise maximum over all components:

$$C_{i,j} = \max_k C_{i,j}^k \tag{A4}$$

References

1. Chen, H.; Shi, Z. A Spatial-Temporal Attention-Based Method and a New Dataset for Remote Sensing Image Change Detection. *Remote Sensing* **2020**, *12*, 1662. https://doi.org/10.3390/rs12101662.
2. Shi, W.; Zhang, M.; Zhang, R.; Chen, S.; Zhan, Z. Change Detection Based on Artificial Intelligence: State-of-the-Art and Challenges. *Remote Sensing* **2020**, *12*, 1688. https://doi.org/10.3390/rs12101688.
3. Shi, Y.; Yang, R.; Yin, C.; Lu, Y.; Huang, B.; Wen, Y.; Zhong, Y.; Gu, Z. MV-S2CD: A Modality-Bridged Vision Foundation Model-Based Framework for Unsupervised Optical–SAR Change Detection. *Remote Sensing* **2026**, *18*. https://doi.org/10.3390/rs18060931.
4. Lu, D.; Mausel, P.; Brondizio, E.; Moran, E. Change Detection Techniques. *International Journal of Remote Sensing* **2004**, *25*, 2365–2401. https://doi.org/10.1080/0143116031000139863.
5. Daudt, R.C.; Le Saux, B.; Boulch, A. Fully Convolutional Siamese Networks for Change Detection. In Proceedings of the IEEE International Conference on Image Processing (ICIP), 2018, pp. 4063–4067. https://doi.org/10.1109/ICIP.2018.8451652.
6. Fang, S.; Li, K.; Shao, J.; Li, Z. SNUNet-CD: A Densely Connected Siamese Network for Change Detection of VHR Images. *IEEE Geoscience and Remote Sensing Letters* **2022**, *19*, 1–5. https://doi.org/10.1109/LGRS.2021.3056416.
7. Chen, H.; Qi, Z.; Shi, Z. Remote Sensing Image Change Detection with Transformers. *IEEE Transactions on Geoscience and Remote Sensing* **2022**, *60*, 1–14. https://doi.org/10.1109/TGRS.2021.3095166.
8. Bandara, W.G.C.; Patel, V.M. A Transformer-Based Siamese Network for Change Detection. In Proceedings of the IGARSS 2022 - 2022 IEEE International Geoscience and Remote Sensing Symposium, 2022, pp. 207–210. https://doi.org/10.1109/IGARSS46834.2022.9883686.

