2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

TERDNet: Transformer Encoder-Recurrent Decoder Network for Scene Change Detection

Jiae Yoon and Ue-Hwan Kim*

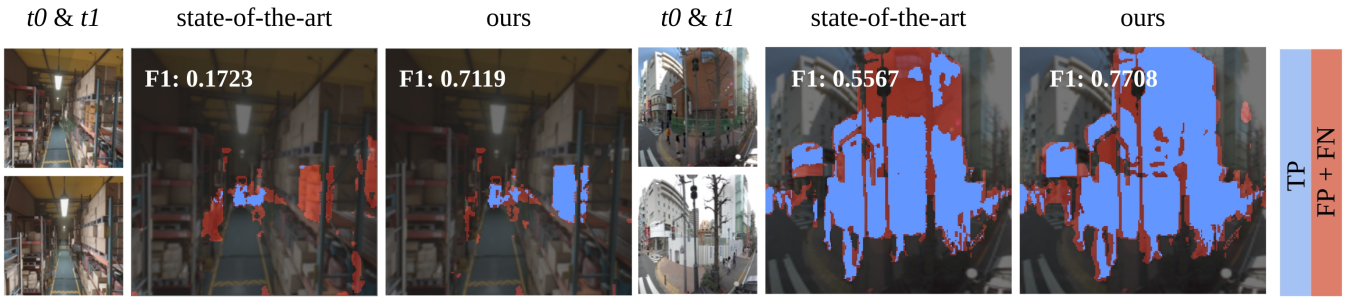


Fig. 1. **Comparative results of the current state-of-the-art C-3PO [1] and our TERDNet on four benchmark datasets.** TERDNet achieves superior quantitative performance and produces more precise change masks with clearer boundaries compared to the existing state-of-the-art approach.

***Abstract*— In this work, we address the challenge of Scene Change Detection (SCD), where the goal is to identify variations between two images of the same location captured at different times. Existing SCD models often overlook the varying importance of features across layers, employ single-step decoders that confine refinement, and provide limited insight into encoder pretraining strategies. We propose TERDNet, a Transformer Encoder–Recurrent Decoder Network designed to overcome these limitations. TERDNet consists of a transformer-based encoder that extracts multi-level representations, a feature fusion module that integrates correlation volumes with these features, a recurrent 3-gate-GRU decoder that performs iterative refinement, and a combined convolution–interpolation upsampler that restores fine-grained resolution. Extensive experiments on four public benchmarks show that TERDNet consistently outperforms prior approaches and produces more accurate and detailed change masks. Ablation studies confirm the benefit of segmentation-based pretraining and the effectiveness of our fusion design. In addition, robustness tests under viewpoint misalignment confirm TERDNet's potential for deployment in real-world robotic systems, where reliable perception is critical. Our code is at** `https://github.com/AutoCompSysLab/TERDNet`.

I. Introduction

Scene Change Detection (SCD) generates a change mask indicating areas of variation from two images captured at different times but depicting the same location [2]. In robotics, SCD is particularly valuable as it enables mobile agents to reason about dynamic environments [3], identify object-level changes in real-world deployments, and adapt navigation strategies accordingly [4]. Further, autonomous robots and vehicles operating in urban or indoor environments must continuously assess environmental shifts [5], such as moved furniture or newly placed obstacles, to make informed decisions and ensure safe traversal [6]. SCD is similar to the segmentation task [7] in that it discriminates the class of each pixel. The two tasks share the commonality of requiring sophisticated edge extraction, but they also differ in that SCD involves inputting two images; in comparing two images with a time difference, SCD resembles optical flow [8]. As a result, most SCD models adopt an encoder-decoder architecture similar to segmentation and optical flow models.

Conventional SCD studies leveraging deep neural networks [1], [9] have predominantly employed multi-level feature maps derived through feature pyramids [10]. This approach does not explicitly account for the varying importance of the feature maps extracted from each layer. With the emergence of transformer-based models, recent SCD works [11], [12], [4], [13] have begun adopting transformer encoders. However, prior studies have not systematically analyzed which pretraining strategies are most suitable for SCD. On the decoder side, the prevailing architecture within SCD models typically employs single-update decoders. While this approach is computationally efficient, it can limit the models' ability to generate detailed and accurate change masks. Single-update decoders are constrained in their capacity to refine and enhance the output through iterative processes, which can significantly detract from the overall effectiveness and accuracy of SCD models.

To overcome these limitations of SCD models, we propose the Transformer Encoder and Recurrent Decoder Network (TERDNet). TERDNet incorporates a feature fusion module that integrates correlation volumes with multi-level transformer features. This module also learns the relative importance of each layer's feature map, leading to better fusion of features across layers. In addition, we conduct experiments on the encoder, including comparisons between CNN and transformer backbones as well as different pretraining strategies, to analyze their impact on SCD.

All authors are with the Department of AI Convergence, Gwangju Institute of Science and Technology (GIST), Gwangju 61005, Republic of Korea. jiaeyoon@gm.gist.ac.kr
*Corresponding author: Ue-Hwan Kim (uehwan@gist.ac.kr).

Further, we devise a recurrent structure for the decoder to perform iterative refinement, which leads to performance enhancement. While prior optical flow models [14], [15] have demonstrated the benefit of convGRU for iterative refinement, these approaches have not been designed for SCD. In contrast, our proposed 3-gate-GRU explicitly incorporates the feature pyramid and dynamically weights layer importance, enabling recurrent refinement tailored specifically to SCD. The 3-gate-GRU receives inputs from the current input, past output, and the feature pyramid with applied weights. As illustrated in Fig. 1, this design addresses the limitations of existing SCD decoders by enabling iterative updates that refine the output progressively.

