Describing and Localizing Multiple Changes with Transformers

Yue Qiu\textsuperscript{1,*}, Shintaro Yamamoto\textsuperscript{1,2,*}, Kodai Nakashima\textsuperscript{1}, Ryota Suzuki\textsuperscript{1}, Kenji Iwata\textsuperscript{1}, Hirokatsu Kataoka\textsuperscript{1}, Yutaka Satoh\textsuperscript{1}

\textsuperscript{1}National Institute of Advanced Industrial Science and Technology (AIST), \textsuperscript{2}Waseda University

\{qiu.yue, yamamoto.shintaro, nakashima.kodai, ryota.suzuki, kenji.iwata, hirokatsu.kataoka, yu.satou\}@aist.go.jp

Abstract

Change captioning tasks aim to detect changes in image pairs observed before and after a scene change and generate a natural language description of the changes. Existing change captioning studies have mainly focused on a single change. However, detecting and describing multiple changed parts in image pairs is essential for enhancing adaptability to complex scenarios. We solve the above issues from three aspects: (i) We propose a simulation-based multi-change captioning dataset; (ii) We benchmark existing state-of-the-art methods of single change captioning on multi-change captioning; (iii) We further propose Multi-Change Captioning transformers (MCCFormers) that identify change regions by densely correlating different regions in image pairs and dynamically determines the related change regions with words in sentences. The proposed method obtained the highest scores on four conventional change captioning evaluation metrics for multi-change captioning. Additionally, our proposed method can separate attention maps for each change and performs well with respect to change localization. Moreover, the proposed framework outperformed the previous state-of-the-art methods on an existing change captioning benchmark, CLEVR-Change, by a large margin (+6.1 on BLEU-4 and +9.7 on CIDEr scores), indicating its general ability in change captioning tasks. The code and dataset are available at the project page\textsuperscript{1}.

1. Introduction

Detecting and describing the changed parts in scenes at different times is essential in various scenarios, such as urbanization analysis [1, 2, 3], resource management [4, 5, 6, 7, 8], updating street-view maps for navigation [9, 10], damage detection [11, 12], video surveillance [13], and robotic applications [14, 15]. Recently, Jhamtani and Berg-Kirkpatrick [13] proposed the change captioning task to describe changes from image pairs of before and after scene changes. Describing change is useful for extracting semantic contents and conveying information to humans.

Several methods have been proposed for the change captioning [13, 16, 17, 18, 19]. Most existing works focus on describing a single change. As a practical matter, multiple changes could be manifested within an image pair. Jhamtani and Berg-Kirkpatrick [13] studied scene change captioning with multiple changes; however, they addressed the problem with a known number of changes which is not provided in real-world problems. Practically, detecting and describing scene changes without prior information regarding changes is more useful in terms of providing information to users. We address multi-change captioning, where changed regions are localized and language descriptions of scene changes are generated from pair of images with an unknown number of changes, as shown in Figure 1.

Change captioning requires capturing relationships between image pairs, localizing changed regions, and gen-

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure1.png}
\caption{Given two images of a scene observed before and after multiple changes (the first column), we generate change captioning for each scene change along with attention maps (the second and third columns) indicating the region of changed objects.}
\end{figure}
erating language descriptions. In this work, we introduce a simple but effective framework Multi-Change Captioning transformers (MCCFormers) based on encoder-decoder transformer [20] which performs well in natural language processing. The encoder transformer captures relationships between local regions in two images to detect scene changes. Then, the decoder transformer attends over changed regions and generates language descriptions of the changes. In contrast to existing methods that generate a static attention map [16, 17], the decoder transformer changes the spatial attention for each generated word. Consequently, the decoder transformer can distinguish between different changes and avoid confusing them for one another.

To evaluate multi-change captioning and localization ability, we build a novel CLEVR-Multi-Change dataset consisting of image pairs containing multiple changes, change captions, and bounding boxes of the changed region. We compare the proposed MCCFormers model with several state-of-the-art methods under the multi-change setup. The experimental results show that the proposed method performed well in both change captioning and localization.

The contributions of our work are three-fold: (i) We address a novel task of multi-change captioning and propose a dataset for this task where multiple changes exist in before-and after-change images and the number of changes is unknown; (ii) We propose MCCFormers, which consists of encoder-decoder transformers that capture relationships between images and densely correlates image regions with words; (iii) The proposed MCCFormers outperforms existing methods in terms of four conventional image captioning evaluation metrics and shows promising ability on localization for multi-sentence change captioning.

2. Related Work

2.1. Change Detection

Change detection from scenes captured from different moments has been studied in various research fields. [14, 15, 21] discussed change detection from indoor scenes. There are also existing studies which discuss change detection for disaster management [11, 12], resource monitoring [6, 7], and vehicle navigation [9]. Among existing studies, [14, 15, 21] proposed rule-based methods for detecting changed parts from a set of 3-D maps. [6, 7, 9, 11, 12] discuss generating pixel-level maps for indicating the changed region between image pairs. Instead of change detection, we address localizing and describing changes.

2.2. Image Captioning

Image captioning is a well studied topic at the intersection of computer vision and natural language processing. Vinyals et al. [22] proposed encoder-decoder architecture where an encoder extracts image features and a decoder generates a description of an image. Xu et al. [23] introduced the attention mechanism to align each word and relevant region in an image. Inspired by the human visual system, Anderson et al. proposed a combined bottom-up and top-down attention mechanism [24]. Following the success of transformers [20] in natural language processing, transformer-based approaches have been introduced for image captioning [25, 26, 27]. Different from image captioning for a single image, we address change captioning which requires capturing the relationship between two images.