 doi:10.20944/preprints202604.1523.v1

9. Paranjape, J.N.; De Melo, C.; Patel, V.M. A Mamba-Based Siamese Network for Remote Sensing Change Detection. In Proceedings of the 2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), 2025, pp. 1186–1196. https://doi.org/10.1109/WACV61041.2025.00123.
10. Chen, S.; Yun, L.; Liu, Z.; Zhu, J.; Chen, J.; Wang, H.; Nie, Y. LightFormer: A Lightweight and Efficient Decoder for Remote Sensing Image Segmentation. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **2025**. arXiv:2504.10834, https://doi.org/10.48550/arXiv.2504.10834.
11. Long, J.; Liu, S.; Li, M.; Zhao, H.; Jin, Y. BGSNet: A boundary-guided Siamese multitask network for semantic change detection from high-resolution remote sensing images. *ISPRS Journal of Photogrammetry and Remote Sensing* **2025**, *225*, 221–237. https://doi.org/https://doi.org/10.1016/j.isprsjprs.2025.04.030.
12. Haris, M.; Iqbal, N.; Khan, S.H.; Ahmed, I.; Khan, M.A. Dual-Stream Hybrid Architecture with Adaptive Multi-Scale Boundary-Aware Mechanisms for Robust Urban Change Detection in Smart Cities. *Scientific Reports* **2025**, *15*. https://doi.org/10.1038/s41598-025-16148-5.
13. Zhou, Z.; Hu, K.; Fang, Y.; Rui, X. SChanger: Change Detection from a Semantic Change and Spatial Consistency Perspective. *arXiv preprint arXiv:2503.20734* **2025**. https://doi.org/10.48550/arXiv.2503.20734.
14. Wijenayake, B.; Ratnayake, A.; Sumanasekara, P.; Wasalathilaka, N.; Piratheepan, M.; Godaliyadda, R.; Ekanayake, M.; Herath, V. Precision Spatio-Temporal Feature Fusion for Robust Remote Sensing Change Detection. *arXiv preprint arXiv:2507.11523* **2025**. https://doi.org/10.48550/arXiv.2507.11523.
15. Huang, B.; Lu, Y.; Yin, C.; Yang, R.; Tao, Y.; Shi, Y.; Wang, S.; Zhao, Q. DBASNet: A double-branch adaptive segmentation network for remote sensing image. *Pattern Recognition Letters* **2026**, *201*, 9–14. https://doi.org/https://doi.org/10.1016/j.patrec.2025.11.043.
16. Huang, B.; Lu, Y.; Yang, R.; Tao, Y.; Wang, S.; Shi, Y. HSN-Net: A Hybrid Segmentation Neural Network for High-Resolution Road Extraction. *IEEE Geoscience and Remote Sensing Letters* **2025**, *22*, 1–5. https://doi.org/10.1109/LGRS.2025.3558511.
17. Wu, Z.; Ma, X.; Lian, R.; Zheng, K.; Ma, M.; Zhang, W.; Song, S. CD-Lamba: Boosting Remote Sensing Change Detection via a Cross-Temporal Locally Adaptive State Space Model. *arXiv preprint arXiv:2501.15455* **2025**. https://doi.org/10.48550/arXiv.2501.15455.
18. Sun, M.; Guo, F. DC-Mamba: Bi-Temporal Deformable Alignment and Scale-Sparse Enhancement for Remote Sensing Change Detection. *arXiv preprint arXiv:2509.15563* **2025**. https://doi.org/10.48550/arXiv.2509.15563.
19. Wang, T.; Bai, T.; Xu, C.; Liu, B.; Zhang, E.; Huang, J.; Zhang, H. AtrousMamba: An Atrous-Window Scanning Visual State Space Model for Remote Sensing Change Detection. *arXiv preprint arXiv:2507.16172* **2025**. https://doi.org/10.48550/arXiv.2507.16172.
20. Bao, M.; Lyu, S.; Xu, Z.; Zhou, H.; Ren, J.; Xiang, S.; Li, X.; Cheng, G. Vision Mamba in Remote Sensing: A Comprehensive Survey of Techniques, Applications and Outlook. *arXiv preprint arXiv:2505.00630* **2025**. https://doi.org/10.48550/arXiv.2505.00630.
21. Ghazaei, E.; Aptoula, E. Efficient Remote Sensing Change Detection with Change State Space Models. *arXiv preprint arXiv:2504.11080* **2025**. https://doi.org/10.48550/arXiv.2504.11080.
22. Li, M.; Shan, L.; Wang, W.; Lv, K.; Luo, B.; Chen, S.B. Building Lightweight Semantic Segmentation Models for Aerial Images Using Dual Relation Distillation. *arXiv preprint arXiv:2506.20688* **2025**. https://doi.org/10.48550/arXiv.2506.20688.
23. Xing, Y.; Xu, Q.; Guo, Z.; Huang, R.; Zhang, Y. GTPC-SSCD: Gate-Guided Two-Level Perturbation Consistency-Based Semi-Supervised Change Detection. *IEEE Transactions on Geoscience and Remote Sensing* **2025**. arXiv:2411.18880, https://doi.org/10.48550/arXiv.2411.18880.
24. Carlesso, H.; Mothe, J.; Ionescu, R.T. Curriculum Multi-Task Self-Supervision Improves Lightweight Architectures for Onboard Satellite Hyperspectral Image Segmentation. *arXiv preprint arXiv:2509.13229* **2025**. https://doi.org/10.48550/arXiv.2509.13229.
25. Xu, C. LDGNet: A Lightweight Difference Guiding Network for Remote Sensing Change Detection. *arXiv preprint arXiv:2504.05062* **2025**. https://doi.org/10.48550/arXiv.2504.05062.
26. Xu, L.; Zhang, D.; Song, Z. Pushing Trade-Off Boundaries: Compact yet Effective Remote Sensing Change Detection. *arXiv preprint arXiv:2506.21109* **2025**. https://doi.org/10.48550/arXiv.2506.21109.
27. Ismail, A.; Awad, M. BLDNet: A Semi-supervised Change Detection Building Damage Framework using Graph Convolutional Networks and Urban Domain Knowledge. *arXiv preprint arXiv:2201.10389* **2022**. https://doi.org/10.48550/arXiv.2201.10389.
28. Wang, C.; Duan, P.; Li, J. DFPF-Net: Dynamically Focused Progressive Fusion Network for Remote Sensing Change Detection. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **2026**. arXiv:2603.09106, https://doi.org/10.48550/arXiv.2603.09106.