In summary, the main contributions of our work are as follows:

- **Encoder and Feature Fusion Module**: We leverage transformer encoders to extract multi-level features and introduce a fusion module that integrates correlation volumes with these features, while explicitly learning the relative importance of each layer's representation.
- **Recurrent Structure in Decoder**: We propose a 3-gate-GRU tailored for SCD, which enables iterative refinement and dynamic integration of features from the transformer backbone.
- **SoTA Performance**: Our model achieves state-of-the-art results on multiple public benchmarks.

II. Related Work

Change Detection. Change Detection (CD) aims to identify differences in the state of an object or phenomenon by comparing data collected at different times [16]. With the advent of deep learning, significant advancements have been made by leveraging the capabilities of CNNs and RNNs. For example, ChangeNet [17] has employed a fully convolutional network to learn and detect changes directly from raw image pairs. ConvLSTM [18] has captured temporal dependencies, especially beneficial in video surveillance. L-UNet [19] has enhanced CD performance by applying an Atrous convolution structure to convLSTM, thereby leveraging both temporal and spatial information. Huang et al. [20] have enhanced weakly supervised learning by mixing background information to create more robust training samples.

Scene Change Detection. Scene Change Detection (SCD) is a task that identifies areas of change at the pixel level in images captured at different times, $t0$ and $t1$, of the same location. While CD encompasses various domains such as remote sensing [21], SCD focuses specifically on changes in visual scenes. In contrast to CD, where images often align well due to controlled or satellite-based capture, SCD must additionally contend with imperfect alignment, variations in lighting, and occlusions. These factors complicate direct pixel-level comparisons and make the task inherently more challenging.

Several works have led to the advancement of SCD. FC-Siam [22] has applied a siamese network to SCD to process each image pair independently; the model, composed of a fully convolutional network (FCN) [23], has combined high-resolution features extracted from early layers with abstract features from deeper layers, inspired by U-Net [24]. CSCDNet[25] has proposed that utilizes CNN encoder and correlation layers. DR-TANet [26] has introduced a dynamic receptive temporal attention module inspired by self-attention to represent the relationship between two feature maps. C-3PO [1] has replaced the decoder of the SCD model with a segmentation model decoder from DeepLab [27] or FCN [23]. Conventional SCD models have predominantly employed CNN-based architectures. These architectures generally provide weaker representational capability than transformers. They also generate feature maps of varying resolutions, which forces uniform input into the decoder regardless of layer importance.

Recently, several SCD models have adopted transformer-based architectures. RobustSCD [12] utilizes the robust feature extraction capabilities of DINOv2 [28]. Meanwhile, ZSSCD [11], ZeroSCD [4], and GeSCD [13] leverage the Segment Anything Model (SAM) [7], a segmentation foundation model, to enable zero-shot SCD. Although models leveraging foundation models can exploit richer representational power, prior studies do not provide an analysis of which pretrained weights of such models are most appropriate for SCD. Furthermore, the simplicity of existing SCD decoders reduces computational costs but at the same time limits their ability to produce detailed change masks. This limitation indicates a significant opportunity for further advancements, particularly in the design of foundation model–based encoders and recurrent decoder structures.

Foundation Models. Foundation models refer to models trained on large datasets that can be applied to a variety of application tasks, with transformer-based foundation models recently gaining prominence [29]. Following the introduction of transformer models based on attention modules in natural language processing (NLP) [30], vision transformer models have emerged for vision tasks [31]. These models offer higher representation power than CNNs but demand large datasets due to their weaker inductive bias [32]. Vision foundation models utilizing the transformer structure have achieved high performance in various vision tasks such as classification [33], object detection [34], and segmentation [7], and researchers actively have utilized these foundation models.

Recurrent Decoders. Applying Recurrent Neural Networks (RNNs) to decoders can improve outputs through iterative updates. Especially, convLSTM [18] and convGRU [35] integrate convolution operations with RNNs to effectively learn spatial features within data and enhance computational efficiency, making them well-suited for vision tasks. Due to these advantages, various researchers have utilized convLSTM and convGRU as recurrent decoders for different tasks. PredRNN [36] and PredRNN++ [37] have utilized convLSTM to predict video frames. RAFT [14] has applied convGRU to an optical flow estimation model, and subsequently, many models have utilized an iterative estimator in the decoder [15]. Inspired by these applications, we uniquely reconfigure convGRU to suit SCD and incorporate convGRU

