CLIP4IDC: CLIP for Image Difference Captioning

Zixin Guo, Tzu-Jui Julius Wang, Jorma Laaksonen
Department of Computer Science, Aalto University, Finland
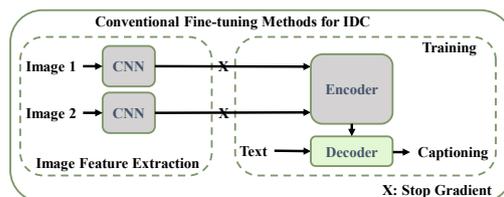{zixin.guo, tzu-jui.wang, jorma.laaksonen}@aalto.fi

Abstract

Image Difference Captioning (IDC) aims at generating sentences to describe differences between two similar-looking images. Conventional approaches learn an IDC model with a pre-trained and usually frozen visual feature extractor. Accordingly, two major issues may arise: (1) a large domain gap usually exists between the pre-training datasets used for training such a visual encoder and that of the downstream IDC task, and (2) the visual feature extractor, when separately encoding two images, often does not effectively encode the visual changes between two images. Due to the excellent zero-shot performance of the recently proposed CLIP, we thus propose CLIP4IDC to transfer a CLIP model for the IDC task to address those issues. Different from directly fine-tuning CLIP to generate sentences, we introduce an adaptation training process to adapt CLIP's visual encoder to capture and align differences in image pairs based on the textual descriptions. Experiments on three IDC benchmark datasets, CLEVR-Change, Spot-the-Diff, and Image-Editing-Request, demonstrate the effectiveness of CLIP4IDC.
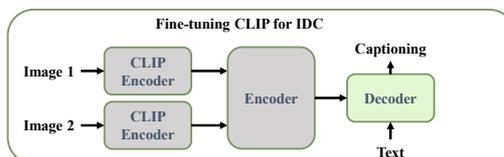
1 Introduction

Tasks involving understanding and expressing visual contents are hard for machines because modelling relationships between the visual and textual domains requires sophisticated computational reasoning. As one of the tasks, image Captioning (IC) (Vinyals et al., 2015; Xu et al., 2015) aims at generating a coherent description given an image. Extended from image captioning, Image Difference Captioning (IDC) (Jhamtani and Berg-Kirkpatrick, 2018; Park et al., 2019) describes the subtle changes that appear in a pair of two similar images. It is more challenging as a machine is required to recognize both visual objects and nuances in the pair.

A conventional approach to IDC is shown in Figure 1a. First, the visual features of an image pair are

(a) The fine-tuning strategy with a frozen (CNN) feature extractor.

(b) CLIP's fine-tuning strategy with an image encoder also fine-tuned.

Figure 1: Different conventional fine-tuning (FT) strategies may suffer from poor task accuracy due to: (1) not accounting for gaps introduced by either different objectives in pre-training (PT) and FT, and (2) domain shift in datasets used in PT and FT.

extracted offline with pre-trained models (He et al., 2016; Ren et al., 2015). Then a captioning network generates sentence(s) to describe the changes in the pair. Even though such approaches have made great progress (Park et al., 2019; Kim et al., 2021; Huang et al., 2021; Hosseinzadeh and Wang, 2021; Sun et al., 2022), they suffer from the fact that the visual features do not account for the domain gap between the pre-training and IDC tasks. Lei et al. (2021) demonstrated that the purpose of the feature extractor trained on the original task introduces a gap with that of the subsequent tasks. For example, the features extracted by models trained on image classification task focus on high-level context and lose fine-grained information required for IDC. Moreover, the extracted visual representations of single modality are uncorrelated with the textual ones.

As an effective approach to deal with the drawbacks, fine-tuning models on the target dataset narrows the gap between the tasks. Yao et al. (2022)

showed that a Transformer (Vaswani et al., 2017) model that was pre-trained and fine-tuned on the same offline-extracted features achieves state-of-the-art results in IDC. However, it does not yet fully exploit the knowledge from the large-scale dataset as in the recent advancements in vision-language (VL) pre-training (Zhou et al., 2020; Li et al., 2021) (VLP). In particular, CLIP (Radford et al., 2021), a contrastive VLP model has demonstrated its zero-shot superiority in numerous VL downstream tasks (Luo et al., 2021; Tang et al., 2021).