2.3. Change Captioning

Several studies have focused on change captioning which describes a change between two images from different moments. The Spot-the-Diff dataset which consists of 13,192 scene change image pairs was constructed by Jhamtani and Berg-Kirkpatrick [13]. Each image pair has 1.86 change description sentences on average. However, they addressed the problem with a known number of changes. By contrast, we study the multi-change captioning task in which the number of scene changes is not given. CLEVR-Change dataset was introduced by Park et al. [16] to overcome several limitations of the Spot-the-Diff dataset including lack of viewpoint change and localization ground truth. The authors of [28] and [29] discussed change captioning from image pairs observed from multiple viewpoints. This work addresses multi-change captioning with an unknown number of changes and we develop CLEVR-Multi-Change dataset to evaluate localization ability as well as captioning.

Jhamtani and Berg-Kirkpatrick [13] proposed DDLA which computes a pixel-level difference between image pairs, limiting the ability for situations with viewpoint changes. By contrast, DUDA [16] utilizes feature-level differences to enhance the robustness to viewpoint change. Similarly, the Siamese difference captioning model was proposed by Oluwasanmi et al. [18, 19]. M-VAM [17] separates viewpoint changes from semantic changes by evaluating the similarity of different patches of image pairs. Spatial attentions used in DUDA and M-VAM are static, which limits their ability to distinguish different changes.

In this work, we build transformer-based encoder-decoder models for multi-change captioning. The encoder transformer computes patch-level similarity with multi-head attention which captures different types of changes between image pairs. The decoder transformer performs multi-head attention over image patches from the encoder, which captures the relation between generated words and image regions and thus can distinguish different changes.

3. CLEVR-Multi-Change Dataset

Existing change captioning studies mainly focus on single changes. However, identifying and distinguishing multiple change regions simultaneously manifested in image
After was replaced by a small red rubber sphere. (replace)

e. We consider predicting all sentences as a single sequence such as $(w_1^i, \ldots, w_M^i, <SEP>, w_1^f \ldots)$. In our proposed framework, spatial attention is dynamically associated with each word and therefore localization of each change can be computed by averaging the attention maps of each word.

**Preliminary Study.** We evaluated a previous state-of-the-art method, DUDA, on this dataset and one example result is shown in Figure 3. Although DUDA can determine the changed region, it is confused by multiple changes and generates change captions which are only partially correct. For example, DUDA generated “The small green rubber cube was replaced by a large purple rubber cube. The small red rubber cylinder is missing. The large gray rubber sphere has disappeared.”
A dense correlation between different image regions of before- and after-change images is necessary for identifying changes. Furthermore, to distinguish and generate captions for each change, the correlation between change regions and sentences is critical.

4. Approach

Figure 4 shows the proposed Multi-Change Captioning transformers (MCCFormers). Given two images ($I_{bef}$ and $I_{aft}$) of before and after multiple changes, MCCFormers generates a paragraph of descriptions of changes in image pairs. Following existing methods like DUDA [16] and M-VAM [17], we first extract image features $f_{bef}$ and $f_{aft}$ using the CNN structure. We then feed the features to the transformer-based encoder and decoder model. A transformer encoder densely correlates each image patch of before- and after-change image pairs and a decoder further correlates each word with image patches for generating descriptions of multiple changes.

4.1. Change Encoder

It is necessary to distinguish and separate different change regions in scenes involving multiple changes, requiring a dense correlation of different regions between image pairs. To obtain the relationships of different image patches in image pairs, a mechanism to compare and correlate each image patch between image pairs is required. M-VAM correlates feature pairs by introducing an inner production operation of features. Compared with inner-production, the multi-head attention mechanism introduced in transformer-based encoders computes multiple types of attentions to correlate different patches. Thus, we consider adopting transformer-based encoders.

Different from recent transformer-based models for computer vision tasks such as DETR [31] which takes a single image as its input, change captioning tasks feed two images. Given image feature pairs $f_{bef}$ and $f_{aft}$ with dimension $\mathbb{R}^{W \times H \times D}$ (where W, H, and D are, respectively, the width, height, and channel of features), we consider two variants of encoder: Multi-Change Captioning transformers-Dual (MCCFormers-D) and Multi-Change Captioning transformers-Single (MCCFormers-S) (Figure 4 (b-i) and (b-ii), respectively). For both variants, we first transform $f_{bef}$ and $f_{aft}$ to $f'_{bef}$ and $f'_{aft}$ with dimension $\mathbb{R}^{W \times H \times d_{encoder}}$. To accomplish this, we use linear transformation and add a position embedding as follows:

$$f'(x, y) = W_{lt}f(x, y) + b_{lt} + pos(x, y)$$  (1)

where $W_{lt}$ and $b_{lt}$ are learnable parameters of linear transformation and $pos(x, y)$ is learnable position embedding.

MCCFormers-D. In this variant, we use two transformer encoders with shared weights. To capture relevance between local regions of two images, we employ the co-attention mechanism [32]. In contrast to the original co-attention mechanism which takes linguistic tokens and object proposals from an image as input, we consider a set of patches of before and after images as input. Given two feature maps from before and after images $f'_{bef}$ and $f'_{aft}$, we consider that the query feature is either from before- or after-change images and the key and value feature is from the other one. After processing with the encoder with $N_e$ layers, we concatenate the output from features $g_{bef}$ and
\[ g_{\text{aft}} \] over feature dimension as \( g \in \mathbb{R}^{W \times H \times 2d_{\text{encoder}}} \).

**MCCFormers-S.** Different from MCCFormers-D, we use MCCFormers-S to capture image patch relationships among both inter- and intra-image pairs. We first concatenate \( f'_{\text{bef}} \) and \( f'_{\text{aft}} \) to \( f'' \in \mathbb{R}^{2W \times H \times d_{\text{encoder}}} \). We then pass \( f'' \) to the standard transformer structure, which is similar to the BERT model [33] that takes a sequence of two sentences as input. Following MCCFormers-D, the feature maps are converted to \( g \in \mathbb{R}^{W \times H \times 2d_{\text{encoder}}} \). Compared with MCCFormers-D, which only considers relevance between image pairs, this structure also captures image patches’ relevance within both before- and after-change images.