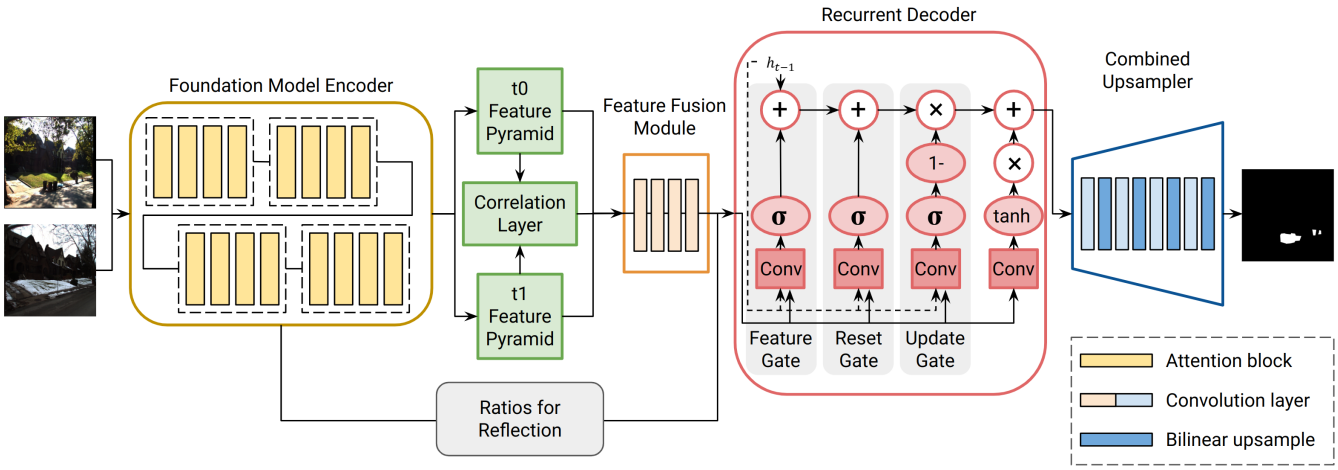


Fig. 2. **The architecture of the proposed TERDNet.** The foundation model-based backbone encoder extracts the feature pyramid from the two images. The decoder performs recurrent updates with the proposed GRU, and Ratios for Reflection computes the gating map f_t in Eq. (2) from pyramid differences. The Feature Fusion Module combines the two feature maps and the correlation volume, feeding the combined feature into the decoder. The decoder uses three gates of the proposed GRU to determine the contribution weight of each source of information and performs recurrent updates. The combined upsampler alternates between convolution layers and bilinear interpolation and restores the output change mask to the original resolution.

into the decoder of our model.

III. Methodology

A. Foundation Model-based Encoder

Fig. 2 illustrates the architecture of TERDNet. The encoder extracts features from the intermediate layers of the model. This involves dividing the attention layers into four sections and extracting features from the last layer of each section. For instance, in a ViT-b model with 12 attention layers, features are extracted from layers 2, 5, 8, and 11. Each selected block corresponds to one quarter of the encoder depth to provide representations from low- to high-level abstraction with minimal redundancy. In contrast to CNN-based feature pyramids, where feature maps vary in resolution, transformer-based encoders produce features of uniform size. This uniformity enhances the consistency of feature representation. By preserving a consistent feature size, the encoder also enables seamless integration with subsequent modules and ensures that the extracted features retain rich and detailed information across all scales.

B. Feature Fusion Module

A critical aspect of SCD is understanding both the individual characteristics of each input image and the relationship between the pair. To achieve this, TERDNet employs a feature fusion module that combines correlation volumes with the feature maps of each image.

For relationship information, we employ the correlation volume from the optical flow research [38]. Given feature maps extracted from images at times $t0$ and $t1$, denoted as m_0 and m_1 respectively, for a pixel (x_1, x_2) in m_0, the correlation with $(2r+1)^2$ neighboring pixels around (x_1, x_2) in m_1 is computed through dot products, representing the relationship between the two feature maps as follows:

$$c(x_1, x_2) = \sum_{o \in [-r,r] \times [-r,r]} \langle m_0(x_1 + o), m_1(x_2 + o) \rangle, \quad (1)$$

where $\langle \cdot, \cdot \rangle$ and r denote the dot product and correlation range, respectively. While optical flow models require wide-range correlation calculations to detect fast and extensive object movements, SCD focuses on detecting the appearance, disappearance, or change of objects [39], [40]. In this case, a narrow-range correlation is sufficient. Additionally, the correlation layer has a high computational demand, but limiting the search range enhances computational efficiency.

Next, to utilize the information from both images, we pass the feature map of each image through a convolution layer to reduce the number of channels. This reduction process ensures that the subsequent fusion steps are computationally efficient while retaining essential information. The reduced feature map is then fused with the correlation volume. We repeat this process for each layer of the feature pyramid, and the concatenated feature maps from all layers yield the final fused feature. This module preserves the distinct information present in each image as well as highlights the interactions between them, providing a comprehensive basis for detecting changes.

C. Recurrent Decoder

The advantage of the recurrent structure lies in its ability to iteratively update outputs, and it can reintegrate lost information at each iteration. Inspired by previous research [14] that applied the separable convGRU to the decoder in image comparison tasks, we reconfigure GRU for TERDNet and incorporate it into the decoder.

We propose a 3-gate-GRU that has three gates for SCD. We integrate the feature pyramid into the GRU, allowing the decoder to update outputs using information from all

feature map layers. Our 3-gate-GRU adds a feature gate p_t to the reset and update gates of the traditional GRU, which allows the model to incorporate input from the feature pyramid. The inclusion of the feature gate p_t enables the model to weigh the importance of features from different layers dynamically. This capability is particularly valuable for SCD, as it allows the model to focus on the most relevant features. The Ratios for Reflection block computes a sigmoid-gated map f_t by applying a learnable projection P to the concatenated pyramid differences $(m_0^i - m_1^i)$, and f_t serves as the pyramid-derived gating signal for the feature gate p_t. Unlike hierarchical decoders using CNN backbones, which sequentially input features from different layers, our structure allows for prioritizing information of more critical layers from the feature pyramid. p_t decides the extent to which f_t influences the candidate state $\tilde{h_t}$, subsequently calculating the current state h_t by considering the importance of the previous state h_{t-1}, current input x_t, and f_t as follows:

$$f_t = \sigma\left(P * \text{Concat}_i(m_0^i - m_1^i)\right), \quad (2)$$

$$r_t = \sigma(W_r * x_t + U_r * h_{t-1} + F_r * f_t), \quad (3)$$

$$z_t = \sigma(W_z * x_t + U_z * h_{t-1} + F_z * f_t), \quad (4)$$

$$p_t = \sigma(W_p * x_t + U_p * h_{t-1} + F_p * f_t), \quad (5)$$

$$\tilde{h_t} = \tanh(W * x_t + U * (r_t \odot h_{t-1}) + F * (p_t \odot f_t)), \quad (6)$$

$$h_t = (1 - z_t)h_{t-1} + z_t\tilde{h_t}, \quad (7)$$

where m_0^i denotes the ith layer of the feature pyramid m_0, $\text{Concat}_i(\cdot)$ denotes channel-wise concatenation across pyramid levels, and P is a learnable convolutional projection.

