

# Change-Agent: Towards Interactive Comprehensive Remote Sensing Change Interpretation and Analysis

Chenyang Liu, Keyan Chen, Haotian Zhang, Zipeng Qi, Zhengxia Zou, and Zhenwei Shi\*, *Senior Member, IEEE*  
Beihang University

**Abstract**—Monitoring changes in the Earth’s surface is crucial for understanding natural processes and human impacts, necessitating precise and comprehensive interpretation methodologies. Remote sensing satellite imagery offers a unique perspective for monitoring these changes, leading to the emergence of remote sensing image change interpretation (RSICI) as a significant research focus. Current RSICI technology encompasses change detection and change captioning, each with its limitations in providing comprehensive interpretation. To address this, we propose an interactive Change-Agent, which can follow user instructions to achieve comprehensive change interpretation and insightful analysis according to user instructions, such as change detection and change captioning, change object counting, change cause analysis, etc. The Change-Agent integrates a multi-level change interpretation (MCI) model as the eyes and a large language model (LLM) as the brain. The MCI model contains two branches of pixel-level change detection and semantic-level change captioning, in which multiple BI-temporal Iterative Interaction (BI3) layers utilize Local Perception Enhancement (LPE) and the Global Difference Fusion Attention (GDFA) modules to enhance the model’s discriminative feature representation capabilities. To support the training of the MCI model, we build the LEVIR-MCI dataset with a large number of change masks and captions of changes. Extensive experiments demonstrate the effectiveness of the proposed MCI model and highlight the promising potential of our Change-Agent in facilitating comprehensive and intelligent interpretation of surface changes. To facilitate future research, we will make our dataset and codebase of the MCI model and Change-Agent publicly available at <https://github.com/Chen-Yang-Liu/Change-Agent>

**Index Terms**—Interactive Change-Agent, change captioning, change detection, multi-task learning, large language model.

## I. INTRODUCTION

**T**HE changes in the Earth’s surface, as a dynamic indicator of the Earth’s system, profoundly affect the planet’s evolution and the survival of humankind. Observing and analyzing these changes is pivotal for advancing sustainable human development [1]. Remote sensing (RS) satellite imagery, offering a unique “God’s perspective,” emerges as an effective tool for monitoring Earth’s dynamic changes. Remote sensing image change interpretation (RSICI) has emerged as a significant research focus [2]. RSICI aims to identify and analyze changes from images captured at different times in the same geographical area. It can provide decision-making support for environmental protection [3], urban planning [4], resource management [5], etc.

Current RSICI technology primarily encompasses change detection and change captioning. Change detection accurately localizes the spatial location of surface changes at the pixel

level [6]–[10]. In contrast, change captioning aims at articulating the attributes and meanings of changes using natural language, emphasizing the semantic-level understanding of changes [11]–[15]. Despite significant advancements in both areas, their respective limitations prevent us from obtaining comprehensive change interpretation information by utilizing a single technology. Specifically, while change detection can accurately localize the changed area, it lacks a deeper understanding of changes’ underlying meanings, such as the characteristics of ground objects or the spatial relationships between them. Conversely, change captioning can provide rich semantic-level interpretation information but may fall short in providing precise pixel-level change localization. Therefore, there is a pressing need to explore a multi-level change interpretation (MCI) approach providing both pixel-level and semantic-level change information, facilitating precise change localization while delving into the essence and implications of the changes.

Moreover, practical applications often require comprehensive analysis and further processing of both pixel-level and semantic-level change interpretation results to meet specific needs. For instance, users may require statistical analysis of changed object numbers, consuming significant time and effort from researchers and demanding users’ technical proficiency.

To address these challenges, we propose an interactive Change-Agent based on a novel MCI model and a large language model (LLM). Fig.1 illustrates the advancements achieved by our Change-Agent compared to previous technologies. Different from the previous single technology, with the help of the MCI model, our Change-Agent can simultaneously achieve precise pixel-level change detection and semantic-level change captioning, thereby providing users with comprehensive change interpretation information. Notably, our Change-Agent boasts interactive capabilities, enabling users to communicate their queries or requirements regarding surface changes. Leveraging the MCI model and the LLM, the Change-Agent can adeptly interpret changes according to user needs, intelligently analyze and process the change interpretation information, and ultimately deliver tailored outcomes aligning with user expectations. The interactive Change-Agent bridges the gap between users and remote sensing expertise.

As illustrated in Fig.1 (b), the MCI model is a pivotal cornerstone toward the realization of the Change-Agent. It enables the Change-Agent to perceive visual changes comprehensively, serving as the change perception foundation of the Change-Agent. However, there has been no prior work investigating the multi-task learning of change detection and

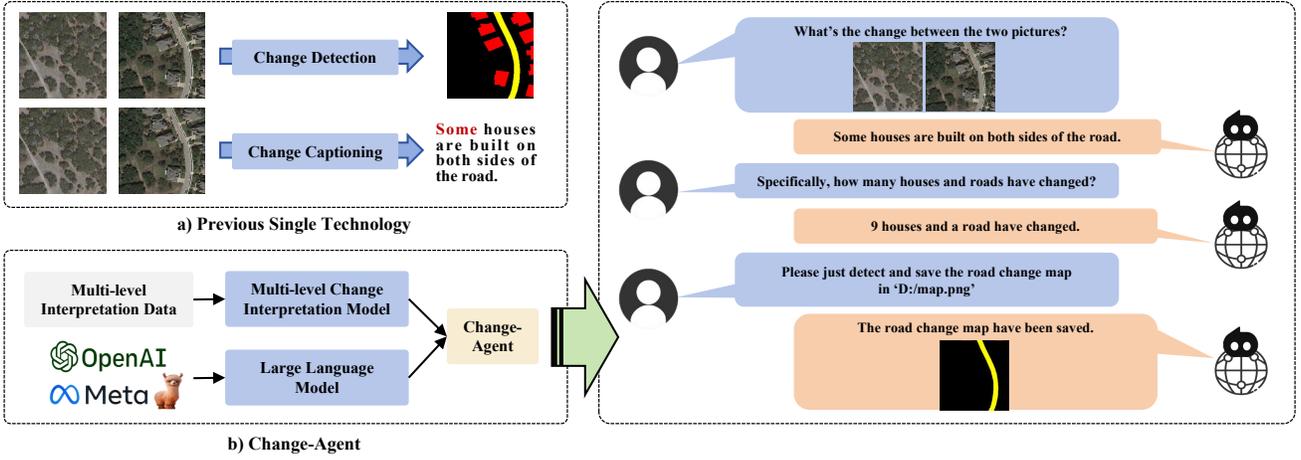


Fig. 1. The comparison between previous single technology and our Change-Agent. Our Change-Agent can simultaneously achieve precise pixel-level change detection and semantic-level change captioning. Besides, it boasts interactive capabilities, enabling users to communicate their queries or requirements regarding surface changes.

change captioning. To address this gap, we built the LEVIR-MCI dataset. Each pair of bi-temporal remote sensing images in this dataset contains both pixel-level change detection masks and semantic-level change descriptions. Building upon this dataset, we propose a novel MCI model with two branches of change detection and change captioning. Within the two branches, we propose BI-temporal Iterative Interaction (BI3) layers utilizing Local Perception Enhancement (LPE) and the Global Difference Fusion Attention (GDFA) modules to enhance the model’s discriminative feature representation capabilities. The trained MCI model equips our Change-Agent with eyes, enabling the agent to realize comprehensive change interpretations of surface changes.

Recent advancements in LLMs such as ChatGPT [16] and Llama2 [17] have been remarkable. Trained on vast corpora, LLMs have acquired extensive knowledge and demonstrated powerful capabilities in instruction comprehension, planning, reasoning, question answering, and text generation [18], [19]. These notable progress have provided robust support for our LLM-based Change-Agent. In this work, the LLM serves as the brain of the Change-Agent, assuming a pivotal role in internal scheduling, planning, and user-intent understanding. Integration with LLMs imbues our agent with enhanced flexibility and intelligence, enabling a nuanced understanding of user intentions and provision of customized change interpretation and intelligent analysis services such as changed object counting, estimation of change causes, prediction of future changes, etc.

Our Change-Agent facilitates accurate and comprehensive change interpretation of RS images. It reduces the workload and time costs for researchers while enhancing the efficiency and convenience of interaction between users and remote sensing data. Extensive experiments demonstrate the effectiveness of the proposed MCI model and highlight the promising potential of our Change-Agent. Besides, the integration of the MCI model and LLM presents novel perspectives for analyzing surface changes.

Our contributions can be summarized as follows:

- We built a multi-level change interpretation dataset. The dataset contains bi-temporal images as well as diverse change detection masks and descriptive sentences. It provides a crucial data foundation for exploring multi-task learning for change detection and change captioning.
- We propose a dual-branch MCI model, which can provide pixel-level and semantic-level change interpretation. Besides, we propose BI3 layers with LPE and GDFA to enhance the change interpretation capability of the model. Experiments validate the effectiveness of our method.
- Leveraging the MCI model and an LLM, we construct a Change-Agent that achieves an interactive and comprehensive interpretation and analysis of surface changes. It has intelligent dialogue and customized service capabilities, opening up new opportunities for intelligent remote sensing applications.

## II. RELATED WORK

A comprehensive change interpretation model should provide both pixel-level and semantic-level change information. However, previous work has primarily focused on either change detection or change captioning alone. Change detection aims to generate pixel-level change maps revealing the locations of changes in bi-temporal images, while change captioning aims to describe the changes via language. In this section, we will briefly review these approaches.

### A. Remote Sensing Change Detection

Early change detection approaches were traditionally categorized into algebra-based, transformation-based, and classification-based methods. These methods relied on techniques such as change vector analysis (CVA) [20], [21], principal component analysis (PCA) [22], [23], multivariate alteration detection (MAD) [24], [25], and post-classification comparisons [26], [27]. Traditional change detection methods have laid the foundation for modern techniques in remote sensing change detection [28].

With the emergence of deep learning, the field of remote sensing change detection has witnessed significant advancements [29]–[33]. Compared to traditional methods, deep learning significantly improves the performance of change detection with its superior multi-temporal feature learning capability. These methods utilize various architectures including convolutional neural networks (CNNs) [34]–[37], recurrent neural networks (RNNs) [38]–[42], auto-encoders [43]–[45], and Transformers [6], [46]–[50]. For instance, Chen *et al.* [46] introduced a bi-temporal image Transformer (BIT) [46], where images are represented as semantic tokens, and the Transformer encoder and decoder refine visual tokens to identify changed areas efficiently and effectively. Li *et al.* [6] proposed an end-to-end TransUNetCD model based on Transformer and UNet structure. The model utilizes a Transformer encoder to extract global contextual features. A CNN-based decoder upsamples the multi-scale features to generate the change map. Bandara *et al.* proposed a pure transformer model named ChangeFormer [51], whose backbone is a siamese SegFormer [52]. Zhang *et al.* [9] propose a deep siamese network with multi-scale multi-attention. They enhance global representation using a contextual attention module and fuse multi-scale features from the siamese feature extractor for detecting objects of varying sizes and irregularities. Some surveys, such as [1], [2], [5] provide a more comprehensive review of change detection in deep learning.

