Different Changes Require Different Reasoning: Change-Type-Specialized Experts for Robust Change Captioning

Jiyoung Park*, InJae Oh*, and Jung Uk Kim†

Kyung Hee University, Yong-in, South Korea
{jy0117, seanoh, ju.kim}@khu.ac.kr

Abstract. Change captioning is the task of generating natural language descriptions that explain the changes between a pair of images. Although different change types (*e.g.,* color shifts, object additions) exhibit distinct visual cues and require specialized reasoning processes, existing methods often overlook these distinctions. To address this limitation, we propose Multi-Expert Diagnosis for Image Change (MEDIC), a novel framework that introduces change-type awareness by explicitly modeling change categories. We build our MEDIC as a memory network to dynamically retrieve type-relevant visual patterns conditioned on the input. This design allows each expert to flexibly capture diverse variations within each change type and focus on the most informative cues for its designated change type. By routing inputs through type-specialized experts and learning dedicated representations for each change category, MEDIC generates more precise and type-aware change descriptions. Extensive experiments demonstrate that proposed MEDIC consistently outperforms across diverse and challenging datasets. The code is available at https://github.com/VisualAIKHU/MEDIC.

Keywords: Change Captioning · Memory Network · Mixture-of-Experts

1 Introduction

In many vision-based applications, understanding how a scene evolves over time is essential. Change captioning [15, 35] addresses this need by generating natural language descriptions that explain the visual differences between two images taken at different times, enabling intuitive communication of what has changed and how it appears. Such capability is particularly valuable in real-world scenarios such as surveillance [11, 15] and medical diagnostics [26], where precise and reliable interpretation of visual changes is critical.

Research in change captioning has progressed through several stages: early pixel- or feature-level differencing with simple encoder-decoder captioning [15, 35], improved alignment and robustness via cross-view matching, distractor suppression, and structure-aware modeling [44–46], and more recent generative and

* Equal contribution. † Corresponding author.

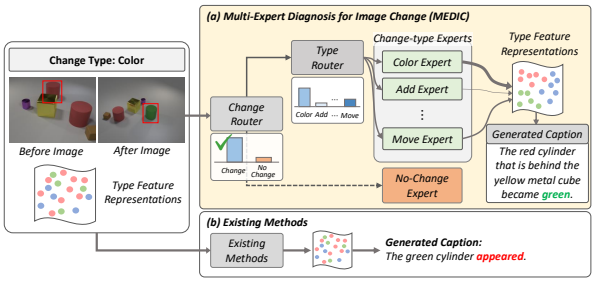


Fig. 1: Conceptual comparison between (a) MEDIC and (b) existing methods using a color-change example.

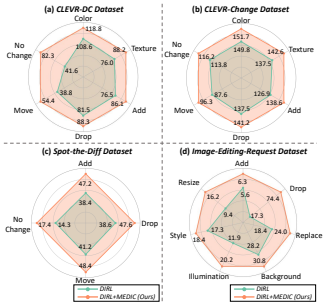


Fig. 2: CIDEr comparisons across change types.

reasoning-centric approaches that incorporate contrastive or diffusion objectives, adversarial hard negatives, MLLM/VLM-based reasoning, and textual compositional reasoning [3, 5, 27, 36, 56].

Despite these advances, most prior methods still assume that diverse visual changes can be addressed through a single reasoning process. Color change depends on subtle pixel-level intensity variations [30], and object addition requires background separation and object-level reasoning [31]. Nevertheless, existing methods are constrained to process color variations, object additions, and spatial movements through a single pipeline even though each type relies on different visual evidence. As shown in Fig. 1 (b), when these heterogeneous cues are processed within a single unified pipeline, the model often misinterprets the nature of the change-type (*e.g.,* describing a red cylinder that became green (color change-type) as 'the green cylinder appeared' (add change-type) rather than 'the red cylinder that is behind the yellow metal cube became green'). They overlook the fact that different change types rely on distinct visual cues.

To address this issue, we introduce **MEDIC** (**M**ulti-**E**xpert **D**iagnosis for **I**mage **C**hange), a new framework that models change captioning as a type-aware process. As shown in Fig. 1 (a), MEDIC employs a two-stage routing mechanism inspired by mixture-of-experts (MoE) [25]: a change router first identifies whether a meaningful change is present, and a type router then assigns the input to a specialized expert dedicated to the corresponding change type. We design each expert as a memory network that retrieves diverse type-relevant visual patterns, enabling adaptive and discriminative reasoning beyond conventional feed-forward experts [25]. To support type-aware specialization, we introduce three training losses—routing loss, expert consistency loss, and expert disentangle loss—that collectively promote correct expert assignment, enforce type-specific representation learning, and maintain clear separation among expert representations. As a result, MEDIC consistently outperforms existing state-of-the-art, code-released methods such as DIRL [44] across all change types and datasets (see Fig. 2), showing the effectiveness of explicitly modeling change types in change captioning.

The main contributions of our work are as follows:

– We introduce **MEDIC**, a novel type-aware change captioning framework that addresses the limitations that does not consider change types. By using

specialized experts for each change category, MEDIC prevents reasoning confusion and enables type-aware interpretation.
– We design specialized memory-based experts that dynamically retrieve input-dependent patterns within each change type, resulting in consistent performance gains across all datasets and change types.
– We develop three training losses—routing loss, expert consistency loss, and expert disentangle loss—to learn type-specific representations while encouraging clear separation across change categories.

2 Related Works

2.1 Change Captioning

Change captioning generates descriptions of semantic differences between image pairs. Early methods used aligned pairs [15] or feature subtraction [35], but struggled with distractors such as viewpoint shifts. Recent methods improve robustness through relation-aware difference modeling and distractor suppression: SRDRL [49] and R^3Net [48] learn semantic or relation-embedded differences, SCORER [46] and DIRL [44] use contrastive and relational learning, SMART [45], NCT [43], and I3N [54] exploit syntactic, neighborhood, or cross-view cues, and MURAT [53] and CHEERS [28] further improve multi-grained aggregation and change-entity-guided disentanglement.

Despite these advances, most methods still use a unified, type-agnostic pipeline. They do not explicitly adapt visual difference reasoning to the change type MEDIC addresses this limitation with type-specific experts.

2.2 Mixture of Experts

The Mixture of Experts (MoE) framework routes inputs to specialized subnetworks, or 'experts,' each handling a subset of the input space [14]. Sparsely-Gated MoE [39] enabled scalable training by activating only a few experts per input. This design was later adopted in large-scale models such as GShard [24] and Switch Transformers [7] to boost capacity without proportional compute.

Beyond language modeling, MoE has been effective in vision/multimodal learning. V-MoE [38] introduced sparsely activated experts into Vision Transformers, and later works extended MoE to structured domains like scene decomposition [55] and visual tracking [4], validating its strength in handling diverse inputs and enabling specialization. Unlike conventional MoE designs that rely on shared representations between experts, MEDIC specializes in distinct change categories.