D. Combined Upsampler

Given the reduced resolution resulting from the transformer backbone encoder, TERDNet employs a combined upsampler to restore the original image resolution. Unlike traditional upsampling methods, which may struggle with the uniform output size of transformer-based features, our combined upsampler utilizes a sequence of convolution and bilinear interpolation steps. Our combined upsampler stacks blocks of 3×3 convolution layers and bilinear interpolation to recover resolution. The output of the GRU decoder, initially $\frac{H}{16} \times \frac{W}{16} \times 512$, is upscaled to $H \times W \times 2$ resolution using four blocks, with convolution reducing channel numbers to 256, 128, 64, and finally 2. Each bilinear interpolation follows the convolution layers and gradually increases the resolution rather than expanding sixteenfold at once. This design improves upsampling efficiency and preserves the quality of the output change mask. This approach addresses the challenges posed by the uniform output size of transformer-based features and helps produce accurate and detailed change masks.

E. Training Loss

We train TERDNet with a sequential weighted cross-entropy loss, tailored to suit the recurrent decoder and SCD datasets. The integration of sequential and weighted cross-entropy losses ensures that the model is trained to handle the unique challenges of SCD. The sequential loss calculates the loss for all predictions, and later iterations are weighted more heavily in the total loss through the utilization of the weight factor γ ($\gamma = 0.8$ in all experiments). On the other hand, pixels with changes tend to be less frequent than those without changes in SCD. To mitigate this class imbalance, we integrate the sequential loss for the 3-gate-GRU decoder with the weighted cross-entropy loss for SCD; the training loss for our network becomes as follows:

$$L = \sum_{k=1}^{M} \gamma^{M-k} \sum_{i=0}^{N} w_i y_i \log p_i^k, \quad (8)$$

$$w_i = \frac{\sum_{j=0}^{N} n_j - n_i}{N \sum_{j=0}^{N} n_j}, \quad (9)$$

where p_i^k and y_i represent the predicted probability of kth iteration and ground truth for the two classes, respectively, w_i is the balance weight, and since the SCD task only distinguishes between the presence and absence of change, $N = 1$. This comprehensive approach for the loss function configuration is critical for achieving high performance in SCD tasks, where the precision of change detection is paramount.

TABLE I

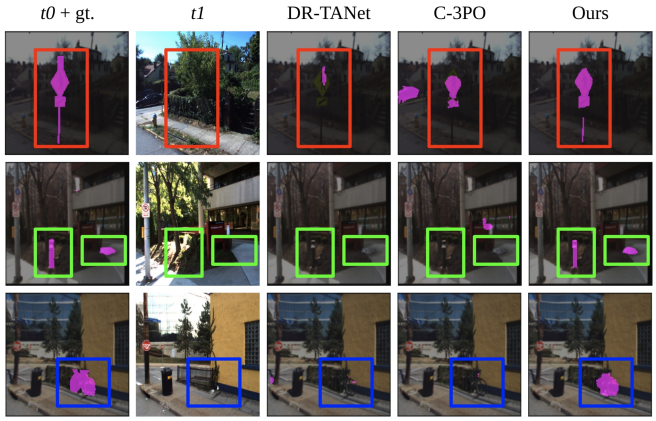
COMPARISON STUDY RESULTS. (A) F1-SCORES (%), (B) MIOU (%) OF PREVIOUS METHODS AND THE PROPOSED MODEL. S AND C REPRESENT THE IOU FOR THE STATIC AND CHANGE CLASSES, RESPECTIVELY.

(a) F1-scores on VL-CMU-CD and TSUNAMI

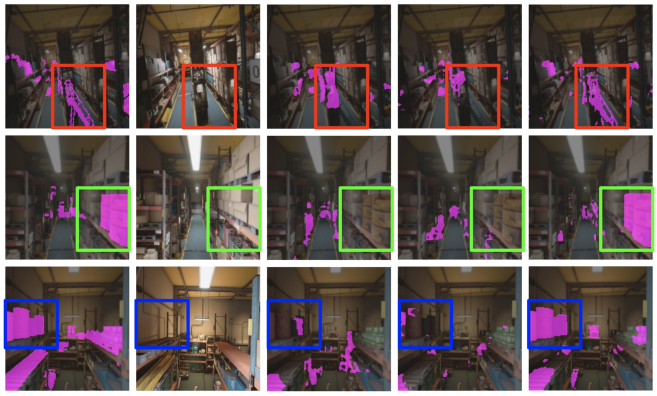
Method	VL-CMU-CD	TSUNAMI
FC-EF	44.6	77.7
FC-Siam-diff	65.3	79.5
FC-Siam-conc	65.6	81.6
CSCDNet	76.6	84.8
DR-TANet	75.1	84.5
C-3PO	<u>80.2</u>	86.5
ZSSCD	51.6	56.5
RobustSCD	79.5	<u>88.1</u>
GeSCD	75.4	72.8
Ours	**83.4**	**89.5**