### B. Remote Sensing Change Captioning

Remote sensing change captioning (RSCC) is a recently emerged multimodal task involving remote sensing image processing and natural language generation. Given the intricacy of this task, prevailing methodologies predominantly leverage deep learning techniques. Early methods in this domain largely drew inspiration from the encoder-decoder framework commonly utilized in image captioning tasks. Within this framework, the encoder is responsible for extracting features from bi-temporal images and performing the difference-aware fusion of these features. The decoder typically employs recurrent neural networks or Transformers to convert visual features into natural language. Compared to tasks like image captioning, which extract semantic representations from single images, change description faces the significant challenge of extracting robust semantic change representations.

Chouaf and Hoxha *et al.* [11], [12] pioneered remote sensing change captioning by introducing a model that employs a pre-trained CNNs [53] as the encoder and tried to utilize an RNN and Support Vector Machines (SVMs) as the decoder to generate textual descriptions. To improve the change-awareness of the model, they proposed an image-based fusion strategy and a feature-based fusion strategy. To facilitate research on change captioning, Liu *et al.* [13] introduced a large LEVIR-CC dataset and benchmarked some methods from other fields. Besides, they proposed a Transformer-based method named RSICCformer. The approach incorporates Siamese cross-encoding modules and multistage bitemporal fusion modules, emphasizing change regions through differencing features and capturing multiple changes of interest.

In a further advancement, some attention-based improvements have emerged [14], [54], [55]. For instance, PSNet [54] enhances the perception of changed objects of varying sizes using a progressive scale-aware network with difference perception layers and scale-aware reinforcement modules. Chang *et al.* [14] proposed an attention network containing a hierarchical self-attention module followed by a residual unit for identifying change-related attributes and constructing the semantic change embedding. Different from the encoder-decoder framework, Liu *et al.* [15] proposed a new paradigm to decouple the RSCC task by designing an image-level classifier and a feature-level encoder. Besides, a multi-prompt learning strategy is proposed to effectively exploit the potential of the LLM for captioning.

### III. LEVIR-MCI DATASET

To cultivate a robust MCI model, foundational to the Change-Agent, we propose an MCI dataset named LEVIR-MCI (LEVIR Multi-level Change Interpretation)<sup>1</sup>, featuring both pixel-level change information in the form of change detection masks and semantic-level insights encapsulated in descriptive sentences. The LEVIR-MCI dataset contains 10077 bi-temporal images. Each image has a spatial size of  $256 \times 256$  pixels with a high resolution of 0.5 m/pixel and has a corresponding annotated mask and five annotated captions. The dataset will be publicly available at: <https://github.com/Chen-Yang-Liu/Change-Agent>

The proposed LEVIR-MCI dataset is an extension of our previously established change captioning dataset, LEVIR-CC [13]. In this dataset, we further provided each pair of bi-temporal images with additional change detection masks highlighting changed roads and changed buildings. Unlike its predecessor, the LEVIR-MCI dataset provides diverse annotations from different interpretation perspectives for each pair of images, further enhancing its utility for comprehensive change interpretation.

Fig. 2 presents some examples from our proposed LEVIR-MCI. Each pair of bi-temporal images within the dataset is annotated by a change detection mask delineating pixel-level alterations, along with one of five descriptive sentences elucidating the semantic-level changes. Such complementary annotations provide a multifaceted view of change interpretation, thereby bolstering the comprehensiveness of change interpretation models. To facilitate a nuanced understanding of the dataset, and considering our previous research has analysed the change captions, we will elucidate additional insights into the annotated change detection masks.

#### A. Number of Changed Objects

In Table I, we meticulously analyse changed object masks in the LEVIR-MCI dataset. The dataset comprises over 40,000 annotated instances of changed roads and buildings. While the number of changed roads is fewer than buildings, generally exhibit longer spans and cover larger areas, as evidenced in

<sup>1</sup>LEVIR is our laboratory name: the Learning, Vision, and Remote Sensing Laboratory.

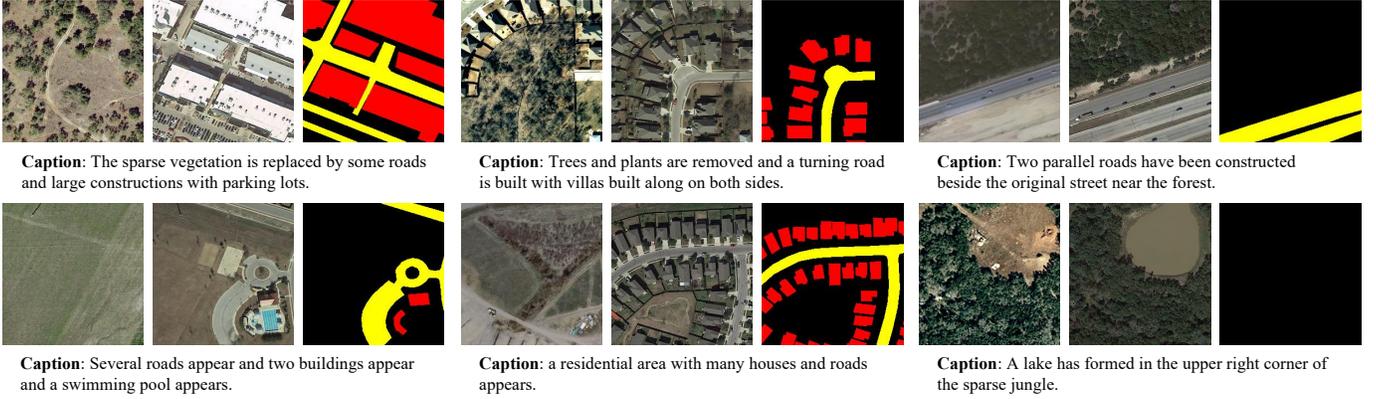


Fig. 2. Examples of the LEVIR-MCI dataset. Each pair of bi-temporal images is provided with a change detection mask and one of the five sentences describing changes. In the change detection mask, changed buildings are highlighted in red, while changed roads are depicted in yellow.

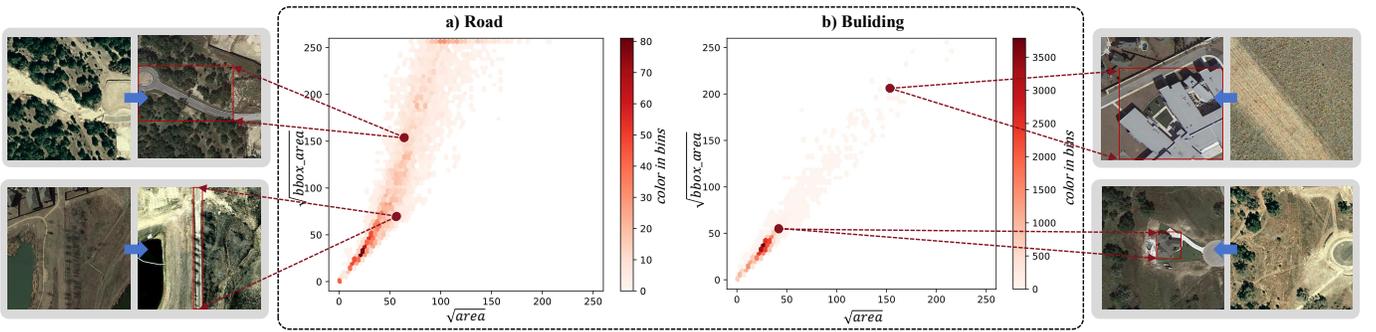


Fig. 3. Distribution of the scale and deformation of changed roads and buildings. The 'area' on the horizontal axis represents the area of a single object, and the 'bbox\_area' on the vertical axis represents the area of the corresponding rectangular bounding box. The dispersion of points offers insights into the diversity of object scale and deformation.

TABLE I

THE NUMBER OF CHANGED ROADS AND BUILDINGS IN THE LEVIR-MCI DATASET. THE DATASET COMPRISES OVER 40,000 ANNOTATED INSTANCES OF CHANGED ROADS AND BUILDINGS.

| Set            | Number of changed objects |          | Number of images with changed objects |          | Average number of objects per image |          |
|----------------|---------------------------|----------|---------------------------------------|----------|-------------------------------------|----------|
|                | Road                      | Building | Road                                  | Building | Road                                | Building |
| Training-set   | 3457                      | 26155    | 2364                                  | 2972     | 1.46                                | 8.80     |
| Validation-set | 611                       | 4709     | 424                                   | 567      | 1.44                                | 8.31     |
| Test-set       | 934                       | 8514     | 603                                   | 878      | 1.55                                | 9.70     |
| Total dataset  | 5002                      | 39378    | 3391                                  | 4417     | 1.48                                | 8.92     |

Fig. 3. With such a vast collection of annotated masks, our LEVIR-MCI dataset can also serve as a fertile data ground for developing innovative multi-category change detection methodologies.

### B. Scale and Deformation of Changed Objects

An insightful examination of object scale and deformation is conducted through an analysis of road and building areas and the corresponding rectangular bounding boxes, as illustrated in Fig. 3. The 'area' on the horizontal axis represents the area of a single object, and the 'bbox\_area' on the vertical axis represents the area of the corresponding bounding box. The dispersion of points offers insights into the diversity of object scale and deformation. Notably, points of roads exhibit a relatively broader spectrum with dispersed diversity,

attributable to their narrow and curved nature. In contrast, points of buildings display a more concentrated distribution, reflecting their predominantly rectangular form. This analysis elucidates the intricate interplay between object scale and shape, shedding light on the nuanced characteristics of changed objects.