2.3 Memory Network

Memory networks were introduced to equip models that support content-based retrieval. Notable examples include Dynamic Memory Networks [22] and Key-Value Memory Networks [33], which improve reasoning by explicitly modeling

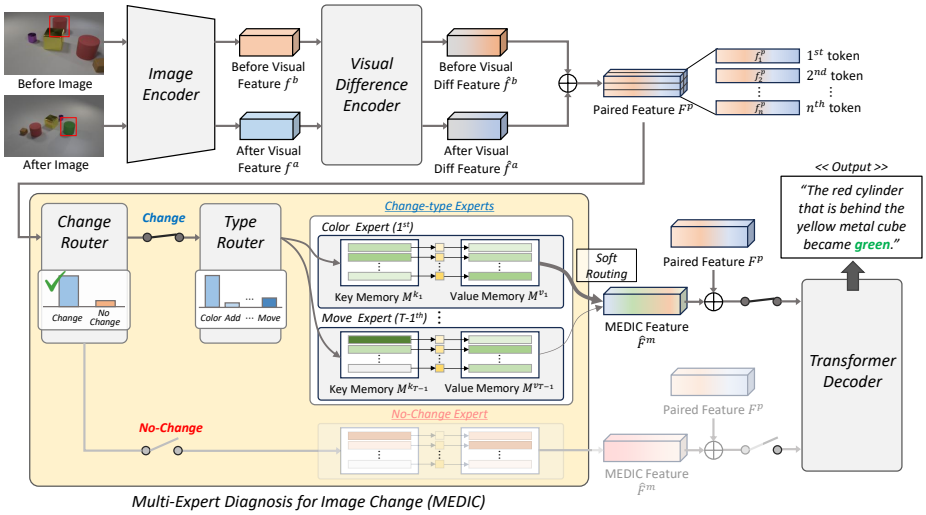


Fig. 3: Overview of our Multi-Expert Diagnosis for Image Change (MEDIC) with a color change example. MEDIC consists of (1) a two-stage router, which first determines the presence of change and then assigns input to specialized experts based on the predicted change type, and (2) change-type experts, each designed to capture fine-grained, type-specific visual cues via dynamic memory-based retrieval. $\bigoplus$ denotes concatenation.

memory access mechanisms. In computer vision, such networks have been applied to various tasks, including object tracking [57], predictive representation learning [8], and video question answering [6]. Key-value memory, in particular, enables efficient and interpretable retrieval through query-key matching [23, 32], and recent works have extended this design by incorporating multi-modal cues or spatial priors for fine-grained alignment [18–21, 51].

Motivated by these advances, we design each expert as a key-value memory. It enables type-specific reasoning by dynamically attending to relevant features, supporting more adaptive and semantically grounded change reasoning.

3 Methodology

3.1 Overall Architecture

As shown in Fig. 3, we propose MEDIC to generate accurate and type-aware change descriptions by explicitly modeling change-type diversity. We describe how MEDIC can be integrated into existing change captioning pipelines that use conventional visual difference encoders and transformer decoders. Given a pair of before image I^b and after image I^a, visual features f^b and f^a are first extracted using the same image encoder, ResNet-101 [9]. These features are passed to a visual difference encoder that highlights potential change cues between the two images, referred to as visual difference features $\hat{f}^b$ and $\hat{f}^a$. They are concatenated

to form paired feature $F^p=[\hat{f}^b;\hat{f}^a] \in \mathbb{R}^{h\times w\times 2d}$ (h, w, and d indicate height, width, and feature dimension, respectively). Then it is fed into the MEDIC.

In the MEDIC, two-stage routers guide the interaction with a set of T change-type experts. A change router first detects whether a change has occurred. If the change is detected, then a type router estimates the distribution over the $(T-1)$ actual change types (*e.g.,* color, texture, add) and softly routes the paired feature F^p to the change-type experts based on the estimated type distribution. The set of expert outputs are denoted as $F^m=\{F^{m_t}\}_{t=1}^{T-1}$ where each $F^{m_t} \in \mathbb{R}^{h\times w\times 2d}$. These outputs are then aggregated via a weighted summation according to the predicted type distribution to form a MEDIC feature $\hat{F}^m \in \mathbb{R}^{h\times w\times 2d}$. In contrast, if no-change is detected, only the output of the no-change expert (T-th expert) is used as $\hat{F}^m$. Then, $\hat{F}^m$ and F^p are concatenated and fed into a transformer decoder to generate the change caption.

3.2 Two-stage Router

The two-stage router in MEDIC routes the paired feature F^p based on a two-step decision process: (1) the change router determines whether a change is present, and (2) if so, the type router estimates the distribution over change types.

(1) Change Router. To determine whether a change is present, the change router applies a classifier to the input F^p. For the i-th sample, it encodes pixel-level logits $\mathbf{z}^i_{\text{chg}} \in \mathbb{R}^{h\times w\times 2}$, then aggregated via global average pooling (GAP) and passed through a softmax to obtain the change probability vector $p^i \in \mathbb{R}^2$.

The change router is trained using cross-entropy loss:

$$\mathcal{L}_{chg} = -\frac{1}{B}\sum_{i=1}^{B} \log p^i_{y^i}, \tag{1}$$

where B is the batch size, $y^i \in \{0,1\}$ denotes the ground-truth label for the i-th input (1: change, 0: no-change), and $p^i_{y^i}$ is the predicted probability in p^i.

(2) Type Router. For inputs where a change is detected, the type router takes F^p as input and outputs pixel-level logits $\mathbf{z}_{\text{type}} \in \mathbb{R}^{h\times w\times (T-1)}$. These logits are pooled with GAP and passed through softmax to yield the change-type probability vector $p^i_{\text{type}} \in \mathbb{R}^{T-1}$. Here, $(T-1)$ denotes the number of change types, excluding the no-change category. The type router is also trained using the cross-entropy loss:

$$\mathcal{L}_{type} = -\frac{1}{B_{\text{chg}}}\sum_{i=1}^{B_{\text{chg}}} \log p^i_{\text{type},y^i}, \tag{2}$$

where B_{chg} is the number of samples with actual changes, $y^i \in \{1,2,\ldots,T-1\}$ is the ground-truth change-type label for the i-th sample, and p^i_{type,y^i} is the predicted probability for the correct change type.

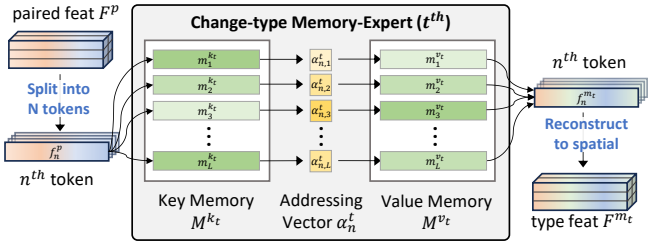


Fig. 4: Detailed architecture of the change-type memory-expert. Paired feature F^p is split into N tokens, each independently processed through a memory network (illustrated here with the n-th token). Each token retrieves type-specific cues via addressing, and the outputs are reassembled into the spatial feature F^{m_t}.

The final routing loss $\mathcal{L}_{router}$ is defined as:

$$\mathcal{L}_{router} = \mathcal{L}_{chg} + \mathcal{L}_{type}, \tag{3}$$

The $\mathcal{L}_{router}$ encourages the router to adaptively select and apply suitable reasoning according to the change type.