(b) mIoU on PSCD and ChangeSim

Method	PSCD			ChangeSim		
	S	C	mIoU	S	C	mIoU
FC-EF	83.7	48.8	66.3	70.2	20.3	45.2
FC-Siam-diff	88.9	55.7	72.3	80.0	25.9	53.0
FC-Siam-conc	86.3	57.2	71.8	80.1	26.1	53.1
CSCDNet	88.8	61.7	75.3	87.3	22.9	55.1
DR-TANet	89.4	60.3	74.9	89.0	24.3	56.7
C-3PO	<u>90.8</u>	<u>67.2</u>	<u>79.0</u>	<u>90.4</u>	28.8	59.6
ZSSCD	10.8	18.9	14.8	89.8	1.4	45.6
RobustSCD	64.5	47.8	56.2	71.8	22.9	47.4
GeSCD	85.1	39.2	62.1	89.8	<u>33.2</u>	<u>61.5</u>
Ours	**91.5**	**69.6**	**80.5**	**91.2**	**38.3**	**64.7**



(a) Comparison study results on VL-CMU-CD



(b) Comparison study results on ChangeSim

Fig. 3. **Qualitative results of Comparative study on VL-CMU-CD, and ChangeSim. Red** boxes highlight thin or low-contrast structures, **green** boxes precise localization of changes, and **blue** boxes region completion. Predicted masks from prior methods [26], [1] and TERDNet visually indicate cleaner boundaries and more complete regions.

IV. Experiments

A. Settings

1) Datasets

For a comparative study between conventional SCD models and TERDNet, we utilize four datasets: VL-CMU-CD [40], TSUNAMI [39], PSCD [25], and ChangeSim [41]. We select datasets that are frequently used as benchmarks in SCD studies and cover diverse environments such as indoor and outdoor scenes. We train and evaluate a separate model for each benchmark using its official split. We follow each dataset's official train/test split and use the dataset-provided input order $(t0, t1)$, since the ground-truth change masks are defined with respect to the image at $t1$.

2) Baselines

To ensure fairness and reproducibility, we compare TERDNet against 9 previous SCD models with publicly released code, including FC-EF [22], FC-Siam-diff [22], FC-Siam-conc [22], CSCDNet [25], DR-TANet [26], C-3PO [1], ZSSCD [11], RobustSCD [12], and GeSCD [13].

B. Comparative Study

1) VL-CMU-CD and TSUNAMI

We evaluated TERDNet using the F1-score metric on the VL-CMU-CD and TSUNAMI datasets. As shown in Table I(a), TERDNet consistently outperformed prior approaches on both benchmarks. These improvements indicate that the proposed feature fusion and recurrent refinement allow the model to capture both everyday scene variations in VL-CMU-CD and the more abrupt changes present in TSUNAMI.

2) PSCD and ChangeSim

Since prior studies [25], [1] on PSCD and ChangeSim reported results in terms of mIoU, we adopt the same metric for comparison. As shown in Table I(b), TERDNet achieved a higher mIoU than all previous approaches on PSCD. This result confirms its ability to handle real-world outdoor imagery with diverse change scenarios. On ChangeSim, a synthetic indoor benchmark, TERDNet also surpassed the baselines. Overall, these results suggest that the proposed design remains effective across different domains, such as real-world outdoor and synthetic indoor environments.

3) Qualitative Comparison

Fig. 3 depicts qualitative comparison results. The qualitative results show that TERDNet can extract more sophisticated change masks. The conventional models often fail to generate complex masks (red boxes), accurately identify the locations of changes (green boxes), or adequately fill in the regions (blue boxes). This is particularly observable in scenes with intricate changes or minimal object displacement, which is challenging for these models to discern accurately. In contrast, TERDNet is robust even in detecting subtle changes where colors are similar or objects are thin and hard to capture (red and green boxes), and it comprehensively covers the changed areas (blue boxes). In summary, TERDNet is more resilient to the SCD challenges, such as discrepancies in viewpoint, illumination, and object positioning between the $t0$ and $t1$ images.

C. Ablation Study

Through a series of experiments, we explore the effects of various pretraining and finetuning methods, the integration of a feature fusion module, the implementation of a recurrent structure, and the number of iterations in the decoder. In our ablation study, we employ the VL-CMU-CD dataset to examine the performance of each component of TERDNet.

1) Backbone Encoder

Table II(a) compares the performance across different encoders. With a ResNet backbone, TERDNet already surpasses prior SCD models and confirms that the overall architecture is effective even without a foundation model encoder. Transformer backbones further improve performance, although larger variants give diminishing returns relative to their parameter count. Based on this trade-off, the ViT-b variant provides a practical balance between accuracy and efficiency.

TABLE II

ABLATION STUDY RESULTS. (A) THE EFFECT OF THE BACKBONE ENCODER, (B) THE EFFECT OF THE PRETRAINING METHOD, (C) THE EFFECT OF THE FINETUNING METHOD, (D) THE EFFECT OF USED FEATURE TYPES, (E) THE EFFECT OF THE DECODER STRUCTURE, AND (F) THE EFFECT OF THE NUMBER OF ITERATIONS.