## IV. METHODOLOGY

In Fig. 4, we present the overview of the proposed Change-Agent. We utilize the MCI model as eyes and an LLM as the brain to build our Change-Agent. The MCI model enables the Change-Agent to perceive visual changes comprehensively, serving as the change perception foundation of the Change-Agent. The constructed LEVIR-MCI dataset supports the multi-task training of the MCI model, which contains

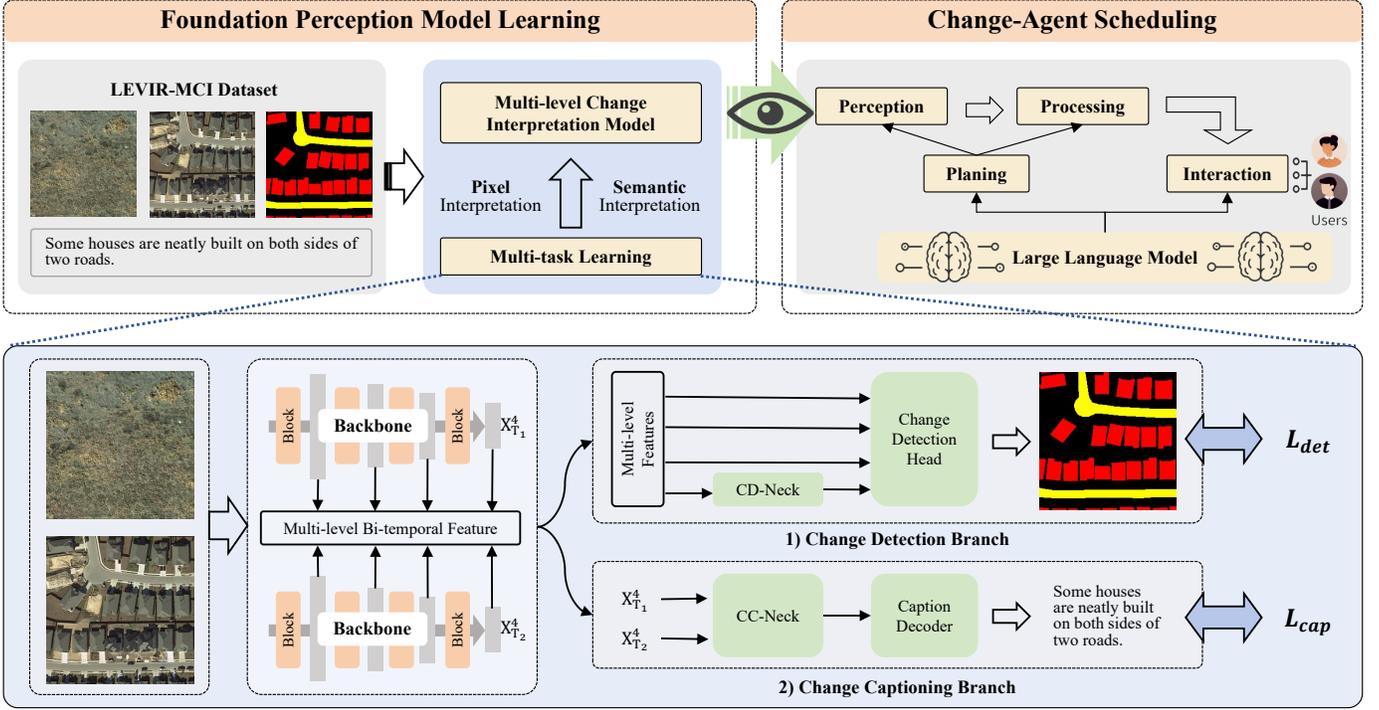


Fig. 4. The overview of Change-Agent. The Change-Agent is equipped with an MCI model and an LLM, serving as its eyes and brain, respectively. The proposed LEVIR-MCI dataset provides a data foundation for training the MCI model.

a change detection branch and a change captioning branch. As another pivotal role of the agent, the LLM can use its inherent rich knowledge to implement agent scheduling and provide insightful analyses and decision-making support. In the subsequent section, we will introduce the composition of the MCI model and the LLM-based scheduling of the agent.

#### A. Multi-level Change Interpretation Model

The MCI model serves as a central component of the Change-Agent, tasked with extracting and interpreting change information from bi-temporal remote sensing images. The proposed LEVIR-MCI dataset lays the groundwork for training the MCI model. The MCI model adopts a dual-branch architecture with a shared bottom, focusing on two pivotal tasks: change detection and change captioning. Specifically, a Siamese backbone network extracts multi-scale visual features from bi-temporal images, enabling learning in two branches. The lower-level features offer detailed information, while higher-level features are rich in semantics. Our change detection branch utilizes multi-scale features to refine change mask predictions, while the change captioning branch leverages the highest-level visual features to generate descriptive sentences. Through multi-task learning, we train a robust MCI model capable of simultaneously generating change detection masks and change captions.

#### B. Bi-temporal Iterative Interaction Layer

In the change detection and change captioning branches, we propose a novel BI-temporal Iterative Interaction (BI3) layer to effectively enhance and fuse bi-temporal features. The

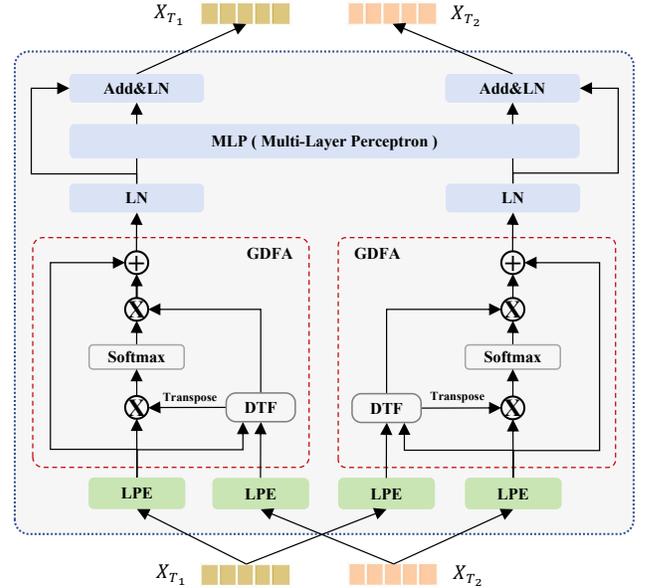


Fig. 5. The structure of the Bi-temporal Iterative Interaction (BI3) layer. The BI3 layer utilizes the Local Perception Enhancement (LPE) and the Global Difference Fusion Attention (GDFA) to enhance the features of interest.

structure of the BI3 layer is illustrated in Fig. 5. The BI3 layer utilizes the Local Perception Enhancement (LPE) module and the Global Difference Fusion Attention (GDFA) module to extract the discriminative features of interest.

Motivated by the analysis presented in Section III-B regarding the various scales and deformations of roads and buildings, we propose the LPE module with a residual structure. The LPE

module employs convolution kernels of different sizes to extract multiple feature maps across different scales. This design enriches the diversity of feature information and improves the model's local feature perception capabilities. Assuming the bi-temporal features are denoted as  $X_{T_1} \in \mathbb{R}^{N \times C}$  and  $X_{T_2} \in \mathbb{R}^{N \times C}$ , the LPE module can be formulated as follows:

$$\Phi_{\text{LPE}}(X_{T_i}) = \text{ReLU}(\text{BN}(\text{conv}_{1 \times 1}(F))) + X_{T_i} \quad (1)$$

$$F = \text{concat}([X_{s_1}, X_{s_2}, X_{s_3}]) \quad (2)$$

$$X_{s_1} = \text{conv}_{3 \times 3}(X_{T_i}) \quad (3)$$

$$X_{s_2} = \text{conv}_{5 \times 5}(X_{T_i}) \quad (4)$$

$$X_{s_3} = \text{conv}_{1 \times 5}(X_{T_i}) \quad (5)$$

where  $\text{conv}_{1 \times 1}$ ,  $\text{conv}_{3 \times 3}$ ,  $\text{conv}_{5 \times 5}$ ,  $\text{conv}_{1 \times 5}$  represent  $1 \times 1$ ,  $3 \times 3$ ,  $5 \times 5$ , and  $1 \times 5$  convolution kernels.  $\text{concat}$  stands for concatenation operation,  $\text{BN}$  (Batch Normalization) [56] for normalization, and  $\text{ReLU}$  (Rectified Linear Unit) for nonlinear activation function.

The GDFA module leverages differencing features to generate spatial attention weights and perform the interaction and fusion between features. This facilitates the model to focus on changes of interest and ignore irrelevant disturbances. Taking the left GDFA module in Figure 5 as an example, assuming  $X_{lp_1} \in \mathbb{R}^{N \times C}$  and  $X_{lp_2} \in \mathbb{R}^{N \times C}$  are the bi-temporal features obtained from the LPE module, the GDFA module can be formulated as follows:

$$\Phi_{\text{GDFA}}(X_{lp_1}, X_{lp_2}) = \sigma\left(\frac{QK^T}{\sqrt{d}}\right)V \quad (6)$$

$$Q = X_{lp_1}W_q + b_q \quad (7)$$

$$K = GW_k + b_k \quad (8)$$

$$V = GW_v + b_v \quad (9)$$

$$Di = X_{lp_2} - X_{lp_1} \quad (10)$$

$$G = \Phi_{\text{DTF}}(Di, X_{L_1}) = (Di * X_{L_1})W_d + b_d \quad (11)$$

where  $\sigma$  is the softmax function,  $W_q, W_k, W_v, W_d \in \mathbb{R}^{C \times d}$  are trainable weight matrices.  $b_q, b_k, b_v, b_d \in \mathbb{R}^d$  are trainable bias.  $C$  denotes the dimension of  $X_{LP_i}$  ( $i=1,2$ ).  $d$  is the dimension of transformed features.

Through the combination of LPE module and GDFA module, the BI3 layer improves the model's feature representation and change discrimination capabilities. Following the LPE and GDFA modules, Layer Normalization (LN) [57] is applied to the bi-temporal features. Subsequently, a Multi-Layer Perceptron (MLP) with residuals further refines the normalized features to obtain enhanced bi-temporal features.

### C. Change Detection and Captioning Branch

The structure of the change detection branch is depicted in Fig. 6. Leveraging multi-scale bi-temporal features extracted from the backbone network, this branch facilitates refined mask prediction. Specifically, multiple Bi-temporal Iterative Interaction (BI3) layers with residual connections iteratively enhance and refine bi-temporal high-level features, effectively capturing semantic changes. Since the low-level features contain more detailed information, they are crucial for refined change boundary detection. We further incorporate multiple

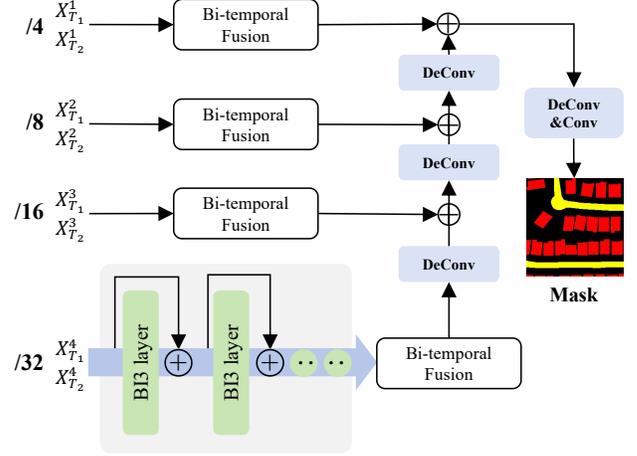


Fig. 6. The structure of the change detection branch. “DeConv” denotes the deconvolution. The multi-scale features are gradually fused from bottom to top for refined mask prediction.