3.3 Memory Network for Change-type Experts

We design expert, specialized for a specific change type $t \in \{1, ..., T\}$, as a key-value memory network to capture distinct visual cues. As shown in Fig. 4, the memory of the t-th expert is defined as $M^t = \{M^{k_t}, M^{v_t}\}$, where $M^{k_t} = \{m_\ell^{k_t}\}_{\ell=1}^L$ and $M^{v_t} = \{m_\ell^{v_t}\}_{\ell=1}^L$ denote the key and value memory, each with L slots of dimension $\mathbb{R}^{2d}$.

To embed diverse grid-level representations of each change type, we divide the paired feature F^p into $N{=}hw$ tokens $\{f_n^p\}_{n=1}^N$ where each $f_n^p \in \mathbb{R}^{2d}$. For each token, we compute cosine similarity with every key memory slot to identify relevant entries for retrieval, denoted as:

$$d(f_n^p, m_\ell^{k_t}) = \frac{f_n^p \cdot m_\ell^{k_t}}{\|f_n^p\| \|m_\ell^{k_t}\|}, \quad \text{for } \ell = 1, ..., L. \tag{4}$$

To retrieve information from the value memory, we convert these similarity scores into normalized read weights over memory slots, which we refer to as the addressing vector $\alpha_n^t{=}\{\alpha_{n,\ell}^t\}_{\ell=1}^L \in \mathbb{R}^L$. Specifically, it is computed by a softmax with temperature τ:

$$\alpha_{n,\ell}^t = \frac{\exp\left(d(f_n^p, m_\ell^{k_t})/\tau\right)}{\sum_{s=1}^L \exp\left(d(f_n^p, m_s^{k_t})/\tau\right)}. \tag{5}$$

Using this addressing vector, the output token $f_n^{m_t}$ is retrieved by a weighted sum over the value memory:

$$f_n^{m_t} = \sum_{\ell=1}^{L} \alpha_{n,\ell}^t \cdot m_\ell^{v_t}. \tag{6}$$

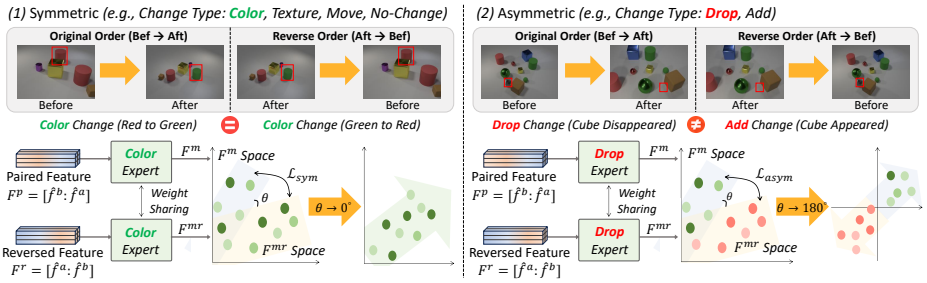


Fig. 5: Overview of the expert consistency loss for (1) symmetric and (2) asymmetric change types, guiding each expert to capture its change-type behavior. For each input pair, paired and reversed features are passed through the expert of the ground-truth change type. Symmetric types (*e.g.,* 'color') generate type-invariant representations under reversal, whereas asymmetric types (*e.g.,* 'drop') yield type-variant representations.

Applying this retrieval process to all N tokens yields output tokens $\{f_n^{m_t}\}_{n=1}^{N}$, which are reshaped to the original spatial layout to obtain the type feature $F^{m_t} \in \mathbb{R}^{h \times w \times 2d}$ for the t-th expert. The memory-based design allows each expert to dynamically retrieve input-adaptive patterns for its change type, enhancing generalization across diverse scenes.

Finally, the ouput of the MEDIC feature $\hat{F}^m \in \mathbb{R}^{h \times w \times 2d}$ is obtained by aggregating the expert outputs based on the routing decisions made by the change and type routers. The aggregation is defined as follows:

$$\hat{F}^m = \begin{cases} \sum_{t=1}^{T-1} p_{\text{type},t} \cdot F^{m_t}, & \text{if a change is detected,} \\ F^{m_T}, & \text{otherwise,} \end{cases} \tag{7}$$

where F^{m_T} denotes the no-change expert output. This soft aggregation adaptively integrates expert outputs based on the predicted change type distribution.

3.4 Expert Consistency Loss

To encourage each change-type expert to learn more type-specific representations, we introduce an expert consistency loss that exploits the intrinsic structural property of change types under input reversal.

As shown in Fig. 5 (1), a 'color' change from red to green remains a 'color' change when reversed to green to red. Since the type label is preserved under swapping, we define such types as *symmetric*. In contrast, as shown in Fig. 5 (2), a 'drop' becomes an 'add' when the input order is reversed. Because the type label changes under reversal, we define such cases as *asymmetric*.

Based on this, we construct two paired features: (1) original $F^p{=}[\hat{f}^b; \hat{f}^a]$ and (2) reversed $F^r{=}[\hat{f}^a; \hat{f}^b]$, where the 'before' and 'after' features are swapped. Both are passed through the same type expert corresponding to the ground-truth change type, producing $F^m, F^{mr} \in \mathbb{R}^{h \times w \times 2d}$. We normalize both along the feature dimension and compute cosine similarity between corresponding features.

The expert consistency loss is defined based on change-type symmetry as:

$$\mathcal{L}_{con}=\begin{cases}\dfrac{1}{B_{sym}}\displaystyle\sum_{i=1}^{B_{sym}}\left(1-\cos(F_i^m,F_i^{mr})\right), & \text{if } \textit{symmetry},\\ \dfrac{1}{B_{asym}}\displaystyle\sum_{i=1}^{B_{asym}}\left(1+\cos(F_i^m,F_i^{mr})\right), & \text{if } \textit{asymmetry},\end{cases} \tag{8}$$

where B_{sym} and B_{asym} denote the number of samples for symmetric and asymmetric types, respectively.

Therefore, the loss enforces directional alignment between F^m and F^{mr} for symmetric types, while encouraging directional opposition for asymmetric types. By leveraging this type-specific orientation signal, each expert learns to encode the intrinsic characteristics that define its corresponding change type.

3.5 Expert Disentangle Loss

We propose an expert disentangle loss to explicitly separate the representation spaces of different change-type experts and the no-change expert. While the expert consistency loss guides each expert to capture type-intrinsic characteristics under input reversal, it does not prevent representations from different experts from becoming overly similar, particularly for visually correlated change types. The disentangle loss addresses this by promoting inter-type separability and isolating no-change features from change features. It consists of (1) an intra-group term that increases separation among change-type experts and (2) an inter-group term that separates change-type experts from the no-change expert.