We set out experimenting with a typical CLIP fine-tuning strategy on the IDC task as shown in Figure 1b, where CLIP's visual encoder is learned and fine-tuned on raw pixels. However, gaps still exist not only between the objectives of CLIP pre-training and IDC, but also between the collected image-text pairs for pre-training and the image difference pairs in IDC. These gaps throttle the model in adapting for the IDC task.

To tackle these problems, we study how to efficiently transfer a pre-trained CLIP for IDC. The overview of the proposed CLIP4IDC model is shown in Figure 2. Compared to directly fine-tuning CLIP for the IDC task, CLIP4IDC employs "*adapt*-and-*fine-tune*" strategy. To *adapt*, the CLIP encoder learns to capture the fine-grained differences in the image pair rather than to produce only high-level semantic information separately for these two images. The visual and textual representations for the image pairs and the sentences are learned to be aligned with a retrieval loss in this stage. To *fine-tune*, the learned vision encoder is followed by a captioning Transformer trained from scratch.

Extensive experiments are conducted on synthetic and real benchmark datasets CLEVR-Change (Park et al., 2019) and Spot-the-Diff (Jhamtani and Berg-Kirkpatrick, 2018), respectively. In addition, results on Image-Editing-Request (Tan et al., 2019), a mixed real–synthetic dataset, are also reported. CLIP4IDC outperforms the strong baselines on all the metrics on these three datasets. The main contributions of this work are:

1) Compared with the conventional approaches that are trained on pre-extracted features, we fine-tune CLIP for IDC on raw pixels. This retains the expressiveness of the pre-trained features as well as adapting them to the new task domain.

2) We propose CLIP4IDC, which consists of adaptation and fine-tuning stages, to narrow the gap between the objectives and data domains during pre-training CLIP and fine-tuning it for IDC. The adaptation is learned by mutually retrieving the visual differences and the descriptions.

3) Extensive experiments show that CLIP4IDC outperforms multiple strong baselines in the IDC task on three datasets on all the metrics.[1]

2 CLIP4IDC

As shown in Figure 1a, the canonical IDC approach generates sentences on pre-extracted features. The bottleneck lies in three aspects: 1) the stopped gradient flow in the feature extraction, 2) the mismatched objectives and data domains between the pre-training and IDC fine-tuning, and 3) the visual features being "purely visual", i.e. they reside in the visual domain, far apart from the textual domain. In the following sections, we introduce CLIP4IDC, a CLIP-based approach to address these bottlenecks.

2.1 CLIP Fine-tuning Approach

An end-to-end approach of fine-tuning CLIP for IDC is shown in Figure 1b. Specifically, the image representations are generated by the vision encoder initialized with CLIP (Dosovitskiy et al., 2020) and are fed into a Transformer encoder to focus on accounting for the differences in the image pair. A Transformer decoder is applied to describe the changes given the visual context.

2.2 Model Architecture

Figure 2 sketches the CLIP4IDC model, containing the vision and language encoders.

Language Encoder. Given a textual caption T, the language encoder G consisting of N_G Transformer layers is used, denoted as:

$$G(T) = G(\{E_{\text{bos}}, E_{t_1}, ..., E_{t_m}, E_{\text{eos}}\} + p_{\text{T}}), \quad (1)$$

where $E_* \in \mathbb{R}^{d_T}$ is a linear projection of each token and $p_T \in \mathbb{R}^{(m+2) \times d_T}$ is a learned positional embedding to retain the positional information. E_{bos} and E_{eos} are token embeddings to represent the start and end of the text, respectively. The language encoder's output $g \in \mathbb{R}^{d_T}$ is generated by collecting the output of the token embedding E_{eos}.