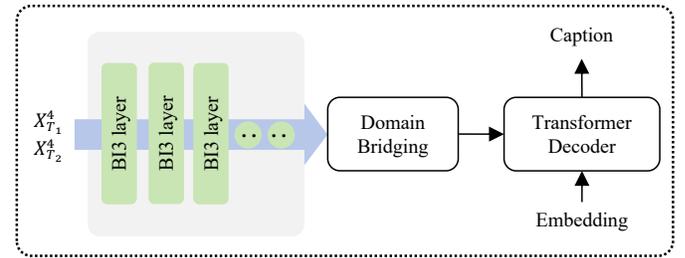


Fig. 7. The structure of the change captioning branch. The domain bridging module further bridges the conversion from the visual domain to the textual domain. The Transformer decoder generates captions.

Convolution-Based Bi-temporal Fusion (CBF) modules for bi-temporal feature fusion across four scales. Subsequently, through deconvolution, features are progressively integrated from bottom to top, bolstering the model's discrimination capability for changes and enhancing change detection accuracy. For the input  $X_{T_1}^s, X_{T_2}^s$  ( $s = 1, 2, 3, 4$ ), the CBF module can be represented as follows:

$$\Phi_{\text{CBF}}(X_{T_1}^s, X_{T_2}^s) = \text{conv}_{1 \times 1}(\text{ReLU}(\text{BN}(\text{conv}_{3 \times 3}(F)))) \quad (12)$$

$$F = \text{concat}([X_{T_1}^s, S, X_{T_2}^s]) \quad (13)$$

$$S = \text{conv}_{3 \times 3}(X_{T_2}^s - X_{T_1}^s) + \cos(X_{T_1}^s, X_{T_2}^s) \quad (14)$$

where  $\cos(\cdot)$  denotes cosine similarity operation.

The change captioning branch focuses on semantic-level change interpretation, translating visual changes into textual descriptions, as depicted in Fig. 7. This branch leverages multiple BI3 layers to interactively process high-level semantic features extracted from the backbone network, thereby obtaining bi-temporal visual features revealing changes of interest. Subsequently, a convolution-based domain bridging module further processes the bi-temporal features to facilitate the transition from the visual domain to the textual domain. Finally, these processed features are fed into a Transformer decoder to generate descriptive sentences elucidating the changes. The processing flow of the domain bridging module can be expressed as follows:

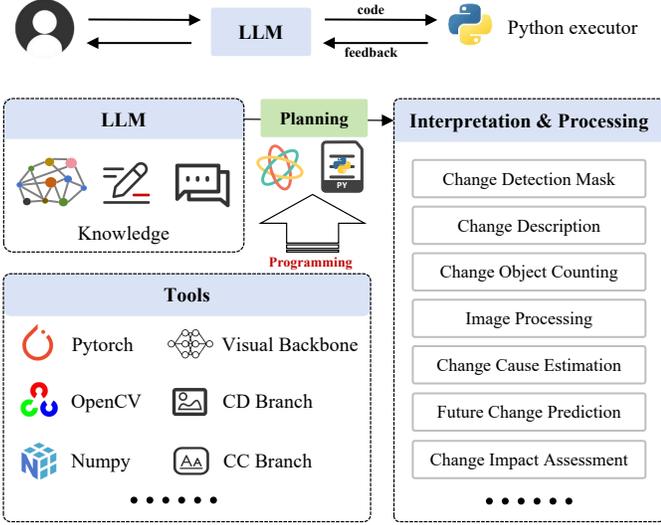


Fig. 8. The LLM plans how to complete the task according to the user’s instructions. We provide a suite of Python tools. The LLM drafts Python programs, subsequently executed by a Python interpreter to accomplish tasks beyond its inherent capabilities.

$$\Phi_{DB}(X_{T_1}, X_{T_2}) = F_1 + \text{conv}_{1 \times 1}(F_2) \quad (15)$$

$$F_1 = \text{conv}_{1 \times 1}(\text{concat}([X_{T_1}, X_{T_2}])) \quad (16)$$

$$F_2 = \text{conv}_{3 \times 3}(\text{ReLU}(\text{BN}(\text{conv}_{1 \times 1}(F_1)))) \quad (17)$$

#### D. LLM: Brain of Change-Agent

Recently, LLMs have garnered significant attention for their remarkable achievements. By harnessing extensive web knowledge, LLMs have demonstrated substantial promise in attaining human-level intelligence, exhibiting noteworthy capabilities in tasks encompassing instruction comprehension, planning, reasoning, and proficient natural language interactions with humans [58]–[61]. Building upon these advancements, there has been a growing research field that employs LLMs as central controllers for constructing autonomous agents [62]–[65].

Inspired by these developments, we harness the capabilities of LLMs as the cognitive core responsible for orchestrating the scheduling of our Change-Agent. As depicted in Fig. 8, the LLM meticulously plans task execution based on user instructions. Despite excelling at text-related tasks, LLMs lack inherent visual perception capabilities. To bridge this gap and enable change interpretation and analysis akin to human capability, we equip our agent with a suite of Python tools. These tools include a visual feature extraction backbone, change detection branch, change captioning branch, and relevant Python libraries. Leveraging these tools, the LLM autonomously crafts Python programs, subsequently executed by a Python interpreter to accomplish tasks beyond its intrinsic capabilities. Subsequently, the LLM processes the results and furnishes feedback to the user. In addition to furnishing change masks and descriptions, our agent can implement object counting, image processing, change cause estimation, prediction

of future changes, and more. In essence, our Change-Agent adeptly provides users with pertinent information, harnessing the rich knowledge of the LLM in tandem with the provided tools.

To facilitate the accurate generation of formatted Python code and tool invocation by the LLM, a meticulously crafted text prompt is essential. Inspired by the prompt utilized in [63], we devised a text prompt instructing the model on proper tool utilization. This prompt is input to the LLM in the role of a system instruction, as shown in Fig. 9.

## V. EXPERIMENTS

### A. Objective Function for Multi-task Training

During the training phase of our MIC model, we perform multi-task learning on both change detection and change captioning. Utilizing the meticulously constructed LEVIR-MCI dataset, we iteratively update the model parameters in a supervised learning framework.

For the change detection, we utilize the cross-entropy loss function to quantify the dissimilarity between predicted change masks and ground truth annotations. Specifically, the loss function can be mathematically defined as follows:

$$\mathcal{L}_{det} = -\frac{1}{H \times W} \sum_{h=1, w=1}^{H, W} \sum_{j=1}^C y_{hw}^{(j)} \log(p_{hw}^{(j)}) \quad (18)$$

where  $C$  is the number of classes.  $H$  and  $W$  are the height and width of the change mask.  $\mathbf{p}_{hw} = [p_{hw}^{(0)}, \dots, p_{hw}^{(C)}]$  is the predicted probability vector at the pixel position  $(h, w)$ .  $\tilde{\mathbf{y}}_{hw} = [\tilde{y}_{hw}^{(0)}, \dots, \tilde{y}_{hw}^{(C)}]$  is the one-hot vector representation of the ground-truth at the pixel position  $(h, w)$ .

Similarly, for change captioning, we utilize the cross-entropy function to compute the loss between predicted sentences and their corresponding ground truth sentences. The formulation of the captioning loss function can be expressed as follows:

$$\mathcal{L}_{cap} = -\frac{1}{L} \sum_{l=1}^L \sum_{v=1}^V y_l^{(v)} \log(p_l^{(v)}) \quad (19)$$

where  $L$  represents the total number of word tokens,  $V$  denotes the vocabulary size,  $y_l^{(v)}$  denotes the ground truth label indicating whether  $l$ -th word belongs to vocabulary index  $v$ , and  $p_l^{(v)}$  denotes the predicted probability of  $l$ -th word being assigned to vocabulary index  $v$ .

To balance the losses of these two tasks during model training, we adopt a normalization approach, scaling the losses of both tasks to the same order of magnitude. This ensures that each task contributes equally to the overall loss function and facilitates effective simultaneous optimization of the change detection and change captioning. Mathematically, the total loss can be expressed as follows:

$$\mathcal{L}_{total} = \frac{\mathcal{L}_{det}}{\text{detach}(\mathcal{L}_{det})} + \frac{\mathcal{L}_{cap}}{\text{detach}(\mathcal{L}_{cap})} \quad (20)$$

where  $\text{detach}(\cdot)$  denotes the operation to detach the gradient flow of the loss.

```

You are a assistant who can utilize external tools.
{'Visual_Change_Process_PythonInterpreter': """"This action tool is used to detection
changed object masks or describe between two images! It also is a Python executor
capable of running Python scripts.
The input for this action tool must be valid Python function.
The function name must be 'solution'. And the function code corresponds to your thinking
process.
Just write the function 'def solution()'. For the input for this action tool, example format is
as follows:
```python
def solution():
    # import dependent packages
    import xxx
    # Initialize some variables
    variable_names_with_real_meaning = xxx
    # step 1
    mid_variable = func(variable_names_with_real_meaning)
    # step 2
    mid_variable = func(mid_variable)
    # step ....

    # final result
    final_answer = func(mid_variable)
    return final_answer
```
Note:
In addition to the commonly used python dependency packages, you can use a custom
library 'Change_Perception' from tools import Change_Perception(), which contains the
following function:
1. change_detection(path_A, path_B, savepath_mask):
    - Parameters:
        - 'path_A': Path to the first image.
        - 'path_B': Path to the second image.
        - 'savepath_mask': Path to save the mask image.
    - Returns:
        - A mask map representing the changed areas. The mask is a numpy array with
        dimensions (256,256), where each pixel value represents the following:
            - 0 stands for unchanged,
            - 1 stands for changed road, and
            - 2 stands for changed building.
2. generate_change_caption(path_A, path_B):
    - Parameters:
        - 'path_A': Path to the first image.
        - 'path_B': Path to the second image.
    - Returns:
        - A String: Sentences describing the changes between the two images.
3. compute_object_num(change_mask,object):
    - Parameters:
        - 'change_mask': The mask from the 'change_detection' function.
        - 'object': The object type to be counted. It can be either 'building' or 'road'.
    - Returns:
        - The number of changed objects.

Here is a sample how use the above function to detect changed buildings and save the
changed building area as red:
```python
mask = Change_Perception().change_detection(path_A, path_B, savepath_mask)
mask_rgb = np.zeros(mask.shape[0], mask.shape[1], 3), dtype=np.uint8)
mask_rgb[mask == 2] = [0, 0, 255] # '2' stands for changed building (red)
cv2.imwrite(savepath_mask, mask_rgb)
```
To use a tool, please use the following format:
Thought: Think about what you need to solve. Do you need to use any tools?
Action: Provide the name of the tool you want to use. It should be one of:
[['Visual_Change_Process_PythonInterpreter']]
Action Input: Specify the input required for the action tool.
The response after utilizing tools should using the following format:
Response: Display the results obtained after calling the tool.
If you already know the answer, or if you don't need to use any tools, please reply using
the following format:
Thought: Explain your thought process to arrive at the final answer.
Final Answer: final answer
Begin!