For each expert t, we compute a summary representation $F_{avg}^{m_t}$ by applying GAP over its output feature F^{m_t}. Let $\mathcal{T}$ denote the set of change types, and T the no-change expert. The cosine similarity between any pair of expert representations is defined as $d_{t,t'}=\cos(F_{avg}^{m_t},F_{avg}^{m_{t'}})$. The intra-group loss is defined as follows:

$$\mathcal{L}_{intra}=\frac{2}{|\mathcal{T}|(|\mathcal{T}|-1)}\sum_{\substack{t,t'\in\mathcal{T}\\ t<t'}}\left[\max\left(0,d_{t,t'}-\delta_{intra}\right)\right]^2. \tag{9}$$

In addition, the inter-group loss is defined as:

$$\mathcal{L}_{inter}=\frac{1}{|\mathcal{T}|}\sum_{t\in\mathcal{T}}\left[\max\left(0,d_{t,T}-\delta_{inter}\right)\right]^2, \tag{10}$$

where δ_{intra} and δ_{inter} are margin hyperparameters for intra- and inter-group separation, respectively. The total expert disentangle loss is computed as follows:

$$\mathcal{L}_{dis}=\mathcal{L}_{intra}+\mathcal{L}_{inter}. \tag{11}$$

Together, these terms explicitly prevent inter-type representation overlap, ensuring that each expert occupies a distinct region in the embedding space while maintaining clear separation from the no-change expert. A detailed figure is included in the supplementary material.

3.6 Overall Objective

The final training objective combines the captioning loss with all auxiliary terms:

$$\mathcal{L}_{total} = \mathcal{L}_{cap} + \lambda_1 \mathcal{L}_{router} + \lambda_2 \mathcal{L}_{con} + \lambda_3 \mathcal{L}_{dis}, \tag{12}$$

where $\lambda_1, \lambda_2, \lambda_3$ are hyperparameters. $\mathcal{L}_{cap}$ denotes the captioning loss used to supervise the transformer decoder.

4 Experiments

4.1 Datasets and Evaluation Metrics

Datasets. We conduct experiments on four datasets: CLEVR-DC [17], CLEVR-Change [35], Spot-the-Diff [15], and Image Editing Request [41]. **CLEVR-DC** includes 48,000 image pairs, featuring more drastic scene differences due to strong distractors. We use the official split: 85% for training, 5% for validation, and 10% for testing. **CLEVR-Change** contains 79,606 image pairs and 493,735 associated captions. We use the official split of 67,660/3,976/7,970 for train/val/test. **Spot-the-Diff** consists of 13,192 surveillance-based image pairs with illumination changes and uses a standard 8:1:1 split. **Image Editing Request (IER)** dataset comprises 3,939 image pairs and 5,695 editing instructions. We adopt the official split: 3,061 pairs for training, 383 for validation, and 495 for testing.

CLEVR-DC and CLEVR-Change follow a single change setting with six predefined change types (color, texture, add, drop, move, no-change). Spot-the-Diff supports both single- and multi-change settings and contains four types (add, drop, move, no-change). Image Editing Request also follows a single change setting with seven types (add, drop, replace, background, illumination, style, resize). The change-type labels for Spot-the-Diff and IER dataset are constructed using an GPT-4o [13] (see supplementary material for details).

Evaluation Metrics. We evaluate the caption quality using five metrics: BLEU-4 ($\mathcal{B}$) [34], METEOR ($\mathcal{M}$) [2], ROUGE-L ($\mathcal{R}$) [29], CIDEr ($\mathcal{C}$) [50], SPICE ($\mathcal{S}$) [1].

4.2 Implementation Details

We apply MEDIC to representative recent change captioning models with publicly available code: SCORER [46], SMART [45], and DIRL [44]. We use L=100 memory slots. The number of experts is determined by the change-type taxonomy of each dataset. Specifically, we allocate one expert to each change type, including the no-change type. Thus, we use T=6 for CLEVR-DC and CLEVR-Change, T=4 for Spot-the-Diff, and $T = 7$ for Image-Editing-Request. We set τ=0.1, δ_{intra}=0.7, and δ_{inter}=0.2. The weights are λ_1=0.1, λ_2=0.01, and λ_3=0.1. All experiments use a single NVIDIA RTX 4090 GPU.

Table 1: Comparison of change captioning performance on CLEVR-DC (left), CLEVR-Change (center), and Spot-the-Diff (right) datasets under single-change setting. Note that, our MEDIC is compared with three state-of-the-art baselines (SCORER [46], SMART [45], and DIRL [44]) with publicly available source code.

Method	CLEVR-DC Dataset					CLEVR-Change Dataset					Spot-the-Diff Dataset				
	$\mathcal{B}$	$\mathcal{M}$	$\mathcal{R}$	$\mathcal{C}$	$\mathcal{S}$	$\mathcal{B}$	$\mathcal{M}$	$\mathcal{R}$	$\mathcal{C}$	$\mathcal{S}$	$\mathcal{B}$	$\mathcal{M}$	$\mathcal{R}$	$\mathcal{C}$	$\mathcal{S}$
DUDA (ICCV'19) [35]	40.3	27.1	-	56.7	16.1	47.3	33.9	-	112.3	24.5	-	-	-	-	-
DUDA+ (CVPR'21) [10]	-	-	-	-	-	51.2	37.7	70.5	115.4	31.1	8.1	12.5	-	34.5	-
M-VAM (ECCV'20) [40]	-	-	-	-	-	50.3	37.0	69.7	114.9	30.5	-	-	-	-	-
VACC (ICCV'21) [16]	45.0	29.3	-	71.7	17.6	-	-	-	-	-	-	-	-	-	-
MCCFormers-D (ICCV'21) [37]	-	-	-	-	-	52.4	38.3	-	121.6	26.8	10.0	12.4	-	43.1	18.3
MCCFormers-S (ICCV'21) [37]	-	-	-	-	-	57.4	41.2	-	125.5	32.4	-	12.3	-	41.6	16.3
PCL w/o Pretrain (AAAI'22) [52]	-	-	-	-	-	32.7	27.7	57.2	89.8	-	-	-	-	-	-
NCT (TMM'23) [43]	47.5	32.5	65.1	76.9	15.6	55.1	40.2	73.8	124.1	32.9	-	-	-	-	-
VARD-Trans (TIP'23) [42]	48.3	32.4	-	77.6	15.4	55.4	40.1	73.8	126.4	32.6	-	12.5	29.3	30.3	17.3
I3N-TD (TMM'23) [54]	-	-	-	-	-	55.8	40.6	73.9	125.6	32.8	-	13.0	31.5	42.7	18.6
RDD+ACR (AAAI'25) [27]	-	-	-	-	-	56.1	41.3	75.0	128.1	33.5	9.2	13.9	31.0	43.6	-
DECIDER (AAAI'25) [56]	-	-	-	-	-	56.4	39.7	75.3	131.3	-	10.7	14.2	41.6	39.9	-
MCT-CCDiff (TIP'25) [12]	-	-	-	-	-	57.5	40.6	75.6	131.7	-	10.8	14.5	35.5	41.7	-
SCORER (ICCV'23) [46]	49.4	33.4	66.1	83.7	16.2	56.3	41.2	74.5	126.8	33.3	**10.2**	12.2	-	38.9	**18.4**
SCORER + MEDIC (Ours)	**56.0**	**35.9**	**70.1**	**97.4**	**19.0**	**57.6**	**41.8**	**75.5**	**130.7**	**33.7**	**10.2**	**12.4**	**32.9**	**39.2**	**18.4**
SMART (TPAMI'24) [45]	48.3	30.7	65.2	81.2	16.0	56.1	40.8	74.2	127.0	33.4	-	13.5	31.6	39.4	19.0
SMART + MEDIC (Ours)	**57.1**	**35.3**	**70.9**	**98.9**	**19.8**	**56.4**	**42.5**	**75.9**	**128.1**	**34.6**	**9.1**	**14.2**	**32.6**	**43.1**	**22.1**
DIRL (ECCV'24) [44]	51.4	32.3	66.3	84.1	16.8	56.2	41.0	73.8	126.0	33.1	10.3	13.8	32.8	40.9	19.9
DIRL + MEDIC (Ours)	**58.5**	**35.6**	**71.0**	**99.8**	**20.0**	**57.5**	**41.3**	**74.6**	**129.2**	**33.5**	**11.1**	**14.6**	**33.7**	**45.5**	**23.0**