### 4.2. Paragraph Decoder

In the multi-change captioning task, due to the coexistence of multiple changes, it is critical to distinguish different change regions and dynamically attend to different regions during the generation of different sentences. Transformer decoders accomplish this by attending to information from different patches during the generation process.

Therefore, we adopt a standard transformer decoder for generating captions. We first use a word embedding layer to transfer input sentences and add a learnable position embedding. Next, the sentence features are processed through a masked self-attention and feed-forward network. The cross-attention between a sentence and the encoder’s output features is then computed and further processed by a feed-forward layer. The decoder layer is iterated for \( N_d \) layers.

The transformer decoder computes attention over image features for each word during the sentence generation process. Thus, image attention can be computed for every sentence by averaging the attention maps of each word in a sentence. In contrast, DUDA and M-VAM compute a single spatial attention map for an entire paragraph.

### 4.3. Learning Process

From the input of images (\( I_{\text{bef}} \) and \( I_{\text{aft}} \)) observed before and after scene changes, the decoder generates a word sequence with length \( T \). We denote the target sequence as \((w^*_1, ..., w^*_T)\). We adopt a cross-entropy loss for network training, where \( \theta \) indicates the learnable parameter:

\[
L_{XE} = \sum_{t=1}^{T} -\log(p_{\theta}(w^*_t|(w^*_1, ..., w^*_{t-1}), I_{\text{bef}}, I_{\text{aft}})) \tag{2}
\]

### 5. Experiments

#### 5.1. Experimental Setup

**Experiments and Datasets.** We conducted experiments on both multi-change and single change setups. We also implemented DUDA and M-VAM without models modification compared to the models proposed in original papers and evaluated the models performance on paragraph generation. We also report the performance of the methods on two previous datasets i.e., Spot-the-Diff (containing multi- and single change setups, where we sampled instances within four changes for the multi-change setup) and the CLEVR-Change dataset (single change).

**Evaluation Metrics.** We adopt conventional image captioning and change captioning evaluation metrics for performance comparison: BLEU-4 [34], CIDEr [35], METEOR [36], and SPICE [37]. These metrics evaluate the similarity of generated sentences with ground truth from different aspects. We also evaluated the accuracy in terms of the number of sentences in multi-sentence generation. We compute accuracy which measures whether the number of sentences of the generated sequence is correct and mean absolute error (MAE) to evaluate the difference in the number of sentences between ground truth and generated sequence. We prepared five ground truth paragraphs with different sentence orders for each image pair. Therefore, the sentence order has less influence on evaluation results. To assess the localization ability in multi-change captioning, we introduce an evaluation metric based on the Pointing Game [38]. We record the bounding boxes for changed objects and transfer the obtained attention map to the original image size with bilinear interpolation. We then select the top-\( K \) pixels with the largest values in attention maps and compute the detected change region (where the top-\( K \) pixels are inside the bounding box of a changed region) over all of the changed regions in the ground truth. The overall accuracy is averaged over all changes contained in the test data. We set \( K \) to 1 for add and delete and 2 for move and replace where the bounding boxes of objects might be different due to the scene change.

**Implementation Details.** Similar to [16, 17], we use ResNet-101 [39] pretrained on the ImageNet dataset [40] to extract image features from images with a \( 224 \times 224 \) resolution. The obtained feature maps dimension is \( 14 \times 14 \times 1024 \). We implemented transformers with two layers and four heads for both encoders and decoders. The dimensions

| Methods | BLEU-4 |
|--------|-------|
| DUDA [16] | 76.1 |
| DUDA Encoder + Transformer Decoder | 79.1 |
| M-VAM [17] | 62.9 |
| M-VAM Encoder + Transformer Decoder | 65.8 |
| MCCFormers-D (concat. over patches) | 80.1 |
| MCCFormers-D (concat. over feature dimension) | 82.3 |
| MCCFormers-S (concat. over patches) | 80.6 |
| MCCFormers-S (concat. over feature dimension) | 83.3 |

Table 3. BLEU-4 evaluation of different methods applied to the CLEVR-Multi-Change dataset. (concat. : concatenation)

---
of input features to encoder $d_{\text{encoder}}$ and decoder $d_{\text{decoder}}$ are 512 and 1024, respectively. For the feedforward network, the dimensions are $4d_{\text{encoder}}$ and $4d_{\text{decoder}}$ for encoders and decoders, respectively. We set the learning rate to 0.0001 and trained models for 40 epochs with the Adam optimizer [41] during all implementations.

**Baselines.** We use DUDA and M-VAM for comparison during experiments and two variants of the transformer network. We set the hidden state dimension for all LSTM structures to 512 in both DUDA and M-VAM (see the supplementary material for details of the implementation of DUDA and M-VAM).

### 5.2. CLEVR-Multi-Change dataset

**Ablation Study.** We evaluated different structure choices of change encoders and decoders (Table 3). DUDA and M-VAM scored 76.1 and 62.9 for BLEU-4, respectively. We removed the sum operation over the spatial region of features (DUDA) and average pooling operations (M-VAM) which are applied before feeding features to decoders. We then replaced the decoders with transformer decoders. The use of transformer decoders improved the performance of both two methods.

Next, we conducted experiments regarding MCCFormers. Before feeding features to the decoder, we concatenate features of before- and after-change images in two ways: concatenation operation over patches (input to decoder: $g \in \mathbb{R}^{2W \times H \times d_{\text{encoder}}}$) and over feature dimension (input to decoder: $g \in \mathbb{R}^{W \times H \times 2d_{\text{encoder}}}$). All methods outperformed previous methods in terms of BLEU-4. MCCFormers-D and MCCFormers-S with concatenation over feature dimension performed the best. In the case of relatively small viewpoint change, patches of the same region in two images are concatenated, which could improve the effectiveness of the concatenation over feature dimension. We will investigate the robustness to viewpoint change (especially larger change) for future work. In the remaining experiments, we use MCCFormers-D and MCCFormers-S with concatenation over feature dimension.