```

Fig. 9. The prompt input to LLM in the role of a system instruction.

## B. Evaluation Metrics

To evaluate the performance of our MCI model in detecting multi-category changes in remote sensing images, we use the Mean Intersection over Union (MIoU). The MIoU metric assesses the spatial overlap between predicted change masks and ground-truth masks, providing a reliable measure of pixel-level change detection accuracy. Mathematically, the calculation of MIoU is defined as:

$$\text{MIoU} = \frac{1}{N} \sum_{i=1}^N \frac{\text{TP}_i}{\text{TP}_i + \text{FP}_i + \text{FN}_i} \quad (21)$$

Where  $N$  is the total number of classes. For the  $i$ -th class,  $\text{TP}_i$ ,  $\text{FP}_i$ , and  $\text{FN}_i$  represent the number of true positive, false positive, and false negative, respectively.

To evaluate the performance of our MCI model in describing changes between bi-temporal images, we adopted several evaluation metrics commonly used in previous change captioning studies [12]–[14], [66]. These metrics are as follows:

- **BLEU-n** [67]: The BLEU-n metric measures the n-gram similarity between generated sentences and ground-truth sentences. We adopt  $n=1, 2, 3$ , and 4 to compute the BLEU-n score.
- **METEOR** [68]: The METEOR computes the harmonic mean of precision and recall of single words. Besides, it incorporates a penalty factor to consider the fluency of generated sentences.
- **ROUGE<sub>L</sub>** [69]: The ROUGE<sub>L</sub> metric measures the similarity of the longest common subsequence between generated sentences and ground-truth sentences. It is more concerned with the recall rate and is suitable for evaluating long sentences.
- **CIDEr-D** [70]: The CIDEr metric treats each sentence as a document and represents it in the form of Term Frequency Inverse Document Frequency (TF-IDF) vectors. After computing TF-IDF for each n-gram, CIDEr-D is obtained by performing the cosine similarity.

By employing this comprehensive suite of evaluation metrics, we aim to provide a thorough assessment of the change detection and captioning capabilities of our MCI model. Higher scores across these metrics indicate more accurate change masks or superior sentence quality.

## C. Experimental Details

We implemented all models using the PyTorch deep learning framework and trained them on the NVIDIA GTX 4090 GPU. During training, we minimized the loss function defined by Equation 20 and employed the Adam optimizer [71] with an initial learning rate of 0.0001 to optimize the model parameters. We set the maximum number of epochs to 200 and utilized word embeddings with a dimensionality of 512. The backbone for feature extraction is the Siamese weight-shared Segformer-B1 [52]. Initially, we trained the backbone until the sum of BLEU-4 and MIoU indicators did not increase for 50 consecutive epochs. Subsequently, we froze the backbone network and continued training the two branches separately.

TABLE II

THE COMPREHENSIVE PERFORMANCE COMPARISONS ON THE LEVIR-MCI DATASET. THERE IS A LACK OF METHODS THAT SIMULTANEOUSLY ADDRESS CHANGE DETECTION AND CHANGE CAPTIONING. TO ASSESS THE EFFICACY OF OUR MODEL IN HANDLING BOTH TASKS, WE COMPARED OUR METHOD WITH ESTABLISHED METHODS IN EACH RESPECTIVE FIELD.

| Method            |                               | MIoU         | BLEU-1       | BLEU-2       | BLEU-3       | BLEU-4       | METEOR       | ROUGE <sub>L</sub> | CIDEr-D       |
|-------------------|-------------------------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------------|---------------|
| Change Detection  | FC-EF [35]                    | 84.54        | –            | –            | –            | –            | –            | –                  | –             |
|                   | FC-Siam-Conc [35]             | 85.42        | –            | –            | –            | –            | –            | –                  | –             |
|                   | FC-Siam-Di [35]               | 85.20        | –            | –            | –            | –            | –            | –                  | –             |
|                   | BIT [46]                      | 84.45        | –            | –            | –            | –            | –            | –                  | –             |
| Change Captioning | Capt-Rep-Diff [66]            | –            | 72.90        | 61.98        | 53.62        | 47.41        | 34.47        | 65.64              | 110.57        |
|                   | Capt-Att [66]                 | –            | 77.64        | 67.40        | 59.24        | 53.15        | 36.58        | 69.73              | 121.22        |
|                   | Capt-Dual-Att [66]            | –            | 79.51        | 70.57        | 63.23        | 57.46        | 36.56        | 70.69              | 124.42        |
|                   | DUDA [66]                     | –            | 81.44        | 72.22        | 64.24        | 57.79        | 37.15        | 71.04              | 124.32        |
|                   | MCCFormer-S [72]              | –            | 79.90        | 70.26        | 62.68        | 56.68        | 36.17        | 69.46              | 120.39        |
|                   | MCCFormer-D [72]              | –            | 80.42        | 70.87        | 62.86        | 56.38        | 37.29        | 70.32              | 124.44        |
|                   | RSICCFormer <sub>c</sub> [13] | –            | 83.09        | 74.32        | 66.66        | 60.44        | 38.76        | 72.63              | 130.00        |
|                   | PSNet [54]                    | –            | 83.86        | 75.13        | 67.89        | 62.11        | 38.80        | 73.60              | 132.62        |
|                   | Chg2Cap [14]                  | –            | 86.14        | 78.08        | 70.66        | 64.39        | 40.03        | 75.12              | 136.61        |
| MCI               | Ours                          | <b>86.43</b> | <b>86.68</b> | <b>78.75</b> | <b>71.74</b> | <b>65.95</b> | <b>40.80</b> | <b>75.96</b>       | <b>140.29</b> |

#### D. Multi-level Change Interpretation Performance

Currently, there is a dearth of methods that simultaneously address change detection and change captioning. To assess the efficacy of our model in handling both tasks, we conducted comprehensive performance evaluations by comparing against established methods in each respective field on the proposed dataset.

For the change detection evaluation, we have benchmarked several well-known methods, including FC-EF [35], FC-Siam-Conc [35], FC-Siam-Di [35], and BIT [46]. Meanwhile, for the change captioning assessment, we compare our model with several existing state-of-the-art (SOTA) methods, including Capt-Rep-Diff [66], Capt-Att [66], Capt-Dual-Att [66], DUDA [66], MCCFormer-S [72], MCCFormer-D [72], RSICCFormer [13] and Chg2Cap [14].

The comprehensive performance comparisons, as shown in Table II, underscore the superiority of our proposed methodology across both change detection and change captioning tasks. Our method not only achieves simultaneous proficiency in both tasks but also surpasses previous methods. Specifically, our model exhibits superior performance in change detection with a +1.01% improvement on MIoU. Moreover, in terms of change captioning, our model exhibits a noteworthy advancement, showcasing a +1.56% improvement on BLEU-4 and a substantial +3.68% improvement on CIDEr-D.

Through the above performance analysis, we gain insights into the model’s ability to generate pixel-level and semantic-level interpretation information of changes observed in remote sensing images. This highlights the efficacy of our proposed MCI model in providing a comprehensive and in-depth analysis of surface changes.

#### E. Ablation Studies

1) *BI3 Layer*: As a critical component of our MCI model, the BI3 layer incorporates LPE and GDFA modules. The LPE module enhances feature representation through local

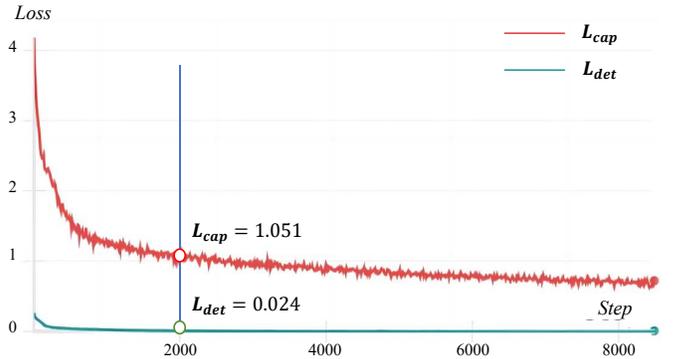


Fig. 10. The loss curve shows the imbalance in the training process of the two tasks when we add the detection loss and caption loss to form the total loss.

perception enhancement, improving the model’s sensitivity to multi-scale changes. The GDFA module effectively integrates the differential information, allowing the model to capture the change area more accurately. In Table III, we conducted ablation experiments on the effectiveness of the LPE and GDFA modules. The baseline utilizes Transformer encoding layers to process bi-temporal features. The experimental results show that integrating the LPE and GDFA modules leads to improved performance in both change detection and change captioning tasks.