4.3 Comparison under Single-Change Setting

Table 1 shows the performance comparison across three datasets, where MEDIC is applied to three baselines (SCORER [46], SMART [45], DIRL [44]). As SMART and DIRL lack CLEVR-DC and CLEVR-Change results respectively, and no method reports type-wise performance, we reproduced all missing results using publicly available official source code. Paired t-tests on all datasets confirm that the improvements of MEDIC over baselines are statistically significant ($p < 0.05$).

Results on the CLEVR-DC Dataset. CLEVR-DC presents a particularly challenging setting, where severe viewpoint shifts act as strong distractors within the scene. As shown in Table 1 (left), despite these challenges, applying MEDIC consistently yielded superior performance across all metrics. It demonstrates that explicitly separating change types during training is crucial for accurately capturing changes, especially in the presence of distractors.

Results on the CLEVR-Change Dataset. We also evaluate MEDIC on CLEVR-Change, which shares the same synthetic domain as CLEVR-DC but involves fewer distractors and more controlled scene variations. As shown in Table 1 (center), MEDIC consistently improved performance across all evaluation metrics. Even in less cluttered environments, explicitly modeling change types contributes to better scene understanding and caption generation.

Results on the Spot-the-Diff Dataset. In Table 1 (right), we evaluate our method on Spot-the-Diff, a low-resolution real-world dataset with subtle illumination changes that make localization challenging. Despite these challenges, MEDIC consistently improves across all evaluation metrics, demonstrating strong

Table 2: CIDEr performance comparison by change type across datasets. Types: 'Color' (Clr), 'Texture' (Tex), 'Add' (Add), 'Drop' (Drp), 'Move' (Mv), and 'No-Change' (NC).

CLEVR-DC Dataset						
Method	**Clr**	**Tex**	**Add**	**Drp**	**Mv**	**NC**
DIRL [44]	108.6	76.0	76.5	81.5	38.8	41.6
DIRL + MEDIC	**118.8**	**88.2**	**86.1**	**88.3**	**54.4**	**82.3**
CLEVR-Change Dataset						
Method	**Clr**	**Tex**	**Add**	**Drp**	**Mv**	**NC**
DIRL [44]	149.8	137.5	126.9	137.5	87.6	113.8
DIRL + MEDIC	**151.7**	**142.6**	**138.6**	**141.2**	**96.3**	**116.2**
Spot-the-Diff Dataset						
Method	**Add**	**Drp**	**Mv**	**NC**		
DIRL [44]	38.4	38.6	41.2	14.3		
DIRL + MEDIC	**47.2**	**47.6**	**48.4**	**17.4**		

Table 3: Comparison on the Spot-the-Diff under multi-change setting, following [47].

Method	$\mathcal{B}$	$\mathcal{M}$	$\mathcal{R}$	$\mathcal{C}$	$\mathcal{S}$
SCORER [46]	5.1	9.3	23.0	20.9	11.9
SCORER + MEDIC	**6.1**	**9.3**	**25.7**	**25.9**	**16.4**
SMART [45]	3.7	8.7	23.1	25.3	15.5
SMART + MEDIC	**4.5**	**8.9**	**25.7**	**31.6**	**17.4**
DIRL [44]	4.6	9.5	23.8	21.5	14.8
DIRL + MEDIC	**6.8**	**11.2**	**26.7**	**31.2**	**20.1**

Table 4: Expert architecture comparison of MEDIC on CLEVR-DC (DIRL baseline).

Architecture	$\mathcal{B}$	$\mathcal{M}$	$\mathcal{R}$	$\mathcal{C}$	$\mathcal{S}$
MLP (2 layer)	56.1	35.1	70.1	96.1	19.3
Memory Network	**58.5**	**35.6**	**71.0**	**99.8**	**20.0**

robustness and generalization to complex and real-world scenarios.

Type-wise Performance Comparison. We present a detailed comparison across change types in Table 2, which contrasts the baseline DIRL with MEDIC-integrated model. The table includes six change types for CLEVR-DC and CLEVR-Change ('color', 'texture', 'add', 'drop', 'move', 'no-change'), and four for Spot-the-Diff ('add', 'drop', 'move', 'no-change'). MEDIC outperforms DIRL across all datasets and change types. Type-specific experts improve the ability of the model to capture diverse change patterns and achieve more accurate results.

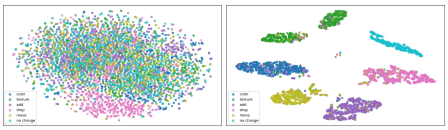
4.4 Comparison under Multi-Change Setting

To verify the scalability of MEDIC to multi-change scenarios, we further conducted experiments on the Spot-the-Diff dataset [15], the only publicly available dataset applicable to multi-change settings. While most previous works [27, 56] used the Spot-the-Diff dataset in a single-change setting, CARD [47] configured it for a multi-change setting and conducted experiments accordingly, which we followed. Using this protocol, we applied MEDIC to SCORER [46], SMART [45], and DIRL [44], reproducing the latter two for fair comparison. For this setting, we computed the consistency loss for each annotated type and took the mean across them, while using Binary Cross-Entropy for multi-label routing supervision. We generated multi-change type labels using a prompt detailed in the supplementary material. Table 3 shows that MEDIC clearly outperforms all baselines, indicating that the soft gating mechanism effectively combines knowledge from multiple type experts and generalizes robustly beyond the single-change assumption.

4.5 Ablation Studies

To better understand the contribution of each component in our method, we conduct ablation studies on the CLEVR-DC, with MEDIC applied to DIRL [44].

Effect of the Memory-based Expert. To examine the impact of expert architecture within the same type-specific framework, we compare two expert



(a) DIRL (b) DIRL+MEDIC (Ours)

Fig. 6: t-SNE visualization of learned representations by change type.

Table 5: Comparison between type-agnostic MoE and our type-specific expert.

Method	# Experts	CIDEr
Standard MoE (type-agnostic)	6	94.2
	12	90.7
	24	93.6
DIRL+MEDIC (type-specific)	6	**99.8**

Table 6: Effect of proposed losses on the CLEVR-DC dataset.