**Sentence Generation.** Experimental results of the different evaluation metrics are shown in Table 4. Both MCCFormers-D and MCCFormers-S outperformed previous methods with respect to all metrics. MCCFormers-S obtained the highest BLEU-4 and outperformed previous methods by +7.2.

For one-change instances, the differences between the proposed methods and previous methods are relatively small. For instances with multiple changes, the two proposed methods exhibited better robustness. The transformer encoder learns a dense correlation among all local regions of change image pairs, and the decoder model further correlates each word with image regions, making the models better at distinguishing different changes.

**Evaluation of Sentence Number Accuracy.** Results concerning sentence number accuracy and MAE are shown in Table 5. Similar to the results in Table 4, the proposed methods obtained higher accuracy for sentence numbers compared to previous methods and achieved promising results for distinguishing changes, with an accuracy of 92.5% for the MCCFormers-D method. The MAE results show that the average errors of the number of sentences generated by all methods are less than 1.

All methods achieved the highest scores for one-change sentence and the MCCFormers-D model obtained 99.4% accuracy. For two-, three- and four-change instances, the accuracy of previous methods was degraded while the two proposed methods show promising stability for scenes with multiple changes.

**Qualitative Results.** We show one example result in Figure 5. This example contains four changes. DUDA predicted three changes and both variants of MCCFormers generated correct sentences in terms of both change number and change contents. In addition, two captions generated by DUDA contain incorrect change types whilst two proposed methods generated correct captions for each change.

DUDA generates a single attention map for each given image pair. Therefore, the network could be strug-

| Methods          | Overall | 1 Change | BLEU-4 [34] | 2 Changes | 3 Changes | 4 Changes | CIDEr [35] | METEOR [36] | SPICE [37] |
|------------------|---------|----------|-------------|-----------|-----------|-----------|------------|-------------|------------|
|                  |         |          | BLEU-4      |           |           |           |            |             |            |
|                  |         |          | 1 Change    | 2 Changes | 3 Changes | 4 Changes |            |             |            |
| DUDA [16]        | 76.1    | 94.7     | 76.3        | 73.1      | 70.9      | 480.1     | 47.4       | 66.6        |
| M-VAM [17]       | 62.9    | 79.0     | 61.7        | 59.9      | 57.9      | 338.1     | 41.3       | 55.9        |
| MCCFormers-D     |         |          |             |           |           |           |            |             |            |
| MCCFormers-S     | 82.3    | 98.4     | 82.9        | 79.2      | 80.6      | 539.3     | 52.1       | 71.7        |

Table 4. Results on the CLEVR-Multi-Change dataset.

| Methods          | All      | 1 change | Accuracy | 2 change | 3 change | 4 change | MAE        |
|------------------|----------|----------|----------|----------|----------|----------|------------|
|                  |          |          |          |          |          |          |            |
| DUDA [16]        | 83.3     | 96.8     | 84.8     | 74.4     | 76.0     | 0.169    |            |
| M-VAM [17]       | 69.9     | 91.2     | 68.4     | 56.5     | 62.0     | 0.317    |            |
| MCCFormers-D     | 92.5     | 99.4     | 94.8     | 81.9     | 93.9     | 0.075    |            |
| MCCFormers-S     | 92.4     | 97.9     | 95.9     | 91.9     | 83.7     | 0.075    |            |

Table 5. Sentence number accuracy (%) and mean absolute error (MAE) on the CLEVR-Multi-Change dataset.
The small blue rubber cube has been moved.

MCCFormers-D: “The small blue rubber cube changed its location.”
MCCFormers-S: “The small blue rubber cube was moved from its original location.”

The small green metal cube changed its location.

MCCFormers-D: “The small green metal cube changed its location.”
MCCFormers-S: “The small green metal cube was moved from its original location.”

There is a new large blue metal sphere.

MCCFormers-D: “A large blue metal sphere shows up.”
MCCFormers-S: “A large blue metal sphere has been added.”

There is no longer a large brown rubber sphere.

MCCFormers-D: “The large brown rubber sphere is no longer there.”
MCCFormers-S: “The large brown rubber sphere is missing.”

DUDA: “The small blue metal cube changed its location.”
“The large brown rubber sphere is in a different location.”
“The small green rubber cube was replaced by a small blue rubber cube.”

Figure 5. Visualization of an example from the CLEVR-Multi-Change dataset. We show the attention maps for the proposed methods and DUDA and generated sentences. Incorrect captions are in red font. We highlighted changed regions in black bounding boxes.

Figure 6. Pointing Game accuracy (%) over different change numbers (left) and types (right) on the CLEVR-Multi-Change dataset.

Among different change types, both methods obtained the highest accuracy for replace change and the lowest accuracy for move change. The move change relates to two image positions, making it challenging for localization.

5.3. Spot-the-Diff Dataset

Spot-the-Diff contains multiple changes within image pairs. We first extract all instances containing one to four changes and report the results of DUDA and MCCFormers in Table 6 (top three rows). For this dataset, MCCFormers obtained comparable results with DUDA. We further show one example in Figure 7. For the example containing two changes, the two methods correctly generated two related

gling to distinguish each changed part from the attention map, limiting its ability in multi-change understanding. MCCFormers-S generated attention maps that tend to focus on unrelated regions as well as changed object regions. This structure refers to the patches of inter- and intra-image pairs, which might weaken the interpretability of attention maps. MCCFormers-D obtained separate attention maps for each sentence and attended to change regions. We refer to the supplementary material for more example results.