(a) The backbone encoder

Architecture	**Model**	**Parameters**	F1-score (%)
CNN	ResNet18	11M	79.6
	ResNet50	26M	80.9
Transformer	ViT-b	86M	83.4
	ViT-l	307M	83.7
	ViT-h	632M	**84.0**

(b) The pretraining method

Method	**Dataset**	Resolution	F1-score (%)
Supervised	ImageNet	384×384	60.0
	SA-1B	1024×1024	**83.4**
MAE	ImageNet	224×224	57.2
	COCO	224×224	56.0
DINOv2	ImageNet	784×784	74.3
	COCO	784×784	74.3

(c) The finetuning method

Method	F1-score (%)
LLRD	77.3
LoRA	82.9
VPT	82.3
Frozen	**83.4**

(d) The used feature types

Method	F1-score (%)
Feature Maps	82.3
Local Corr	81.2
Global Corr	81.3
Ours	**83.4**

(e) The decoder structure

Method	F1-score (%)
Without GRU	79.5
With basic GRU	81.9
Ours	**83.4**

(f) The number of iterations

Iterations	F1-score (%)
3	83.0
5	**83.4**
7	83.2
10	82.9

2) Pretraining Methods

To investigate the effect of pretraining methods, we utilized supervised learning [7], MAE [42], and DINOv2 [28], each trained on classification and segmentation tasks. In Table II(b), different input resolutions were also tested, and models pretrained with sizes below 512×512 showed degraded performance. Increasing the resolution to 784×784 improved results, but models pretrained on the large-scale segmentation dataset SA-1B [7] achieved the best scores. Since dataset scale, supervision type, and input resolution are inherently coupled in publicly available checkpoints, this study highlights consistent trends across representative pretraining strategies rather than isolating a single factor.

3) Finetuning Methods

Table II(c) presents the performance results of various finetuning methods applied to the encoder. The best results were obtained with the frozen foundation model. This shows that the pretrained features already contain information essential for SCD. In contrast, full-parameter finetuning often caused overfitting and reduced performance. This issue was most evident with Layer-wise Learning Rate Decay (LLRD) [43], which produced the lowest accuracy among the tested methods. Parameter-efficient approaches such as Low-Rank Adaptation (LoRA) [44] and Visual Prompt Tuning (VPT) [45] gave results close to the frozen baseline. Overall, in our experiments, keeping the encoder frozen was the most stable strategy for TERDNet.

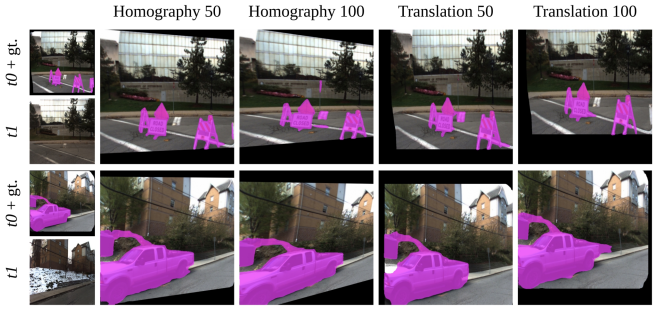


Fig. 4. **Qualitative robustness evaluation under misalignment.** The $t0$ image is perturbed using homography or translation with magnitudes of 50 and 100 pixels. TERDNet generates consistent change masks without any task-specific finetuning, even when geometric distortions are introduced.

4) Feature Fusion Module

Table II(d) shows the effect of the use of the feature map and correlation volume. The evaluation includes both local correlation volume, which calculates the correlation with nearby pixels, and global correlation volume, which calculates the correlation with all pixels. Our method, which utilized both feature maps and correlation volume, achieved the highest performance. These findings indicate that both the features of each image and the relationships between the two images are crucial in the SCD task, and TERDNet effectively leverages these aspects through its feature fusion module. Notably, there was little difference in performance between local and global correlation volumes, suggesting that in SCD, the relationship with distant pixels does not significantly impact the detection of changes since the task focuses solely on identifying the presence or absence of change.

5) Recurrent Decoder

Table II(e) compares different decoder structures. Adding a standard GRU improved performance compared to using no recurrent structure, showing that recurrence helps refine the predictions. The 3-gate-GRU achieved even higher accuracy than the standard GRU. Unlike the standard GRU, which uses only reset and update gates, the 3-gate-GRU introduces an additional feature gate that controls how the fused features and the feature pyramid contribute to the update. This design produces more detailed change masks.

6) Number of Iterations

We examined the effect of the number of iterations in the 3-gate-GRU decoder. Table II(f) shows that performance improved when the iteration count increased from 3 to 5. Beyond 5 iterations, however, the F1-score gradually declined, which suggests that the decoder began to overfit the training data. In our experiments, five iterations provided the best balance between accuracy and stability.

TABLE III
COMPARISON OF SCD MODELS.

Method	Backbone Name	Backbone	Decoder	Total	GFLOPs
FC-Siam-conc	U-Net	0.4M	0.6M	1M	43
FC-Siam-diff	U-Net	0.4M	0.6M	1M	38
CSCDNet	ResNet18	11M	83M	94M	337
DR-TANet	ResNet18	11M	22M	33M	54
C-3PO	ResNet18	11M	30M	41M	490
C-3PO	ResNet50	26M	173M	199M	2715
C-3PO	VGG16	138M	41M	179M	951
RobustSCD	ViT-s	22M	6M	28M	244
ZSSCD	ViT-h	632M	74M	706M	32928
GeSCD	ViT-h	632M	9M	641M	56941
Ours (iters=3)	ViT-b	86M	16M	102M	1806
Ours (iters=5)	ViT-b	86M	16M	102M	2004

D. Robustness to Misalignment

We assess the robustness of TERDNet to misalignment by perturbing the $t0$ image through translation and homography, while keeping $t1$ unchanged. Fig. 4 presents qualitative results under perturbations of 50 and 100 pixels. Without additional finetuning, TERDNet generates consistent change masks and avoids false positives in static areas. These results suggest that the model can maintain robustness even under strong geometric distortions that are likely to appear in real deployments.