Vision Encoder. Each image in the image pair (X^1, X^2) is patchified with the CLIP's initial convolutional layer into n image patches with dimen-
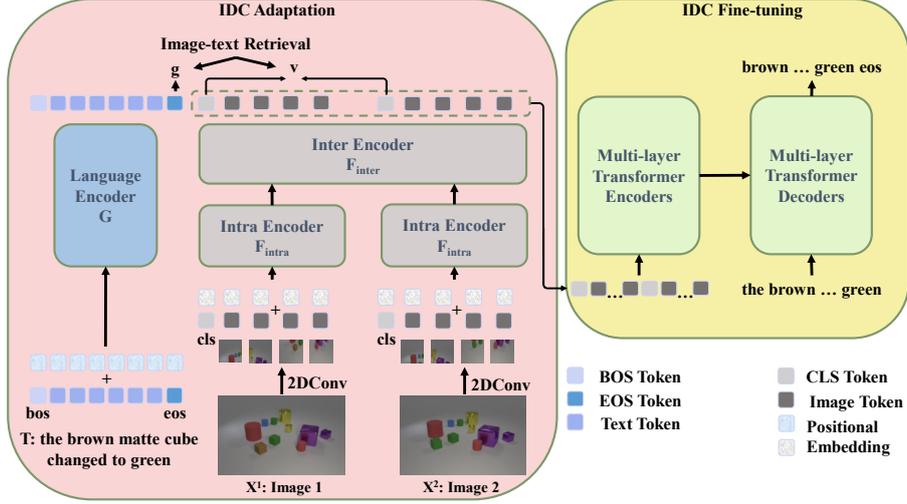

Figure 2: The detailed architecture of CLIP4IDC.

sionality d_I as:

$$X^1 = \{x_{\text{cls}}, x_1^1, ..., x_n^1\} + p_{\text{I}}, \quad (2)$$

$$X^2 = \{x_{\text{cls}}, x_1^2, ..., x_n^2\} + p_{\text{I}}, \quad (3)$$

where x_{cls} is a learned class embedding to represent the global context of the images and the positional embedding $p_{\text{I}} \in \mathbb{R}^{(n+1) \times d_I}$. $\{\cdots\}$ is the sequence of the embeddings. The vision encoder F is constructed to capture the subtle changes in the image pair. F is initialized by CLIP's weights and composed of a *intra* and *inter* Transformer modules. Specifically, the *intra* module F_{intra} containing N_{intra} Transformer layers learns the uni-modal context from the image pairs. The *inter* module F_{inter} with N_{inter} layers is constructed to focus on the subtle difference between the contexts in each pair. These procedures are formulated as:

$$F(X^1, X^2) = F_{inter}(\{F_{intra}(X^1) + e_1, \quad (4)$$
$$F_{intra}(X^2) + e_2\} + p),$$

where $p \in \mathbb{R}^{2(n+1) \times d_I}$. e_1 and $e_2 \in \mathbb{R}^{d_I}$ are special token embeddings to represent the first and second images. Afterwards, a learnable linear projection $W \in \mathbb{R}^{d_I \times d_T}$ is applied to the visual representation $F(X^1, X^2)$, on which the final visual representation $F'(X^1, X^2)$ is generated.

2.3 IDC-specific Adaptation

Next, we propose two novel IDC-specific pretext tasks, which are image-pair-to-text (IP-T) and text-to-image-pair (T-IP) retrieval, for better adapting the visual representations for captioning.

Prior to fine-tuning CLIP for the actual IDC task, we adapt the visual features to the domain of the IDC task via IP-T and T-IP retrieval. Our adaptation methodology follows the contrastive approach,

where the encoded image pairs are drawn closer to the encoded difference captions. Although other kinds of adaptation strategies exist, such as the one focusing more on matching the domain distributions (Tzeng et al., 2014), we only focus on testifying if adding such an adaptation step is useful. We aggregate a combined visual representation $v \in \mathbb{R}^{d_T}$ of the image pair from their x_{cls} embeddings, denoted as:

$$v = f(\{F'(X^1, X^2)_1, F'(X^1, X^2)_{n+2}\}), \quad (5)$$

where f is the mean-pooling operation. The subscript is the position (1-indexed) of the embeddings in the representation. Given B image pairs and difference captions in a batch, the target is to match $B \times B$ similarities between the difference representations of the image pairs and the descriptions to the differences. The loss function is defined as:

$$\mathcal{L}_{i2t} = \frac{-1}{B} \sum_i^B \log \frac{\exp(s(v_i, g_i)/\tau)}{\sum_{j=1}^B \exp(s(v_i, g_j)/\tau)}, \quad (6)$$

$$\mathcal{L}_{t2i} = \frac{-1}{B} \sum_i^B \log \frac{\exp(s(v_i, g_i)/\tau))}{\sum_{j=1}^B \exp(s(v_j, g_i)/\tau)}, \quad (7)$$

$$\mathcal{L} = \mathcal{L}_{i2t} + \mathcal{L}_{t2i}, \quad (8)$$

where $\mathcal{L}_{i2t}$ and $\mathcal{L}_{t2i}$ are the loss functions of IP-T and T-IP retrieval, respectively. $s(\cdot, \cdot)$ denotes the cosine similarity function and τ is a learnable temperature parameter to smooth the gradients.