Pointing Game Evaluation for Attention Maps. We evaluated the localization accuracy of MCCFormers-D and DUDA (Figure 6). Since DUDA generates a single pair of attention maps for each image pair, we use the same attention map to evaluate each change. MCCFormers-D obtained an overall accuracy of 53.9% for change localization and 40.0% for DUDA. In both methods, localization performance degrades with the increase in change number. DUDA obtained higher localization accuracy for one change and our methods outperformed DUDA for two, three, and four changes, indicating the effectiveness of MCCFormers-D for detecting multiple changes.
Table 6. Results on the Spot-the-Diff dataset.

| Methods          | BLEU-4 | CIDEr | METEOR | SPICE |
|------------------|--------|-------|--------|-------|
| DUDA [16]        | 5.4    | 24.8  | 10.6   | 12.9  |
| MCCFormers-D     | 6.2    | 28.8  | 10.2   | 17.8  |
| MCCFormers-S     | 5.8    | 18.2  | 10.5   | 10.1  |

Table 7. Results on the CLEVR-Change dataset.

| Methods          | BLEU-4 | CIDEr | METEOR | SPICE |
|------------------|--------|-------|--------|-------|
| DUDA [16]        | 47.3   | 112.3 | 33.9   | 24.5  |
| M-VAM [17]       | 50.3   | 114.9 | 37.0   | 30.5  |
| M-VAM + RAF [17] | 51.3   | 115.8 | 37.8   | 30.7  |
| MCCFormers-D     | 52.4   | 121.6 | 38.3   | 26.8  |
| MCCFormers-S     | 57.4   | 125.5 | 41.2   | 32.4  |

Compared to the state-of-the-art method M-VAM.

5.4. CLEVR-Change Dataset

We compare the different methods on the previous single change dataset CLEVR-Change in Table 7. The CLEVR-Change dataset requires understanding object relationships inside each image (e.g., in front of) which are not included in the proposed dataset. Therefore, compared with MCCFormers-D, MCCFormers-S obtained the highest scores as MCCFormers-S can capture relationships between image patches within the same image. MCCFormers outperformed the previous methods on this dataset in terms of most evaluation metrics, indicating the ability of proposed structures to correlate different regions in change image pairs and further connect the change region information with words in sentences.

6. Conclusion

In this paper, we propose a novel multi-change captioning task and CLEVR-Multi-Change dataset for this task. To address the novel task, we proposed a transformer-based framework MCCFormers that densely correlates different image regions in image pairs and words. MCCFormers achieved state-of-the-art performance on both multi- and single-change captioning datasets, indicating the effectiveness of MCCFormers for change captioning tasks.

Acknowledgements

We want to thank Yoshitaka Ushiku, Seito Kasai, Hikaru Ishitsuka, and Tomomi Satoh for their helpful comments during research discussions. This paper is based on results obtained from a project, JPNP20006, commissioned by the New Energy and Industrial Technology Development Organization (NEDO). Computational resource of AI Bridging Cloud Infrastructure (ABCI) provided by National Institute of Advanced Industrial Science and Technology (AIST) was used.
Supplementary Material

This supplementary material provides additional implementation details of methods used in this article, including the proposed methods MCCFormers and previous methods, DUDA, and M-VAM. We also give a more detailed introduction of the CLEVR-Multi-Change dataset, including the details of the caption generation process and additional dataset examples. Additional experimental results on the CLEVR-Multi-Change dataset can be found in the last section of this material.

A. Additional Implementation Details

Feature Concatenation of MCCFormers. Here, we provide more details of the feature concatenation operation used in the Ablation Study of subsection 5.2 and Table 3 in the main paper. The MCCFormers-D (encoder) outputs $g_{bef}$ and $g_{aft}$ with dimension of $\mathbb{R}^{W \times H \times d_{\text{encoder}}}$, respectively. The MCCFormers-S (encoder) outputs a feature map with dimension of $\mathbb{R}^{2W \times H \times d_{\text{encoder}}}$. We then separate the output to $g_{bef}$ and $g_{aft}$ with dimension of $\mathbb{R}^{W \times H \times d_{\text{encoder}}}$. For both two MCCFormers, we consider two ways to concatenate $g_{bef}$ and $g_{aft}$ (Figure 8 (a)): concatenation over patches (Figure 8 (b)) and concatenation over feature dimension (Figure 8 (c)), before feeding features to decoders. The experimental results are given in Table 3 of the main paper.

DUDA. We implemented DUDA based on the code provided by the authors of DUDA. We set the dimension of the encoder and LSTM hidden layer of DUDA to 512.

M-V AM. We implemented M-V AM following the approach introduced in the original paper of M-V AM [17]. For encoder of M-V AM, two scalars in Equation (3) in [17] are learned during training. Regarding the sentence decoder, two LSTM with hidden state dimensions of 512 are trained. The network is trained with cross-entropy loss in an end-to-end manner.

For the implementation of DUDA and M-V AM, we used the same input image features, learning rate, optimizer, learning iteration as the proposed methods introduced in Implementation Details of subsection 5.1 of the main paper.

B. Additional Details on CLEVR-Multi-Change Dataset

Caption Generation. As introduced in section 3 of the main paper, the CLEVR-Multi-Change dataset consists of before- and after-change image pairs and captions that describe changes through language text. We record the change information during the generation of image pairs, including change type and attributes of related objects. The change captions are generated based on recorded change information and pre-defined sentence templates.

All templates used in the CLEVR-Multi-Change dataset are shown in Table 8. The tags “<s> <c> <t> <z>” and “<s1> <c1> <t1> <z1>” in each template are instantiated during caption generation. For example, with the template “A <s> <c> <t> <z> has been added.” and an added object with attributes: small, red, metal, cube, the generated caption would be “A small red metal cube has been added.”