E. Efficiency Analysis of SCD Models

1) Model Parameters

Table III compares the number of parameters across representative SCD models. U-Net [22] and ResNet18-based models [25], [26], [1] have relatively small parameter counts, while VGG16-based approaches [46], [47], [1] exceed 170M parameters. Recent transformer-based methods such as ZSSCD [11] and GeSCD [13] employ remarkably large ViT-h backbones, with over 600M parameters. TERDNet, in contrast, integrates a ViT-b backbone (86M) with a recurrent decoder (16M), totaling 102M parameters. This places it at a scale comparable to mid-sized ResNet-based models rather than the largest transformer-based approaches. The decoder remains compact while enabling iterative refinement through the 3-gate-GRU. Overall, TERDNet maintains a relatively modest parameter size compared to large ViT-h models, while achieving strong accuracy in the benchmark results reported in Section IV-B.

2) Computational Complexity

Table III also reports the computational complexity measured in GFLOPs. U-Net models require very few computations but typically underperform compared to larger networks. VGG16-based models [46], [47], [1] require close to one thousand GFLOPs, while ResNet50-based C-3PO [1] exceeds two thousand GFLOPs. In contrast, transformer models with ViT-h backbones, such as ZSSCD [11] and GeSCD [13], require tens of thousands of GFLOPs and are therefore more computationally expensive. TERDNet requires 1806 GFLOPs with three iterations and 2004 GFLOPs with five iterations. While this is higher than U-Net or ResNet18-based models, it remains far below ViT-h approaches and is on the same order as ResNet50-based C-3PO. When considering both efficiency (Table III) and accuracy (Tables I(a), I(b)), TERDNet offers a practical balance between computational cost and performance.

V. CONCLUSION

In this work, we presented TERDNet, a Transformer Encoder–Recurrent Decoder Network for scene change detection. TERDNet introduces a feature fusion module that combines correlation volumes with multi-level transformer features and emphasizes layer-wise importance, and a 3-gate-GRU decoder that iteratively refines predictions by integrating information across layers. Together with a combined upsampler, these components address the main limitations of existing SCD models, such as the lack of feature weighting and the reliance on single-step decoding. Through extensive experiments on four public benchmarks, TERDNet consistently outperformed prior approaches. The results show clear benefits from integrating transformer encoders with feature fusion and recurrent decoding, and the model achieves state-of-the-art performance. Beyond accuracy, evaluations on robustness demonstrated that TERDNet maintains stable performance under misalignment, which is essential for deployment in practical robotic and vision systems. Overall, this study highlights how carefully designed encoder fusion and recurrent decoding mechanisms can improve scene change detection. We believe the insights gained here can support future research on long-term perception in robotics and related vision tasks.

ACKNOWLEDGMENT

This research was partly supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. NRF-2022R1C1C1009989); by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No. RS-2022-II220926, Development of Self-directed Visual Intelligence Technology Based on Problem Hypothesis and Self-supervised Methods); by the National Research Council of Science & Technology(NST) grant by the Korea government(MSIT) (No. GTL25041-000); and by the Korea Agency for Infrastructure Technology Advancement(KAIA) grant funded by the Ministry of Land Infrastructure and Transport (Grant RS-2023-00256888).

REFERENCES

[1] G.-H. Wang, B.-B. Gao, and C. Wang, "How to reduce change detection to semantic segmentation," *Pattern Recognition*, vol. 138, p. 109384, 2023.

[2] R. J. Radke, S. Andra, O. Al-Kofahi, and B. Roysam, "Image change detection algorithms: a systematic survey," *IEEE Transactions on Image Processing*, vol. 14, pp. 294–307, 2005.

[3] Z. J. Yew and G. H. Lee, "City-scale scene change detection using point clouds," in *IEEE International Conference on Robotics and Automation*, 2021, pp. 13 362–13 369.

[4] S. S. Kannan and B.-C. Min, "Zeroscd: Zero-shot street scene change detection," in *IEEE International Conference on Robotics and Automation*, 2025, pp. 4665–4671.

[5] J. Rowell, L. Zhang, and M. Fallon, "Lista: Geometric object-based change detection in cluttered environments," in *IEEE International Conference on Robotics and Automation*, 2024, pp. 3632–3638.

[6] S. Looper, J. Rodriguez-Puigvert, R. Siegwart, C. Cadena, and L. Schmid, "3d vsg: Long-term semantic scene change prediction through 3d variable scene graphs," in *IEEE International Conference on Robotics and Automation*, 2023, pp. 8179–8186.

[7] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, *et al.*, "Segment anything," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 4015–4026.

[8] Y. Wu, F. Paredes-Vallés, and G. C. De Croon, "Lightweight event-based optical flow estimation via iterative deblurring," in *IEEE International Conference on Robotics and Automation*, 2024, pp. 14 708–14 715.

[9] Y. Gan, W. Xuan, H. Chen, J. Liu, and B. Du, "Rfl-cdnet: Towards accurate change detection via richer feature learning," *Pattern Recognition*, vol. 153, p. 110515, 2024.

[10] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2117–2125.

[11] K. Cho, D. Y. Kim, and E. Kim, "Zero-shot scene change detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 3, 2025, pp. 2509–2517.

[12] C.-J. Lin, S. Garg, T.-J. Chin, and F. Dayoub, "Robust scene change detection using visual foundation models and cross-attention mechanisms," in *IEEE International Conference on Robotics and Automation*, 2025, pp. 8337–8343.

[13] J.-W. Kim and U.-H. Kim, "Towards generalizable scene change detection," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 24 463–24 473.