2) *Balancing Change Detection and Captioning*: In our exploration of multi-task learning for change detection and change captioning, achieving an effective balance between these tasks posed a significant challenge. Initially, when we simply combined the individual losses  $\mathcal{L}_{det}$  and  $\mathcal{L}_{cap}$  to form the total loss and minimized it to optimize the multi-task model parameters, we observed large discrepancies in the magnitudes of these two losses, as illustrated in Fig. 10. This imbalance during training could potentially impact overall change interpretation performance. To address this issue, we adopt a

TABLE III  
ABLATION STUDIES ON THE EFFECTIVENESS OF THE PROPOSED BI3 LAYER, WHERE THE BOLDDED RESULTS ARE THE BEST.

| Method   | G DFA | L PE | M IoU        | BLEU-1       | BLEU-2       | BLEU-3       | BLEU-4       | METEOR       | ROUGE <sub>L</sub> | CIDEr-D       |
|----------|-------|------|--------------|--------------|--------------|--------------|--------------|--------------|--------------------|---------------|
| Baseline | ✗     | ✗    | 86.40        | 84.34        | 75.83        | 68.72        | 63.05        | 39.22        | 73.82              | 133.58        |
| MCI-G    | ✓     | ✗    | 86.41        | 85.32        | 77.45        | 70.63        | 65.10        | 39.59        | 74.45              | 135.94        |
| MCI-L    | ✗     | ✓    | 86.42        | 86.43        | 78.53        | 71.54        | 65.75        | 40.45        | 75.72              | 138.99        |
| Our MCI  | ✓     | ✓    | <b>86.43</b> | <b>86.68</b> | <b>78.75</b> | <b>71.74</b> | <b>65.95</b> | <b>40.80</b> | <b>75.96</b>       | <b>140.29</b> |

TABLE IV  
EFFECT ON THE LOSS BALANCING STRATEGY DETAILED IN EQUATION 20. “W.O. BALANCE” MEANS THAT WE SIMPLY COMBINED THE DETECTION LOSS AND CAPTION LOSS TO FORM THE TOTAL LOSS TO OPTIMIZE THE MULTI-TASK MODEL PARAMETERS.

| Method       | M IoU        | BLEU-1       | BLEU-2       | BLEU-3       | BLEU-4       | METEOR       | ROUGE <sub>L</sub> | CIDEr-D       |
|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------------|---------------|
| W.o. balance | 85.28        | <b>86.81</b> | <b>78.95</b> | <b>71.84</b> | 65.87        | 40.61        | 75.92              | 139.75        |
| W. balance   | <b>86.43</b> | 86.68        | 78.75        | 71.74        | <b>65.95</b> | <b>40.80</b> | <b>75.96</b>       | <b>140.29</b> |

TABLE V  
COMPARATIVE ANALYSIS OF SINGLE-TASK AND MULTI-TASK LEARNING. WHILE MULTI-TASK LEARNING PERFORMS POORER CHANGE DETECTION THAN SINGLE CHANGE DETECTION TRAINING, IT CONTRIBUTES TO IMPROVING CHANGE CAPTIONING CAPABILITIES, PARTICULARLY ON CRITICAL BLEU-4 AND CIDEr-D METRICS.

| Method                   | M IoU        | BLEU-1       | BLEU-2       | BLEU-3       | BLEU-4       | METEOR       | ROUGE <sub>L</sub> | CIDEr-D       |
|--------------------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------------|---------------|
| Single Change Detection  | <b>86.54</b> | –            | –            | –            | –            | –            | –                  | –             |
| Single Change Captioning | –            | 86.56        | 78.70        | 71.69        | 65.86        | 40.53        | 75.87              | 139.92        |
| Multi-task Training      | 86.43        | <b>86.68</b> | <b>78.75</b> | <b>71.74</b> | <b>65.95</b> | <b>40.80</b> | <b>75.96</b>       | <b>140.29</b> |

balancing strategy detailed in Equation 20 to normalize the two losses and achieve the training balance between the two tasks. Table IV shows the impact of these two strategies on model performance. It’s evident that our balancing strategy plays a crucial role in harmonizing the training process, leading to more balanced performance across both change detection and captioning tasks.

In addition, we conducted a comparative analysis of single-task and multi-task learning, as shown in Table V. While multi-task learning yields poorer change detection performance compared to single change detection training, it contributes to improving change captioning capabilities, particularly on critical BLEU-4 and CIDEr-D metrics. This observation could be attributed to the shared knowledge and feature representation learning across tasks. Pixel-level change detection can be advantageous for accurately localizing changes when generating change descriptions. However, it’s worth noting that ground-truth change descriptions sometimes overlook minor changes, which could be detrimental to pixel-level change detection. The trade-off between change detection and change captioning underscores the complex interplay between pixel-level and semantic-level interpretation of surface changes.

In conclusion, achieving a balance between change detection and captioning, and fostering collaborative improvement between them, remains a subject worthy of further exploration. We look forward to discovering more effective methods to address this challenge in future research.

### F. Qualitative Change Interpretation Results

To illustrate the effectiveness of our proposed model in simultaneously generating change masks and change descriptions, Fig. 11 presents comparative results between our proposed model and the baseline model. The experimental results demonstrate the superior capabilities of our model in joint interpretation of change detection and change captioning.

In terms of change detection, our model exhibits remarkable proficiency in identifying subtle changes in small buildings, as depicted in the first row of Fig. 11. Moreover, our model produces more refined road masks in rows 5 and 7 compared to the baseline. Unlike the baseline, which erroneously detects some buildings in the sixth and eighth rows, our model performs admirably.

Regarding change captioning, our model accurately recognizes and describes the changes in the bi-temporal images in the fourth and eighth rows of Fig. 11. Conversely, the baseline erroneously concludes that no changes have occurred, resulting in a critical error in the description. Furthermore, it is evident from the first four rows of Fig. 11 that our model not only accurately identifies the types of changing objects but also describes their spatial positions and the relationships between the changing objects and surrounding features.

### G. Interaction Between User and Change-Agent

Utilizing the MCI model trained on the LEVIR-MIC dataset, we have constructed an LLM-based Change-Agent. The MCI model provides our agent with powerful change interpretation capability. The LLM serves as the brain behind our agent,

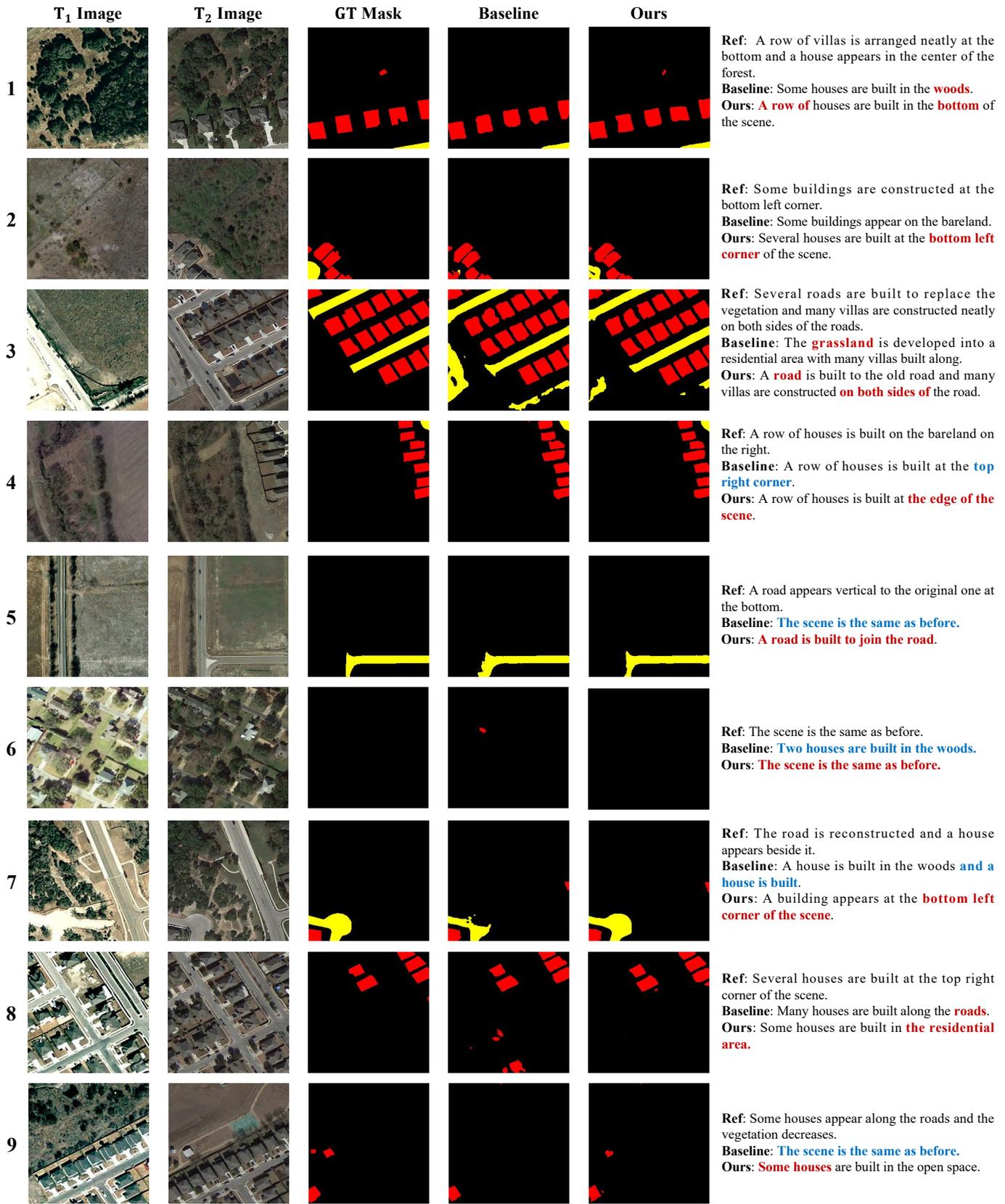


Fig. 11. The qualitative comparison results between the baseline and our model. For captioning results, we provide one of the five ground-truth reference sentences. Besides, more accurate and detailed words are marked in green, while red words are not. The results demonstrate that our model exhibits strong change interpretation capabilities in change detection and change captioning.