Dataset Examples. We show additional dataset examples in Figure 9 (one-change examples), Figure 10 (two-change examples), Figure 11 (three-change examples), and Figure 12 (four-change examples).
### Caption templates

| Change type | Caption templates |
|-------------|-------------------|
| Add         | “A <s> <c> <t> <z> has been added.” |
|             | “A <s> <c> <t> <z> shows up.” |
|             | “There is a new <s> <c> <t> <z>.” |
|             | “A new <s> <c> <t> <z> is visible.” |
|             | “Someone added a <s> <c> <t> <z>.” |
| Delete      | “The <s> <c> <t> <z> has disappeared.” |
|             | “The <s> <c> <t> <z> is no longer there.” |
|             | “The <s> <c> <t> <z> is missing.” |
|             | “There is no longer a <s> <c> <t> <z>.” |
|             | “Someone removed the <s> <c> <t> <z>.” |
| Move        | “The <s> <c> <t> <z> changed its location.” |
|             | “The <s> <c> <t> <z> is in a different location.” |
|             | “The <s> <c> <t> <z> was moved from its original location.” |
|             | “The <s> <c> <t> <z> has been moved.” |
|             | “Someone changed location of the <s> <c> <t> <z>.” |
| Replace     | “The <s> <c> <t> <z> was replaced by a <s1> <c1> <t1> <z1>." |
|             | “A <s1> <c1> <t1> <z1> replaced the <s> <c> <t> <z>.” |
|             | “A <s1> <c1> <t1> <z1> is in the original position of <s> <c> <t> <z>." |
|             | “The <s> <c> <t> <z> gave up its position to a <s1> <c1> <t1> <z1>." |
|             | “Someone replaced the <s> <c> <t> <z> with a <s1> <c1> <t1> <z1>.” |

Table 8. Caption templates used in CLEVR-Multi-Change dataset. `<s>`, `<s1>`: size; `<c>`, `<c1>`: color; `<t>`, `<t1>`: material; `<z>`, `<z1>`: shape.

### BLEU-4 scores of different network designs

| Models      | Layers | Heads | BLEU-4 (Overall) |
|-------------|--------|-------|------------------|
| MCCFormers-D| 1      | 1     | 59.0             |
|             |        | 2     | 65.8             |
|             |        | 4     | 71.0             |
|             |        | 8     | 76.8             |
|             | 2      | 1     | 81.4             |
|             |        | 2     | 81.2             |
|             |        | 4     | 82.3             |
|             |        | 8     | **82.5**         |
|             | 4      | 1     | 60.3             |
|             |        | 2     | 59.8             |
|             |        | 4     | 64.8             |
|             |        | 8     | 77.2             |
| MCCFormers-S| 1      | 1     | 58.1             |
|             |        | 2     | 64.0             |
|             |        | 4     | 75.8             |
|             |        | 8     | 79.9             |
|             | 2      | 1     | **80.0**         |
|             |        | 2     | 82.2             |
|             |        | 4     | **83.3**         |
|             |        | 8     | 83.0             |

Table 9. BLEU-4 evaluation of different network designs (Layers and Heads) of MCCFormers on CLEVR-Multi-Change dataset.

### Additional Experimental Results on CLEVR-Multi-Change Dataset

#### Additional Visualization of Examples

We show three examples with two changes on the CLEVR-Multi-Change dataset in Figure 13, Figure 14, and Figure 15. For the first two examples (Figure 13 and Figure 14), both two MCCFormers correctly generated two related sentences, while for the second example, both two MCCFormers generated a sentence with incorrect object shapes. For the third example (Figure 15), MCCFormers-D only generated one sentence, while the attention maps show that the model captured two change regions.

Overall, MCCFormers-D obtained attention maps that attend to related change regions while the MCCFormers-S tends to attend to related change regions as well as unrelated regions.

#### Allocations of Network Design of MCCFormers (Layers and Heads)

The overall BLEU-4 scores of MCCFormers-D and MCCFormers-S with different layers and heads are shown in Table 9. We found that models with two layers and four heads perform relatively well for both two methods among different network designs. Therefore, we used MCCFormers-D and MCCFormers-S with two layers and four heads in experiments described in the main paper.
Figure 9. One-change examples from the CLEVR-Multi-Change dataset. The changed objects are highlighted by rectangles with the same color as the associated change captions.

**Ground Truth 1:** A large gray metal cylinder is visible.
**Ground Truth 2:** There is a new large gray metal cylinder.
**Ground Truth 3:** Someone added a large gray metal cylinder.
**Ground Truth 4:** A large gray metal cylinder has been added.
**Ground Truth 5:** A large gray metal cylinder shows up.

Figure 10. Two-change examples from the CLEVR-Multi-Change dataset. The changed objects are highlighted by rectangles with the same color as the associated change captions.

**Ground Truth 1:** Someone changed location of the large green metal cylinder.
**Ground Truth 2:** The large green metal cylinder changed its location.
**Ground Truth 3:** The large green metal cylinder was moved from its original location.
**Ground Truth 4:** The large green metal cylinder has been moved.
**Ground Truth 5:** The large green metal cylinder is in a different location.

**Ground Truth 1:** The small cyan rubber sphere gave up its position to a small blue metal sphere. A small red metal cube shows up.
**Ground Truth 2:** A small red metal cube shows up. The small cyan rubber sphere was replaced by a small blue metal sphere.
**Ground Truth 3:** A small red metal cube shows up. A small blue metal sphere replaced the small cyan rubber sphere.
**Ground Truth 4:** The small cyan rubber sphere was replaced by a small blue metal sphere. A new small red metal cube is visible.
**Ground Truth 5:** There is a new small red metal cube. A small blue metal sphere is in the original position of small cyan rubber sphere.