Settings	$\mathcal{B}$	$\mathcal{M}$	$\mathcal{R}$	$\mathcal{C}$	$\mathcal{S}$
DIRL + MEDIC	**58.5**	**35.6**	**71.0**	**99.8**	**20.0**
w/o $\mathcal{L}_{dis}$	57.7	35.2	70.6	98.2	19.3
w/o $\mathcal{L}_{con}$	57.0	35.0	70.5	99.1	19.7
w/o $\mathcal{L}_{dis}$, $\mathcal{L}_{con}$	54.8	34.9	69.9	96.7	18.4
w/o $\mathcal{L}_{router}$, $\mathcal{L}_{dis}$, $\mathcal{L}_{con}$	53.2	33.0	69.9	87.7	18.5

Table 7: Ablation on the number of memory slots per expert on CLEVR-DC dataset.

# Slot	$\mathcal{B}$	$\mathcal{M}$	$\mathcal{R}$	$\mathcal{C}$	$\mathcal{S}$	# Params(M)	Time(ms/sample)
-	51.4	32.3	66.3	84.1	16.8	13.90	26.90
50	56.6	35.2	70.5	97.8	19.0	24.50	30.11
100	**58.5**	**35.6**	**71.0**	**99.8**	**20.0**	**25.12**	**30.16**
200	57.8	35.1	70.3	98.8	18.2	26.35	30.21
400	56.9	35.0	70.9	97.7	18.5	28.80	30.22

designs: a standard MLP-based expert [25] and our memory expert. Table 4 shows that ours outperforms the MLP-based expert across all metrics, verifying the advantages of input-adaptive retrieval in capturing type-specific cues.

t-SNE Visualization of Type-aware Experts. To see whether each expert captures type-specific semantics, we visualize the expert outputs using t-SNE. As shown in Fig. 6, the baseline DIRL shows entangled clusters with unclear boundaries across change types. In contrast, MEDIC generates well-separated clusters, which shows that each expert learns more discriminative and type-specific representations and highlights the effectiveness of type-aware modeling.

Effectiveness of the Type-Specific Expert Design. To validate the effectiveness of our type-specific expert design, we compare MEDIC with a standard type-agnostic MoE. As shown in Table 5, our design consistently achieves higher performance across different expert configurations. This result indicates that a type-specific expert design is more effective than a generic MoE design. We further compare MEDIC with a simple type-aware baseline trained with an auxiliary change-type classification loss in the supplementary material, showing that the gains achieved by MEDIC are not merely due to access to type labels but come from explicit type-specialized expert reasoning.

Effect of the Proposed Losses. We analyze the impact of the expert consistency loss $\mathcal{L}_{con}$, expert disentangle loss $\mathcal{L}_{dis}$, and routing loss $\mathcal{L}_{router}$ (Table 6). Removing $\mathcal{L}_{dis}$ or $\mathcal{L}_{con}$ reduces performance, while each individually still improves results, indicating their complementary role of expert specialization. Removing $\mathcal{L}_{router}$ forces implicit expert selection and prevents explicit routing training, and leads to further performance drops. Combining all three losses achieves the best performance and demonstrates their synergy in learning type-specialized experts.

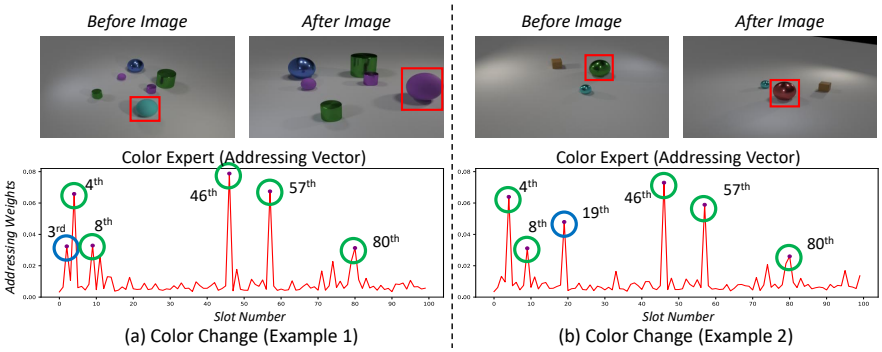


Fig. 7: Visualization of two image pairs with the same change type (color), and their slot activation profiles. Green circles indicate commonly activated slots across inputs, capturing type-specific patterns, while blue circles highlight input-specific activations.

Effect of the Memory Slot Size and Overhead. We analyze how the number of memory slots per expert affects the results and computational cost. As shown in Table 7, increasing the number of slots improves performance up to 100, achieving the best results with only a marginal cost increase. Although performance slightly drops beyond this point, it still surpasses the non-memory baseline.

4.6 Discussion

Dynamic Slot Activation. To verify that our memory-based experts adapt to input content, we visualize the addressing vectors of two color-type samples in Fig. 7. Both consistently attend to specific slots (green circles), indicating anchors for type-specific information, while distinct slots (blue circles) reflect input-specific responses. This shows that our memory architecture captures both shared patterns and adaptive behavior within each change type.

Effect of the Hyper-parameters. As shown in Table 8, MEDIC remains stable across a wide range of hyper-parameters λ_1, λ_2, and λ_3 values, with all configurations achieving strong results and only minor variation.

Generalization to Object- and Global-Level Changes. To see that MEDIC can extend beyond object-level changes, we evaluate it on the Image Editing Request (IER) dataset [41] that contains more diverse change scenarios. While most existing datasets primarily focus on object-level changes, IER includes three object-level types ('add', 'drop', 'replace') and four global-level types ('background', 'illumination', 'style', 'resize'). As shown in Table 9, when using DIRL as the baseline, MEDIC achieves consistent improvements across all change types. It demonstrates robustness across both object-level and global changes.

Compatibility with LVLM-Based Methods. To further verify compatibility with recent LVLM-based approaches, we apply MEDIC to BLIP2IDC [5] and eval-

Table 8: Evaluation results under different settings of the hyper-parameters on the CLEVR-DC dataset.

Target	λ	$\mathcal{B}$	$\mathcal{M}$	$\mathcal{R}$	$\mathcal{C}$	$\mathcal{S}$
λ_1	**0.1**	**58.5**	**35.6**	**71.0**	**99.8**	**20.0**
	0.01	58.1	35.5	71.1	99.2	19.3
	0.001	56.5	35.6	70.7	98.6	18.6
λ_2	0.1	58.2	35.5	71.1	99.3	19.5
	0.01	**58.5**	**35.6**	**71.0**	**99.8**	**20.0**
	0.001	57.4	35.3	70.8	99.6	19.5
λ_3	**0.1**	**58.5**	**35.6**	**71.0**	**99.8**	**20.0**
	0.01	57.7	35.6	70.9	98.9	19.9
	0.001	57.4	35.3	70.6	98.2	20.0

Table 9: Results on IER dataset.

Method	$\mathcal{B}$	$\mathcal{M}$	$\mathcal{R}$	$\mathcal{C}$
DIRL [44]	10.9	15.0	41.0	34.1
DIRL+MEDIC	**11.2**	**15.6**	**42.5**	**37.2**

Table 10: Results on LVLM-based works.