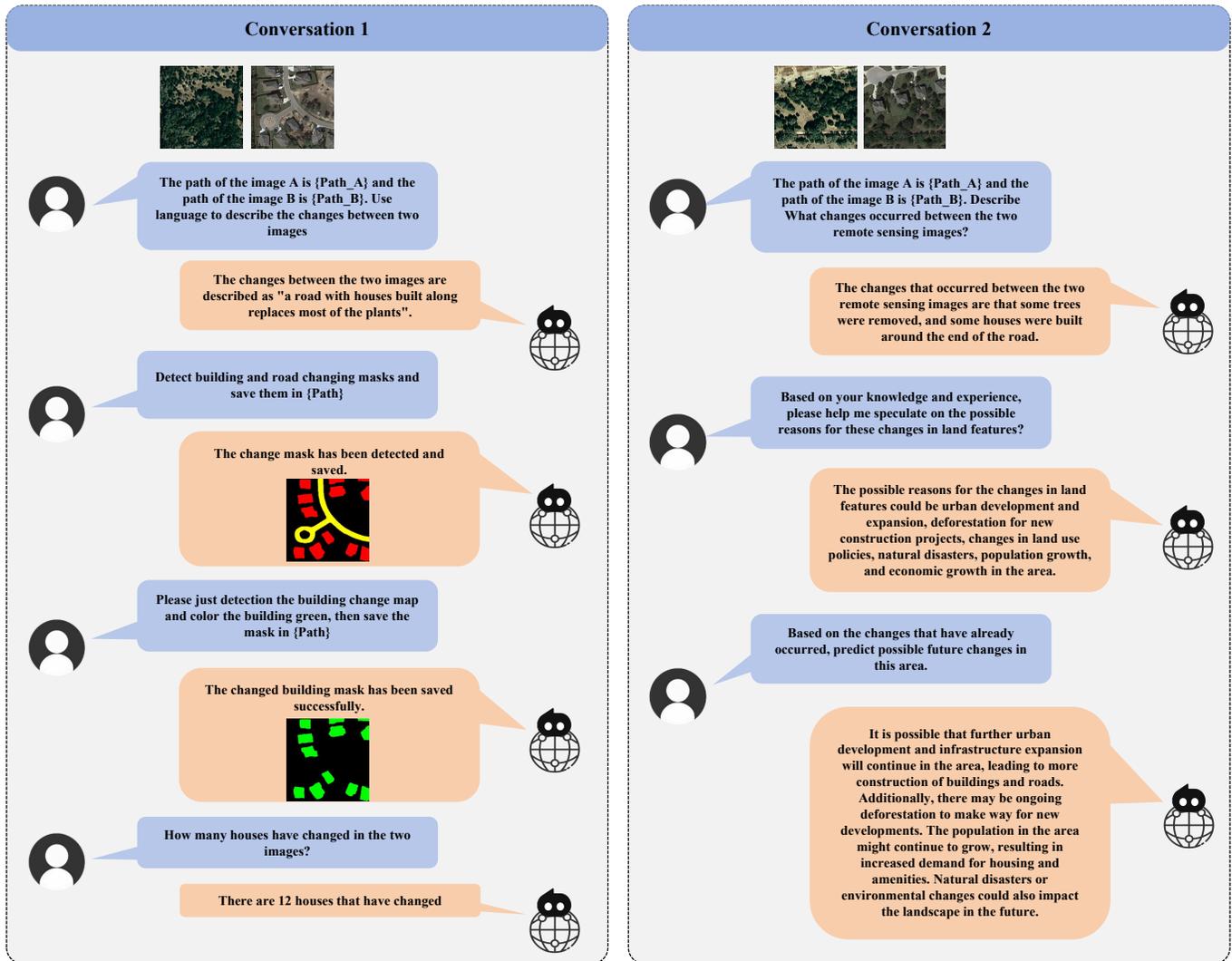


Fig. 12. Conversations between users and our Change-Agent. Our Change-Agent can understand users and carry out tasks such as change mask prediction, change captioning, change object counting, mask post-processing, change causes estimation, and future change prediction.

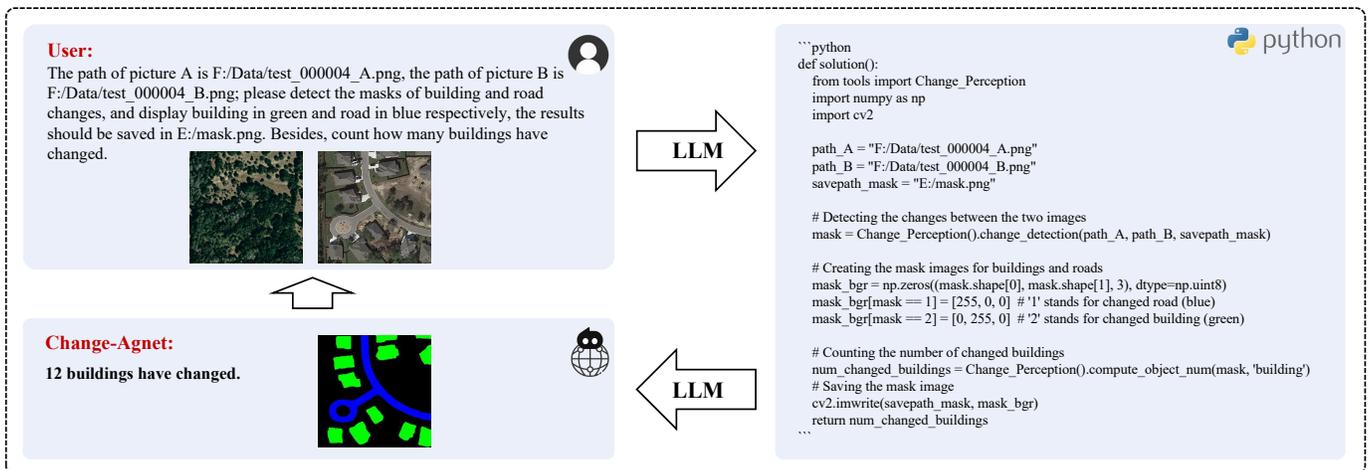


Fig. 13. The executable Python function code generated by LLM within our Change-Agent when a user requests to perform change detection, display building areas in green, display road areas in blue, and count changed buildings.

facilitating seamless interaction and comprehensive interpretation of changes.

In Fig. 12, we provide two examples of conversations between users and our Change-Agent, illustrating its adaptability and versatility. In the first conversation, our agent showcases its adeptness in understanding user instructions and subsequently carries out tasks such as change mask prediction, change captioning, changed object counting, and mask post-processing. This automated approach significantly enhances user efficiency and offers a more intuitive experience in change interpretation. In the second conversation, our agent leverages the MCI model to describe the changes in the images, and leverages the rich knowledge inherent in the LLM to provide insightful answers to user queries regarding change causality and future change predictions, aiding in user’s decision-making processes.

Fig. 13 presents a specific example to illustrate the executable code generated by the LLM within our Change-Agent. When the user requests to perform change detection, display building areas in green, display road areas in blue, and count changed buildings, the LLM efficiently generates executable Python code, demonstrating the agent’s versatility and responsiveness to user needs.

Through these interactive capabilities, our Change-Agent serves as a versatile tool for facilitating nuanced and insightful analysis of changes in remote sensing images, fostering seamless communication and collaboration between users and remote sensing expertise.

## VI. DISCUSSION AND FUTURE

Our research introduces the LEVIR-MIC dataset, which lays a solid data foundation for facilitating multi-task learning and exploring the interconnection between change detection and change captioning. Besides, our Change-Agent opens up new avenues for change interpretation and brought a new thinking to the remote sensing field. However, there are still some issues worthy of further exploration:

- **Multi-Task Learning.** Challenges such as task balance and inter-task relationships remain in multi-task learning. Future work can focus on designing better model structures and training strategies to improve overall interpretation performance on change detection and change captioning.
- **Scheduling of Change-Agent.** The hand-craft system prompts inputted into the LLM play a crucial role in facilitating accurate agent scheduling. Optimizing system prompts could enable to better grasp user intentions and plan more accurate and rational task execution pathways.
- **Tool Expansion.** Beyond change interpretation, providing the agent with additional models can enhance its capabilities for handling complex remote sensing image processing tasks, offering users more comprehensive interpretation services. Additionally, the agent’s updateability ensures it remains up-to-date with new tools and models.
- **Multi-Agent Systems.** Introducing multiple agents with diverse capabilities and fostering collaboration among them could lead to more flexible and collaborative image processing and analysis, contributing to the development of more efficient and intelligent remote sensing systems.

## VII. CONCLUSION

In this paper, we have addressed the critical need for a comprehensive and intelligent interpretation of Earth’s surface changes through developing a novel Change-Agent. Our Change-Agent can follow user instructions to achieve comprehensive change interpretation and insightful analysis according to user instructions. The Change-Agent, fueled by an MCI model and an LLM, exhibits robust visual change perception capabilities and adept scheduling. The MCI model, comprising two branches of change detection and captioning, offers pixel-level change masks and semantic-level textual change descriptions. In these two branches, we propose B3 layers with LPE and GDFA to enhance the model’s discriminative feature representation capabilities. Additionally, we build a dataset comprising diverse change detection masks and descriptions to support model training. Experiments validate the effectiveness of the proposed MCI model and underscore the promising potential of our Change-Agent in facilitating comprehensive and intelligent interpretation of surface changes.

## REFERENCES

- [1] L. Khelifi and M. Mignotte, “Deep learning for change detection in remote sensing images: Comprehensive review and meta-analysis,” *IEEE Access*, vol. 8, pp. 126 385–126 400, 2020.
- [2] W. Shi, M. Zhang, R. Zhang, S. Chen, and Z. Zhan, “Change detection based on artificial intelligence: State-of-the-art and challenges,” *Remote Sensing*, vol. 12, no. 10, p. 1688, 2020.
- [3] S. H. Khan, X. He, F. Porikli, and M. Bennamoun, “Forest change detection in incomplete satellite images with deep neural networks,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 9, pp. 5407–5423, 2017.
- [4] I. R. Hegazy and M. R. Kaloop, “Monitoring urban growth and land use change detection with gis and remote sensing techniques in daqahlia governorate egypt,” *International Journal of Sustainable Built Environment*, vol. 4, no. 1, pp. 117–124, 2015.
- [5] L. Wang, M. Zhang, X. Gao, and W. Shi, “Advances and challenges in deep learning-based change detection for remote sensing images: A review through various learning paradigms,” *Remote Sensing*, vol. 16, no. 5, p. 804, 2024.
- [6] Q. Li, R. Zhong, X. Du, and Y. Du, “Transunetcd: A hybrid transformer network for change detection in optical remote-sensing images,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–19, 2022.
- [7] Z. Zheng, Y. Zhong, S. Tian, A. Ma, and L. Zhang, “Changemask: Deep multi-task encoder-transformer-decoder architecture for semantic change detection,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 183, pp. 228–239, 2022.
- [8] Y. Feng, J. Jiang, H. Xu, and J. Zheng, “Change detection on remote sensing images using dual-branch multilevel intertemporal network,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 61, pp. 1–15, 2023.
- [9] M. Zhang, Z. Liu, J. Feng, L. Liu, and L. Jiao, “Remote sensing image change detection based on deep multi-scale multi-attention siamese transformer network,” *Remote Sensing*, vol. 15, no. 3, p. 842, 2023.
- [10] Y. Feng, J. Jiang, H. Xu, and J. Zheng, “Change detection on remote sensing images using dual-branch multilevel intertemporal network,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 61, pp. 1–15, 2023.
- [11] S. Chouaf, G. Hoxha, Y. Smara, and F. Melgani, “Captioning changes in bi-temporal remote sensing images,” in *2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS*, 2021, pp. 2891–2894.
- [12] G. Hoxha, S. Chouaf, F. Melgani, and Y. Smara, “Change captioning: A new paradigm for multitemporal remote sensing image analysis,” *IEEE Transactions on Geoscience and Remote Sensing*, pp. 1–1, 2022.
- [13] C. Liu, R. Zhao, H. Chen, Z. Zou, and Z. Shi, “Remote sensing image change captioning with dual-branch transformers: A new method and a large scale dataset,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–20, 2022.