**Ground Truth 1:** A small blue metal cylinder replaced the small green metal cylinder. The large gray metal sphere changed its location.
**Ground Truth 2:** The small green metal cylinder was replaced by a small blue metal cylinder. The large gray metal sphere has been moved.
**Ground Truth 3:** The large gray metal sphere is in a different location. The small green metal cylinder gave up its position to a small blue metal cylinder.
**Ground Truth 4:** Someone replaced the small green metal cylinder with a small blue metal cylinder. The large gray metal sphere has been moved.
**Ground Truth 5:** Someone changed location of the large gray metal sphere. The small green metal cylinder gave up its position to a small blue metal cylinder.
Figure 11. Three-change examples from the CLEVR-Multi-Change dataset. The changed objects are highlighted by rectangles with the same color as the associated change captions.

**Ground Truth 1:** A small blue rubber cylinder replaced the small purple rubber cylinder. The large red metal cube has been moved. A new large gray metal cylinder is visible.

**Ground Truth 2:** There is a new large gray metal cylinder. The small purple rubber cylinder gave up its position to a small blue rubber cylinder. The large red metal cube was moved from its original location.

**Ground Truth 3:** A large gray metal cylinder shows up. The large red metal cube is in a different location. A small blue rubber cylinder is in the original position of small purple rubber cylinder.

**Ground Truth 4:** The large red metal cube is in a different location. Someone replaced the small purple rubber cylinder with a small blue rubber cylinder. Someone added a large gray metal cylinder.

**Ground Truth 5:** Someone added a large gray metal cylinder. Someone changed location of the large red metal cube. A small blue rubber cylinder replaced the small purple rubber cylinder.

**Ground Truth 1:** There is no longer a large blue rubber sphere. A new large blue rubber cube is visible. The small gray metal cube was replaced by a large gray rubber cylinder.

**Ground Truth 2:** Someone replaced the small gray metal cube with a large gray rubber cylinder. Someone removed the large blue rubber sphere. A new large blue rubber cube is visible.

**Ground Truth 3:** There is a new large blue rubber cube. The small gray metal cube was replaced by a large gray rubber cylinder. The large blue rubber sphere is no longer there.

**Ground Truth 4:** Someone added a large blue rubber cube. Someone replaced the small gray metal cube with a large gray rubber cylinder. The large blue rubber sphere is no longer there.

**Ground Truth 5:** The small gray metal cube was replaced by a large gray rubber cylinder. There is no longer a large blue rubber sphere. A large blue rubber cube shows up.
Figure 12. Four-change examples from the CLEVR-Multi-Change dataset. The changed objects are highlighted by rectangles with the same color as the associated change captions.

**Ground Truth 1**: A small purple metal cube shows up. A small cyan rubber sphere shows up. Someone replaced the large brown rubber cylinder with a large blue rubber cube. The large yellow rubber sphere gave up its position to a small red rubber sphere.

**Ground Truth 2**: The large yellow rubber sphere was replaced by a small red rubber sphere. A small purple metal cube shows up. Someone replaced the large brown rubber cylinder with a large blue rubber cube. A small cyan rubber sphere has been added.

**Ground Truth 3**: A small purple metal cube has been added. A large blue rubber cube replaced the large brown rubber cylinder. A small red rubber sphere is in the original position of large yellow rubber sphere. A small cyan rubber sphere has been added.

**Ground Truth 4**: Someone added a small purple metal cube. Someone replaced the large yellow rubber sphere with a small red rubber sphere. The large brown rubber cylinder gave up its position to a large blue rubber cube. A small cyan rubber sphere has been added.

**Ground Truth 5**: The large yellow rubber sphere was replaced by a small red rubber sphere. A large blue rubber cube replaced the large brown rubber cylinder. Someone added a small purple metal cube. A small cyan rubber sphere shows up.
Figure 13. Visualization of an example from the CLEVR-Multi-Change dataset. We highlighted changed regions in black rectangles.

**Ground Truth:** The small cyan metal sphere is in a different location.

**MCCFormers-D:** The small cyan metal sphere changed its location.

**MCCFormers-S:** The small cyan metal sphere was moved from its original location.

**Ground Truth:** There is no longer a large brown metal cube.

**MCCFormers-D:** The large brown metal cube is missing.

**MCCFormers-S:** The large brown metal cube is missing.
Figure 14. Visualization of an example from the CLEVR-Multi-Change dataset. Incorrect captions are in red font. We highlighted changed regions in black rectangles.

**Ground Truth:** The small green rubber cube is missing.
**MCCFormers-D:** The small green rubber cylinder is no longer there.
**MCCFormers-S:** The small green rubber cylinder is missing.

**Ground Truth:** Someone replaced the small red metal sphere with a small blue rubber cube.
**MCCFormers-D:** A small blue rubber cube replaced the small red metal sphere.
**MCCFormers-S:** The small red metal sphere gave up its position to a small blue rubber cube.
Figure 15. Visualization of an example from the CLEVR-Multi-Change dataset. Incorrect captions are in red font. We highlighted changed regions in black rectangles.

**Ground Truth:** The small brown metal cube has disappeared.  
**MCCFormers-D:** (No sentence was generated.)  
**MCCFormers-S:** The small brown metal cube is missing.

**Ground Truth:** The small green rubber cube gave up its position to a small green metal sphere.  
**MCCFormers-D:** The small green rubber cube gave up its position to a small green metal sphere.  
**MCCFormers-S:** A small green metal sphere replaced the small green rubber cube.
References

[1] Qiaofeng Zhang, J Wang, X Peng, P Gong, and P Shi. Urban built-up land change detection with road density and spectral information from multi-temporal landsat tm data. *International Journal of Remote Sensing*, 23(15):3057–3078, 2002.

[2] Limin Yang, George Xian, Jacqueline M Klaver, and Brian Deal. Urban land-cover change detection through sub-pixel imperviousness mapping using remotely sensed data. *Photogrammetric Engineering and Remote Sensing*, 69(9):1003–1010, 2003.