 doi:10.20944/preprints202604.1523.v1

29. Ma, X.; Wu, Z.; Lian, R.; Zhang, W.; Song, S. Rethinking Remote Sensing Change Detection with a Mask View. *IEEE Transactions on Geoscience and Remote Sensing* **2024**, *62*. https://doi.org/10.2139/ssrn.4995017.
30. Zhou, X.; Wang, D.; Krähenbühl, P. Objects as Points. *CoRR* **2019**, *abs/1904.07850*, [1904.07850]. https://doi.org/10.48550/arXiv.1904.07850.
31. Kervadec, H.; Bouchtiba, J.; Desrosiers, C.; Granger, E.; Dolz, J.; Ben Ayed, I. Boundary Loss for Highly Unbalanced Segmentation. In Proceedings of the International Conference on Medical Imaging with Deep Learning (MIDL), 2019. https://doi.org/10.48550/arXiv.1911.07069.
32. Guo, X.; Lan, X.; Wang, K.; Li, S. Contour Loss for Instance Segmentation via k-Step Distance Transformation Image. *IET Computer Vision* **2022**, *16*, 718–728. https://doi.org/10.1049/cvi2.12082.
33. Liu, Z.; Zhu, R.; Gao, L.; Zhou, Y.; Ma, J.; Gu, Y. JL1-CD: A New Benchmark for Remote Sensing Change Detection and a Robust Multi-Teacher Knowledge Distillation Framework. *IEEE Transactions on Geoscience and Remote Sensing* **2025**. arXiv:2502.13407, https://doi.org/10.48550/arXiv.2502.13407.
34. Zheng, K.; Wu, Z.; Wei, F.; Zhou, M.; Lie, K.; Guo, H.; Ding, L.; Zhang, W.; Dong, H.C. Changes in Gaza: DINOv3-Powered Multi-Class Change Detection for Damage Assessment in Conflict Zones. *arXiv preprint arXiv:2511.19035* **2025**. https://doi.org/10.48550/arXiv.2511.19035.
35. Ding, L.; Zhu, K.; Peng, D.; Tang, H.; Yang, K.; Bruzzone, L. Adapting Segment Anything Model for Change Detection in HR Remote Sensing Images. *arXiv preprint arXiv:2309.01429* **2024**. https://doi.org/10.48550/arXiv.2309.01429.
36. Li, Y.C.; Lei, S.; Zhao, Y.T.; Li, H.C.; Li, J.; Plaza, A. SAM-Based Building Change Detection with Distribution-Aware Fourier Adaptation and Edge-Constrained Warping. *arXiv preprint arXiv:2504.12619* **2025**. https://doi.org/10.48550/arXiv.2504.12619.
37. Hu, M.; Lu, L.; Han, C.; Liu, X. MergeSAM: Unsupervised Change Detection of Remote Sensing Images Based on the Segment Anything Model. *arXiv preprint arXiv:2507.22675* **2025**. https://doi.org/10.48550/arXiv.2507.22675.
38. Dong, S.; Hu, Y.; Wang, L.; Chen, G.; Meng, X. PeftCD: Leveraging Vision Foundation Models with Parameter-Efficient Fine-Tuning for Remote Sensing Change Detection. *arXiv preprint arXiv:2509.09572* **2025**. https://doi.org/10.48550/arXiv.2509.09572.
39. Ji, S.; Wei, S.; Lu, M. Fully Convolutional Networks for Multisource Building Extraction from an Open Aerial and Satellite Imagery Data Set. *IEEE Transactions on Geoscience and Remote Sensing* **2019**, *57*, 574–586. https://doi.org/10.1109/TGRS.2018.2858817.
40. Noman, M.; Fiaz, M.; Cholakkal, H.; Khan, S.; Khan, F.S. ELGC-Net: Efficient Local-Global Context Aggregation for Remote Sensing Change Detection. *IEEE Transactions on Geoscience and Remote Sensing* **2024**, *62*. https://doi.org/10.1109/TGRS.2024.3362914.
41. Tao, C.; Kuang, D.; Huang, Z.; Peng, C.; Li, H. HASNet: A Foreground Association-Driven Siamese Network with Hard Sample Optimization for Remote Sensing Image Change Detection. *IEEE Transactions on Geoscience and Remote Sensing* **2025**, *63*. https://doi.org/10.1109/TGRS.2025.3545760.
42. Noman, M.; Fiaz, M.; Cholakkal, H. ChangeBind: A Hybrid Change Encoder for Remote Sensing Change Detection. 2024, pp. 8417–8422.
43. Jia, J.; Lee, G.; Wang, Z.; Lyu, Z.; He, Y. Siamese Meets Diffusion Network: SMDNet for Enhanced Change Detection in High-Resolution RS Imagery. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **2024**. https://doi.org/10.1109/JSTARS.2024.3384545.
44. Yang, J.; Wan, H.; Shang, Z. EHCTNet: Enhanced Hybrid of CNN and Transformer Network for Remote Sensing Image Change Detection. *arXiv preprint arXiv:2501.01238* **2025**. https://doi.org/10.48550/arXiv.2501.01238.
45. Shi, Y.; Yang, R.; Yin, C.; Lu, Y.; Huang, B.; Tao, Y.; Zhong, Y. Two-Stage Fine-Tuning of Large Vision-Language Models with Hierarchical Prompting for Few-Shot Object Detection in Remote Sensing Images. *Remote Sensing* **2026**, *18*. https://doi.org/10.3390/rs18020266.