Method	$\mathcal{B}$	$\mathcal{M}$	$\mathcal{C}$
BLIP2IDC [5]	49.3	33.0	88.5
BLIP2IDC+MEDIC	**57.3**	**33.8**	**106.4**

uate it on CLEVR-DC. As shown in Table 10, incorporating MEDIC consistently improves performance across all metrics, demonstrating that our method can be effectively integrated with LVLM-based models. Additional comparisons with zero-shot and fine-tuned LVLM-based change captioning methods are provided in the supplementary material.

Label Reliability & Routing Robustness. For datasets without type annotations, we derive change-type labels from ground-truth captions using GPT-4o. To validate this supervision, applying the same labeling protocol to CLEVR-Change and CLEVR-DC, where ground-truth type labels are available, yields 91.2% and 93.8% accuracy, respectively. Moreover, even when 10/20/40% of CLEVR-DC training type labels are randomly corrupted, MEDIC achieves 95.4/94.3/94.0 CIDEr, still outperforming DIRL (84.1), showing robustness to imperfect type supervision. In addition, stage-1/2 routing accuracies show 92%/81%, respectively, and qualitative routing-error analysis shows that our soft expert aggregation can compensate for some top-1 routing mistakes. Detailed prompts, label validation, noisy-label analysis, and routing-error examples are provided in the supplementary material.

Generalization to In-Domain Held-Out Types As MEDIC is built on a modular design with dedicated experts for each change type, it can be extended when additional supervised change types become available. To examine this property, we conduct a leave-one-type-out evaluation on CLEVR-DC in an in-domain held-out setting: one existing change type, such as 'color', is excluded during training and evaluated afterward. When 'color' is held out, CIDEr on that type drops from 118.8 to 18.4. When this held-out type is later introduced, MEDIC can add a new expert and update the expanded router while freezing the existing experts. With only 0.2M additional parameters (0.82% of the 25.12M total), CIDEr on the held-out type recovers from 18.4 to 113.4 while largely preserving performance on previously learned types. Details are provided in the supplementary material.

Limitations. While MEDIC can flexibly extend to unseen datasets by using LLMs to extract new change-type labels, it still relies on the availability of such

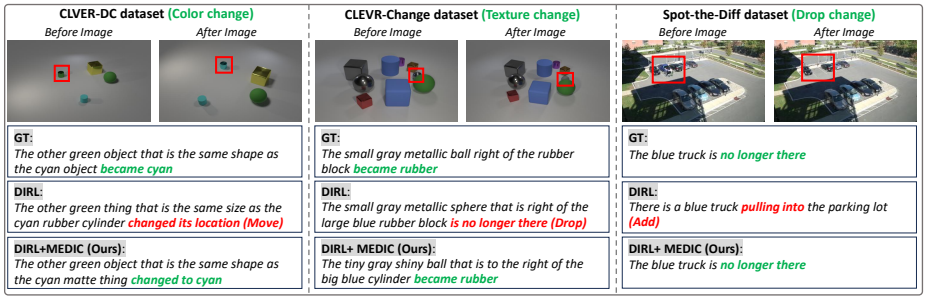


Fig. 8: Qualitative comparisons across datasets. Each example shows ground-truth (GT), baseline (DIRL), and the proposed method (DIRL+MEDIC).

type annotations. Future work will explore self-supervised approaches that can infer change types without explicit labels.

4.7 Qualitative Results

Figure 8 presents qualitative examples on CLEVR-Change, CLEVR-DC, Spot-the-Diff, and Image-Editing-Request. MEDIC generally produces more accurate and type-consistent captions than DIRL by explicitly modeling change types. We additionally provide representative failure cases and their analysis in the supplementary material to highlight the remaining limitations of MEDIC under subtle local changes, crowded multi-change scenes, and ambiguous global edits.

5 Conclusion

We introduce MEDIC, a novel framework that explicitly models change types using memory-based experts specialized for the change type. These experts dynamically retrieve type-relevant visual patterns to enable input-adaptive and type-aware reasoning for accurate change descriptions. Extensive experiments show that explicit change-type modeling leads to consistent performance across diverse and challenging scenarios.

Acknowledgements

This work was partly supported by IITP-ITRC grant funded by the Korea government (MSIT)(IITP-2026-RS-2023-00258649, 30%) and partly supported by IITP grant funded by the Korea government (MSIT)(IITP-2023-RS-2023-00266615: Convergence Security Core Talent Training Business Support Program (20%), IITP-2022-II220078: Explainable Logical Reasoning for Medical Knowledge Generation (25%), No. RS-2024-00509257: Global AI Frontier Lab (25%)).

References

1. Anderson, P., Fernando, B., Johnson, M., Gould, S.: Spice: Semantic propositional image caption evaluation. In: ECCV (2016)
2. Banerjee, S., Lavie, A.: Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In: ACL Workshop (2005)
3. Black, A., Shi, J., Fan, Y., Bui, T., Collomosse, J.: Vixen: Visual text comparison network for image difference captioning. In: AAAI (2024)
4. Cai, W., Liu, Q., Wang, Y.: Spmtrack: Spatio-temporal parameter-efficient fine-tuning with mixture of experts for scalable visual tracking. In: CVPR (2025)
5. Evennou, G., Chaffin, A., Chappelier, V., Kijak, E.: Reframing image difference captioning with blip2idc and synthetic augmentation. In: WACV (2025)
6. Fan, C., Zhang, X., Zhang, S., Wang, W., Zhang, C., Huang, H.: Heterogeneous memory enhanced multimodal attention model for video question answering. In: CVPR (2019)
7. Fedus, W., Zoph, B., Shazeer, N.: Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. JMLR (2022)
8. Han, T., Xie, W., Zisserman, A.: Memory-augmented dense predictive coding for video representation learning. In: ECCV (2020)
9. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)
10. Hosseinzadeh, M., Wang, Y.: Image change captioning by learning from an auxiliary task. In: CVPR (2021)
11. Hoxha, G., Chouaf, S., Melgani, F., Smara, Y.: Change captioning: A new paradigm for multitemporal remote sensing image analysis. TGRS (2022)
12. Hu, J., Zhong, G., Yuan, J., Pan, W., Wang, X.: Mct-ccdiff: Context-aware contrastive diffusion model with mediator-bridging cross-modal transformer for image change captioning. TIP (2025)
13. Hurst, A., Lerer, A., Goucher, A.P., Perelman, A., Ramesh, A., Clark, A., Ostrow, A., Welihinda, A., Hayes, A., Radford, A., et al.: Gpt-4o system card. arXiv preprint arXiv:2410.21276 (2024)
14. Jacobs, R.A., Jordan, M.I., Nowlan, S.J., Hinton, G.E.: Adaptive mixtures of local experts. Neural computation (1991)
15. Jhamtani, H., Berg-Kirkpatrick, T.: Learning to describe differences between pairs of similar images. In: EMNLP (2018)
16. Kim, H., Kim, J., Lee, H., Park, H., Kim, G.: Agnostic change captioning with cycle consistency. In: ICCV (2021)
17. Kim, H., Kim, J., Lee, H., Park, H., Kim, G.: Viewpoint-agnostic change captioning with cycle consistency. In: ICCV (2021)
18. Kim, J.U., Kim, H.I., Ro, Y.M.: Stereoscopic vision recalling memory for monocular 3d object detection. TIP (2023)
19. Kim, J.U., Park, S., Ro, Y.M.: Robust small-scale pedestrian detection with cued recall via memory learning. In: ICCV (2021)
20. Kim, J.U., Park, S., Ro, Y.M.: Towards versatile pedestrian detector with multisensory-matching and multispectral recalling memory. In: AAAI (2022)
21. Kim, J.U., Ro, Y.M.: Enabling visual object detection with object sounds via visual modality recalling memory. TNNLS (2023)
22. Kumar, A., Irsoy, O., Ondruska, P., Iyyer, M., Bradbury, J., Gulrajani, I., Zhong, V., Paulus, R., Socher, R.: Ask me anything: Dynamic memory networks for natural language processing. In: ICML (2016)