[3] Ibrahim Rizk Hegazy and Mosbeh Rashed Kaloop. Monitoring urban growth and land use change detection with gis and remote sensing techniques in daqahliya governorate egypt. *International Journal of Sustainable Built Environment*, 4(1):117–124, 2015.

[4] Pol R Coppin and Marvin E Bauer. Digital change detection in forest ecosystems with remote sensing imagery. *Remote Sensing Reviews*, 13(3-4):207–234, 1996.

[5] Robert E Kennedy, Philip A Townsend, John E Gross, Warren B Cohen, Paul Bolstad, YQ Wang, and Phyllis Adams. Remote sensing change detection tools for natural resource managers: Understanding concepts and tradeoffs in the design of landscape monitoring projects. *Remote Sensing of Environment*, 113(7):1382–1396, 2009.

[6] Salman H Khan, Xuming He, Fatih Porikli, and Mohammed Bennamoun. Forest change detection in incomplete satellite images with deep neural networks. *IEEE Transactions on Geoscience and Remote Sensing*, 55(9):5407–5423, 2017.

[7] Sudipan Saha, Francesca Bovolo, and Lorenzo Bruzzone. Unsupervised deep change vector analysis for multiple-change detection in vhr images. *IEEE Transactions on Geoscience and Remote Sensing*, 57(6):3677–3693, 2019.

[8] Rodrigo Caye Daudt, Bertr Le Saux, and Alexandre Boulch. Fully convolutional siamese networks for change detection. In *2018 IEEE International Conference on Image Processing (ICIP)*, pages 4063–4067, 2018.

[9] Pablo F Alcantarilla, Simon Stent, German Ros, Roberto Arroyo, and Riccardo Gherardi. Street-view change detection with deconvolutional networks. *Autonomous Robots*, 42(7):1301–1322, 2018.

[10] Ken Sakurada, Takayuki Okatani, and Koichiro Deguchi. Detecting changes in 3d structure of a scene from multi-view images captured by a vehicle-mounted camera. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 137–144, 2013.

[11] Ken Sakurada and Takayuki Okatani. Change detection from a street image pair using cnn features and superpixel segmentation. In *Proceedings of the British Machine Vision Conference (BMVC)*, pages 1–12, 2015.

[12] Aito Fujita, Ken Sakurada, Tomoyuki Imaiizumi, Riho Ito, Shuhei Hikosaka, and Ryosuke Nakamura. Damage detection from aerial images via convolutional neural networks. In *2017 International Conference on Machine Vision Applications (MVA)*, pages 5–8, 2017.

[13] Harsh Jhamtani and Taylor Berg-Kirkpatrick. Learning to describe differences between pairs of similar images. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4024–4034, 2018.

[14] Evan Herbst, Peter Henry, Xiaofeng Ren, and Dieter Fox. Toward object discovery and modeling via 3-d scene comparison. In *2011 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2623–2629, 2011.

[15] Rareq Ambruş, Nils Bore, John Folkesson, and Patrik Jensfelt. Meta-rooms: Building and maintaining long term spatial models in a dynamic world. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1854–1861, 2014.

[16] Dong Huk Park, Trevor Darrell, and Anna Rohrbach. Robust change captioning. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 4624–4633, 2019.

[17] Xiangxi Shi, Xu Yang, Juxiang Gu, Shafiq Joty, and Jianfei Cai. Finding it at another side: A viewpoint-adapted matching encoder for change captioning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 574–590, 2020.

[18] Ariyo Oluwasanmi, Enoch Frimpong, Muhammad Umar Aftab, Edward Y Baagerye, Zhiqiang Qin, and Kifayat Ul-lah. Fully convolutional captionnet: Siamese difference captioning attention model. *IEEE Access*, 7:175929–175939, 2019.

[19] Ariyo Oluwasanmi, Muhammad Umar Aftab, Eatedal Al-abdulkreem, Bulbula Kumeda, Edward Y Baagerye, and Zhiqiang Qin. Captionnet: Automatic end-to-end siamese difference captioning model with attention. *IEEE Access*, 7:106773–106783, 2019.

[20] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 6000–6010, 2017.

[21] Maciej Halber, Yifei Shi, Kai Xu, and Thomas Funkhouser. Rescan: Inductive instance segmentation for indoor rgbd scans. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 2541–2550, 2019.

[22] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show, attend and tell: Neural image caption generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3156–3164, 2015.

[23] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 2048–2057, 2015.
[24] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 6077–6086, 2018. 2

[25] Marcella Cornia, Matteo Stefanini, Lorenzo Baraldi, and Rita Cucchiara. Meshed-memory transformer for image captioning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 10578–10587, 2020. 2

[26] Guang Li, Linchao Zhu, Ping Liu, and Yi Yang. Entangled transformer for image captioning. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), pages 8928–8937, 2019. 2

[27] Simao Herdade, Armin Kappeler, Kofi Boakye, and Joao Soares. Image captioning: Transforming objects into words. In Advances in Neural Information Processing Systems (NeurIPS), volume 32, 2019. 2

[28] Yue Qiu, Yutaka Satoh, Ryota Suzuki, Kenji Iwata, and Hirokatsu Kataoka. 3d-aware scene change captioning from multiview images. IEEE Robotics and Automation Letters, 5(3):4743–4750, 2020. 2

[29] Yue Qiu, Yutaka Satoh, Ryota Suzuki, Kenji Iwata, and Hirokatsu Kataoka. Indoor scene change captioning based on multimodality data. Sensors, 20(17):4761, 2020. 2

[30] Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 2901–2910, 2017. 3

[31] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In Proceedings of the European Conference on Computer Vision (ECCV), pages 213–229, 2020. 4

[32] Jiason Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In Advances in Neural Information Processing Systems (NeurIPS), volume 32, 2019. 4

[33] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL), pages 4171–4186, 2019. 5

[34] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL), pages 311–318, 2002. 5, 6

[35] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. arXiv:1412.6980, 2014. 6