[14] Z. Teed and J. Deng, "Raft: Recurrent all-pairs field transforms for optical flow," in *European Conference on Computer Vision*, 2020, pp. 402–419.

[15] S. Zhou, X. Jiang, W. Tan, R. He, and B. Yan, "Mvflow: Deep optical flow estimation of compressed videos with motion vector prior," in *Proceedings of the ACM International Conference on Multimedia*, 2023, pp. 1964–1974.

[16] Y. Sun, L. Lei, X. Li, H. Sun, and G. Kuang, "Nonlocal patch similarity based heterogeneous remote sensing change detection," *Pattern Recognition*, vol. 109, p. 107598, 2021.

[17] R. C. Daudt, B. Le Saux, A. Boulch, and Y. Gousseau, "Urban change detection for multispectral earth observation using convolutional neural networks," in *IEEE International Geoscience and Remote Sensing Symposium*, 2018, pp. 2115–2118.

[18] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo, "Convolutional lstm network: A machine learning approach for precipitation nowcasting," *Advances in Neural Information Processing Systems*, vol. 28, 2015.

[19] S. Sun, L. Mu, L. Wang, and P. Liu, "L-unet: An lstm network for remote sensing image change detection," *IEEE Geoscience and Remote Sensing Letters*, vol. 19, pp. 1–5, 2020.

[20] R. Huang, R. Wang, Q. Guo, J. Wei, Y. Zhang, W. Fan, and Y. Liu, "Background-mixed augmentation for weakly supervised change detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2023, pp. 7919–7927.

[21] M. Gong, J. Zhao, J. Liu, Q. Miao, and L. Jiao, "Change detection in synthetic aperture radar images based on deep neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, pp. 125–138, 2015.

[22] R. C. Daudt, B. Le Saux, and A. Boulch, "Fully convolutional siamese networks for change detection," in *IEEE International Conference on Image Processing*, 2018, pp. 4063–4067.

[23] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[24] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention: International Conference*, 2015, pp. 234–241.

[25] K. Sakurada, M. Shibuya, and W. Wang, "Weakly supervised silhouette-based semantic scene change detection," in *IEEE International Conference on Robotics and Automation*, 2020, pp. 6861–6867.

[26] S. Chen, K. Yang, and R. Stiefelhagen, "Dr-tanet: Dynamic receptive temporal attention network for street scene change detection," in *IEEE Intelligent Vehicles Symposium*, 2021, pp. 502–509.

[27] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, pp. 834–848, 2017.

[28] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, *et al.*, "Dinov2: Learning robust visual features without supervision," *arXiv preprint arXiv:2304.07193*, 2023.

[29] J. D. M.-W. C. Kenton and L. K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of Human Language Technology: Conference of the North American Chapter of the Association of Computational Linguistics*, 2019, pp. 4171–4186.

[30] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[31] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations*, 2021.

[32] L. H. Mormille, C. Broni-Bediako, and M. Atsumi, "Introducing inductive bias on vision transformers through gram matrix similarity based regularization," *Artificial Life and Robotics*, vol. 28, pp. 106–116, 2023.

[33] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 10 012–10 022.

[34] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *European Conference on Computer Vision*, 2020, pp. 213–229.

[35] N. Ballas, L. Yao, C. Pal, and A. Courville, "Delving deeper into convolutional networks for learning video representations," in *International Conference on Learning Representations*, 2016.

[36] Y. Wang, M. Long, J. Wang, Z. Gao, and P. S. Yu, "Predrnn: Recurrent neural networks for predictive learning using spatiotemporal lstms," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[37] Y. Wang, Z. Gao, M. Long, J. Wang, and S. Y. Philip, "Predrnn++: Towards a resolution of the deep-in-time dilemma in spatiotemporal predictive learning," in *International Conference on Machine Learning*, 2018, pp. 5123–5132.

[38] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. Van Der Smagt, D. Cremers, and T. Brox, "Flownet: Learning optical flow with convolutional networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 2758–2766.

[39] K. Sakurada and T. Okatani, "Change detection from a street image pair using cnn features and superpixel segmentation," in *British Machine Vision Conference*, 2015.

[40] P. F. Alcantarilla, S. Stent, G. Ros, R. Arroyo, and R. Gherardi, "Street-view change detection with deconvolutional networks," *Autonomous Robots*, vol. 42, pp. 1301–1322, 2018.

[41] J.-M. Park, J.-H. Jang, S.-M. Yoo, S.-K. Lee, U.-H. Kim, and J.-H. Kim, "Changesim: Towards end-to-end online scene change detection in industrial indoor environments," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2021, pp. 8578–8585.

[42] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 16 000–16 009.

[43] K. Clark, M.-T. Luong, Q. V. Le, and C. D. Manning, "Electra: Pre-training text encoders as discriminators rather than generators," in *International Conference on Learning Representations*, 2020.

[44] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "Lora: Low-rank adaptation of large language models," in *International Conference on Learning Representations*, 2021.

[45] M. Jia, L. Tang, B.-C. Chen, C. Cardie, S. Belongie, B. Hariharan, and S.-N. Lim, "Visual prompt tuning," in *European Conference on Computer Vision*, 2022, pp. 709–727.

[46] Y. Lei, D. Peng, P. Zhang, Q. Ke, and H. Li, "Hierarchical paired channel fusion network for street scene change detection," *IEEE Transactions on Image Processing*, vol. 30, pp. 55–67, 2020.

[47] J.-M. Park, U.-H. Kim, S.-H. Lee, and J.-H. Kim, "Dual task learning by leveraging both dense correspondence and mis-correspondence for robust change detection with imperfect matches," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 13 749–13 759.