23. Lee, S., Kim, H.G., Choi, D.H., Kim, H.I., Ro, Y.M.: Video prediction recalling long-term motion context via memory alignment learning. In: ICCV (2021)
24. Lepikhin, D., Lee, H., Xu, Y., Chen, D., Firat, O., Huang, Y., Krikun, M., Shazeer, N., Chen, Z.: Gshard: Scaling giant models with conditional computation and automatic sharding. arXiv preprint arXiv:2006.16668 (2020)
25. Li, J., Wang, X., Zhu, S., Kuo, C.W., Xu, L., Chen, F., Jain, J., Shi, H., Wen, L.: Cumo: Scaling multimodal llm with co-upcycled mixture-of-experts (2024)
26. Li, M., Lin, B., Chen, Z., Lin, H., Liang, X., Chang, X.: Dynamic graph enhanced contrastive learning for chest x-ray report generation. In: CVPR (2023)
27. Li, R., Li, L., Zhang, J., Zhao, Q., Wang, H., Yan, C.: Region-aware difference distilling with attribute-guided contrastive regularization for change captioning. In: AAAI (2025)
28. Li, Y., Tu, Y., Li, L., Su, L., Huang, Q.: Change entity-guided heterogeneous representation disentangling for change captioning. In: ACL (2025)
29. Lin, C.Y.: Rouge: A package for automatic evaluation of summaries. In: ACL (2004)
30. Liu, C., Chen, K., Qi, Z., Liu, Z., Zhang, H., Zou, Z., Shi, Z.: Pixel-level change detection pseudo-label learning for remote sensing change captioning. In: IGARSS. IEEE (2024)
31. Liu, Y., Wang, R., Shan, S., Chen, X.: Structure inference net: Object detection using scene-level context and instance-level relationships. In: CVPR (2018)
32. Marchetti, F., Becattini, F., Seidenari, L., Bimbo, A.D.: Mantra: Memory augmented networks for multiple trajectory prediction. In: CVPR (2020)
33. Miller, A., Fisch, A., Dodge, J., Karimi, A.H., Bordes, A., Weston, J.: Key-value memory networks for directly reading documents. arXiv preprint arXiv:1606.03126 (2016)
34. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: Bleu: a method for automatic evaluation of machine translation. In: ACL (2002)
35. Park, D.H., Darrell, T., Rohrbach, A.: Robust change captioning. In: CVPR (2019)
36. Park, K.R., Park, J., Kim, S.T., Lee, H.J., Kim, J.U.: Leveraging textual compositional reasoning for robust change captioning. In: AAAI (2026)
37. Qiu, Y., Yamamoto, S., Nakashima, K., Suzuki, R., Iwata, K., Kataoka, H., Satoh, Y.: Describing and localizing multiple changes with transformers. In: ICCV (2021)
38. Riquelme, C., Puigcerver, J., Mustafa, B., Neumann, M., Jenatton, R., Susano Pinto, A., Keysers, D., Houlsby, N.: Scaling vision with sparse mixture of experts (2021)
39. Shazeer, N., Mirhoseini, A., Maziarz, K., Davis, A., Le, Q., Hinton, G., Dean, J.: Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. arXiv preprint arXiv:1701.06538 (2017)
40. Shi, X., Yang, X., Gu, J., Joty, S., Cai, J.: Finding it at another side: A viewpoint-adapted matching encoder for change captioning. In: ECCV (2020)
41. Tan, H., Dernoncourt, F., Lin, Z., Bui, T., Bansal, M.: Expressing visual relationships via language. In: ACL (2019)
42. Tu, Y., Li, L., Su, L., Du, J., Lu, K., Huang, Q.: Adaptive representation disentanglement network for change captioning. TIP (2023)
43. Tu, Y., Li, L., Su, L., Lu, K., Huang, Q.: Neighborhood contrastive transformer for change captioning. TMM (2023)
44. Tu, Y., Li, L., Su, L., Yan, C., Huang, Q.: Distractors-immune representation learning with cross-modal contrastive regularization for change captioning. In: ECCV (2024)
45. Tu, Y., Li, L., Su, L., Zha, Z.J., Huang, Q.: Smart: Syntax-calibrated multi-aspect relation transformer for change captioning. TPAMI (2024)

46. Tu, Y., Li, L., Su, L., Zha, Z.J., Yan, C., Huang, Q.: Self-supervised cross-view representation reconstruction for change captioning. In: ICCV (2023)
47. Tu, Y., Li, L., Su, L., Zha, Z.J., Yan, C., Huang, Q.: Context-aware difference distilling for multi-change captioning. In: ACL (2024)
48. Tu, Y., Li, L., Yan, C., Gao, S., Yu, Z.: R^3net: relation-embedded representation reconstruction network for change captioning. In: EMNLP (2021)
49. Tu, Y., Yao, T., Li, L., Lou, J., Gao, S., Yu, Z., Yan, C.: Semantic relation-aware difference representation learning for change captioning. In: ACL (2021)
50. Vedantam, R., Lawrence Zitnick, C., Parikh, D.: Cider: Consensus-based image description evaluation. In: CVPR (2015)
51. Weng, W., Wu, M., Jiang, H., Kong, W., Kong, X., Xia, F.: Pattern-matching dynamic memory network for dual-mode traffic prediction. T-ITS (2025)
52. Yao, L., Wang, W., Jin, Q.: Image difference captioning with pre-training and contrastive learning. In: AAAI (2022)
53. Yue, S., Tu, Y., Li, L., Gao, S., Yu, Z.: Multi-grained representation aggregating transformer with gating cycle for change captioning. ACM MM (2024)
54. Yue, S., Tu, Y., Li, L., Yang, Y., Gao, S., Yu, Z.: I3n: Intra-and inter-representation interaction network for change captioning. TMM (2023)
55. Zhenxing, M., Xu, D.: Switch-nerf: Learning scene decomposition with mixture of experts for large-scale neural radiance fields. In: ICLR (2022)
56. Zhong, G., Hu, J., Chen, J., Yuan, J., Pan, W.: Decider: Difference-aware contrastive diffusion model with adversarial perturbations for image change captioning. In: AAAI (2025)
57. Zhu, L., Yang, Y.: Inflated episodic memory with region self-attention for long-tailed visual recognition. In: CVPR (2020)