- [14] S. Chang and P. Ghamisi, "Changes to captions: An attentive network for remote sensing change captioning," *arXiv preprint arXiv:2304.01091*, 2023.
- [15] C. Liu, R. Zhao, J. Chen, Z. Qi, Z. Zou, and Z. Shi, "A decoupling paradigm with prompt learning for remote sensing image change captioning," *IEEE Transactions on Geoscience and Remote Sensing*, 2023.
- [16] T. OpenAI, "Chatgpt: Optimizing language models for dialogue," *OpenAI*, 2022.
- [17] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale *et al.*, "Llama 2: Open foundation and fine-tuned chat models," *arXiv preprint arXiv:2307.09288*, 2023.
- [18] X. Han, Z. Zhang, N. Ding, Y. Gu, X. Liu, Y. Huo, J. Qiu, Y. Yao, A. Zhang, L. Zhang *et al.*, "Pre-trained models: Past, present and future," *AI Open*, vol. 2, pp. 225–250, 2021.
- [19] P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, and G. Neubig, "Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing," *ACM Computing Surveys*, vol. 55, no. 9, pp. 1–35, 2023.
- [20] W. A. Malila, "Change vector analysis: An approach for detecting forest changes with landsat," in *LARS symposia*, 1980, p. 385.
- [21] O. A. Carvalho Júnior, R. F. Guimarães, A. R. Gillespie, N. C. Silva, and R. A. Gomes, "A new approach to change vector analysis using distance and similarity measures," *Remote Sensing*, vol. 3, no. 11, pp. 2473–2493, 2011.
- [22] J. Deng, K. Wang, Y. Deng, and G. Qi, "Pca-based land-use change detection and analysis using multitemporal and multisensor satellite data," *International Journal of Remote Sensing*, vol. 29, no. 16, pp. 4823–4838, 2008.
- [23] T. Celik, "Unsupervised change detection in satellite images using principal component analysis and *k*-means clustering," *IEEE geoscience and remote sensing letters*, vol. 6, no. 4, pp. 772–776, 2009.
- [24] A. A. Nielsen, K. Conradsen, and J. J. Simpson, "Multivariate alteration detection (mad) and maf postprocessing in multispectral, bitemporal image data: New approaches to change detection studies," *Remote Sensing of Environment*, vol. 64, no. 1, pp. 1–19, 1998.
- [25] A. A. Nielsen, "The regularized iteratively reweighted mad method for change detection in multi-and hyperspectral data," *IEEE Transactions on Image processing*, vol. 16, no. 2, pp. 463–478, 2007.
- [26] O. Abd El-Kawy, J. Rød, H. Ismail, and A. Suliman, "Land use and land cover change detection in the western Nile delta of Egypt using remote sensing data," *Applied geography*, vol. 31, no. 2, pp. 483–494, 2011.
- [27] T. Chou, T. Lei, S. Wan, and L. Yang, "Spatial knowledge databases as applied to the detection of changes in urban land use," *International Journal of Remote Sensing*, vol. 26, no. 14, pp. 3047–3068, 2005.
- [28] A. P. Tewkesbury, A. J. Comber, N. J. Tate, A. Lamb, and P. F. Fisher, "A critical synthesis of remotely sensed optical image change detection techniques," *Remote Sensing of Environment*, vol. 160, pp. 1–14, 2015.
- [29] Z. Li, C. Yan, Y. Sun, and Q. Xin, "A densely attentive refinement network for change detection based on very-high-resolution bitemporal remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–18, 2022.
- [30] Z. Lv, H. Huang, X. Li, M. Zhao, J. A. Benediktsson, W. Sun, and N. Falco, "Land cover change detection with heterogeneous remote sensing images: Review, progress, and perspective," *Proceedings of the IEEE*, vol. 110, no. 12, pp. 1976–1991, 2022.
- [31] H. Zhang, H. Chen, C. Zhou, K. Chen, C. Liu, Z. Zou, and Z. Shi, "Bifa: Remote sensing image change detection with bitemporal feature alignment," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 62, pp. 1–17, 2024.
- [32] K. Chen, C. Liu, W. Li, Z. Liu, H. Chen, H. Zhang, Z. Zou, and Z. Shi, "Time travelling pixels: Bitemporal features integration with foundation model for remote sensing image change detection," *arXiv preprint arXiv:2312.16202*, 2023.
- [33] Z. Lv, P. Zhong, W. Wang, Z. You, and N. Falco, "Multi-scale attention network guided with change gradient image for land cover change detection using remote sensing images," *IEEE Geoscience and Remote Sensing Letters*, 2023.
- [34] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 770–778.
- [35] R. C. Daudt, B. Le Saux, and A. Boulch, "Fully convolutional siamese networks for change detection," in *2018 25th IEEE International Conference on Image Processing (ICIP)*. IEEE, 2018, pp. 4063–4067.
- [36] Y. Zhan, K. Fu, M. Yan, X. Sun, H. Wang, and X. Qiu, "Change detection based on deep siamese convolutional network for optical aerial images," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 10, pp. 1845–1849, 2017.
- [37] D. Peng, Y. Zhang, and H. Guan, "End-to-end change detection for high resolution satellite images using improved unet++," *Remote Sensing*, vol. 11, no. 11, p. 1382, 2019.
- [38] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [39] 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.
- [40] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.
- [41] 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, 2022.
- [42] H. Chen, C. Wu, B. Du, L. Zhang, and L. Wang, "Change detection in multisource vhr images via deep siamese convolutional multiple-layers recurrent neural network," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 4, pp. 2848–2864, 2019.
- [43] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *science*, vol. 313, no. 5786, pp. 504–507, 2006.
- [44] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.
- [45] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the 25th international conference on Machine learning*, 2008, pp. 1096–1103.
- [46] H. Chen, Z. Qi, and Z. Shi, "Remote sensing image change detection with transformers," *IEEE Transactions on Geoscience and Remote Sensing*, 2021.
- [47] C. Zhang, L. Wang, S. Cheng, and Y. Li, "Swinsunet: Pure transformer network for remote sensing image change detection," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–13, 2022.
- [48] M. Liu, Q. Shi, Z. Chai, and J. Li, "Pa-former: Learning prior-aware transformer for remote sensing building change detection," *IEEE Geoscience and Remote Sensing Letters*, vol. 19, pp. 1–5, 2022.
- [49] M. Zhang, Z. Liu, J. Feng, L. Liu, and L. Jiao, "Remote sensing image change detection based on deep multi-scale multi-attention siamese transformer network," *Remote Sensing*, vol. 15, no. 3, p. 842, 2023.
- [50] L. Ding, J. Zhang, H. Guo, K. Zhang, B. Liu, and L. Bruzzone, "Joint spatio-temporal modeling for semantic change detection in remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, 2024.
- [51] W. G. C. Bandara and V. M. Patel, "A transformer-based siamese network for change detection," in *IGARSS 2022-2022 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 2022, pp. 207–210.
- [52] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, "Segformer: Simple and efficient design for semantic segmentation with transformers," *Advances in Neural Information Processing Systems*, vol. 34, pp. 12 077–12 090, 2021.
- [53] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.
- [54] C. Liu, J. Yang, Z. Qi, Z. Zou, and Z. Shi, "Progressive scale-aware network for remote sensing image change captioning," *arXiv preprint arXiv:2303.00355*, 2023.
- [55] C. Cai, Y. Wang, and K.-H. Yap, "Interactive change-aware transformer network for remote sensing image change captioning," *Remote Sensing*, vol. 15, no. 23, p. 5611, 2023.
- [56] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International conference on machine learning*. pmlr, 2015, pp. 448–456.
- [57] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," *arXiv preprint arXiv:1607.06450*, 2016.
- [58] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.
- [59] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar *et al.*, "Llama: Open and efficient foundation language models," *arXiv preprint arXiv:2302.13971*, 2023.

- [60] Y. Shen, K. Song, X. Tan, D. Li, W. Lu, and Y. Zhuang, "Hugginggpt: Solving ai tasks with chatgpt and its friends in hugging face," *Advances in Neural Information Processing Systems*, vol. 36, 2024.
- [61] T. Schick, J. Dwivedi-Yu, R. Dessì, R. Raileanu, M. Lomeli, E. Hambro, L. Zettlemoyer, N. Cancedda, and T. Scialom, "Toolformer: Language models can teach themselves to use tools," *Advances in Neural Information Processing Systems*, vol. 36, 2024.
- [62] T. Gupta and A. Kembhavi, "Visual programming: Compositional visual reasoning without training," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 14 953–14 962.
- [63] S. Yao, J. Zhao, D. Yu, N. Du, I. Shafraan, K. Narasimhan, and Y. Cao, "React: Synergizing reasoning and acting in language models," *arXiv preprint arXiv:2210.03629*, 2022.
- [64] B. Xu, Z. Peng, B. Lei, S. Mukherjee, Y. Liu, and D. Xu, "Rewoo: Decoupling reasoning from observations for efficient augmented language models," *arXiv preprint arXiv:2305.18323*, 2023.
- [65] S. Hong, X. Zheng, J. Chen, Y. Cheng, J. Wang, C. Zhang, Z. Wang, S. K. S. Yau, Z. Lin, L. Zhou *et al.*, "Metagpt: Meta programming for multi-agent collaborative framework," *arXiv preprint arXiv:2308.00352*, 2023.
- [66] D. H. Park, T. Darrell, and A. Rohrbach, "Robust change captioning," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 4623–4632.
- [67] Papineni, Kishore, Roukos, Salim, Ward, Todd, Zhu, and Wei-Jing, "Bleu: A method for automatic evaluation of machine translation," in *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ser. ACL '02. USA: Association for Computational Linguistics, 2002, p. 311–318. [Online]. Available: <https://doi.org/10.3115/1073083.1073135>
- [68] A. Lavie and A. Agarwal, "Meteor: An automatic metric for mt evaluation with high levels of correlation with human judgments," in *Proceedings of the Second Workshop on Statistical Machine Translation*, ser. StatMT '07. USA: Association for Computational Linguistics, 2007, p. 228–231.
- [69] Lin and C. Yew, "Rouge: A package for automatic evaluation of summaries," in *Proceedings of the Workshop on Text Summarization Branches Out (WAS 2004)*, 2004.
- [70] R. Vedantam, C. L. Zitnick, and D. Parikh, "Cider: Consensus-based image description evaluation," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 4566–4575.
- [71] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
- [72] Y. Qiu, S. Yamamoto, K. Nakashima, R. Suzuki, K. Iwata, H. Kataoka, and Y. Satoh, "Describing and localizing multiple changes with transformers," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 1951–1960.