2.4 Captioning

In the actual captioning stage, the vision encoder is initialized with the weights obtained from the previous adaptation stage and the output $F'(X^1, X^2)$ of the vision encoder is fed into the captioning model. As shown in Figure 2, the captioning model contains multi-layer Transformer encoders and de-

Model	Input	PT	B	M	C	R
Capt-Dual-Att (2019)	ResNet	–	43.5	32.7	108.5	–
DUDA (2019)	ResNet	–	47.3	33.9	112.0	–
VAM (2020)	ResNet	–	50.3	37.0	114.9	69.7
VAM+ (2020)	ResNet	–	51.3	37.8	115.8	70.4
IFDC (2021)	F-RCNN	–	49.2	32.5	118.7	69.1
DUDA+Aux (2021)	ResNet	–	51.2	37.7	115.4	70.5
VACC (2021)	ResNet	–	52.4	37.5	114.2	–
BiDiff (2022)	ResNet	–	54.2	38.3	118.1	–
IDC-PCL (2022)	ResNet	✓	51.2	36.2	128.9	71.7
CLIP4IDC	Raw	✓	56.9	38.4	150.7	76.4
CC-Full (2022)	Raw,ResNet	✓	64.3	36.4	151.4	77.1

Table 1: Results of IDC on CLEVR-Change test split. The main metric CIDer is highlighted. CC-Full is in a separate group as it adopts the policy gradient method directly optimized for the target metrics.

coders for the visual and textual representations, respectively. The decoder is trained to predict the next token given the previous ground truth words and the visual differences. A word-level cross entropy (XE) loss as in Park et al. (2019) is utilized.

3 Experiments

3.1 Benchmark Datasets and Metrics

We conduct experiments on CLEVR-Change (Park et al., 2019), Spot-the-Diff (Jhamtani and Berg-Kirkpatrick, 2018) and Image-Editing-Request (Tan et al., 2019) datasets. Following previous works, e.g. (Huang et al., 2021; Hosseinzadeh and Wang, 2021), captioning models are evaluated on BLEU (B) (Papineni et al., 2002), METEOR (M) (Banerjee and Lavie, 2005), CIDEr-D (C) (Vedantam et al., 2015) and ROUGE-L (R) (Lin, 2004) on the *test* split. IDC adaptation is done via image-pair-to-text (IP-T) and text-to-image-pair (T-IP) retrieval tasks. The standard retrieval metrics are reported: recall at rank K (R@K), median rank (MdR) and mean rank (MnR).

3.2 Captioning Results

We compare CLIP4IDC against the direct CLIP fine-tuning method and the state of the arts which employ the pre-extracted features in Tables 1– 4.

Results on CLEVR-Change. Table 1 shows that CLIP4IDC outperforms all the baselines except CC-Full (Ak et al., 2022) on CIDEr. Note that CC-Full employs the policy gradient method and is directly optimized for generating the target captions, while our proposed CLIP4IDC only relies on standard XE captioning loss. As such, we do not think their results are comparable, however, our results are still rather competitive. As we will see in a later section, CLIP4IDC significantly outperforms CC-Full on a real-world dataset.

Model	C	T	M	A	D	DI
DUDA (2019)	120.4	86.7	56.4	108.2	103.4	110.8
VAM+ (2020)	122.1	98.7	82.0	126.3	115.8	122.6
IFDC (2021)	133.2	99.1	82.1	128.2	118.5	114.2
DUDA+Aux (2021)	120.8	89.9	62.1	119.8	123.4	116.3
BiDiff (2022)	115.9	106.8	71.8	121.3	124.9	116.1
IDC-PCL (2022)	131.2	101.1	81.7	133.3	116.6	145.0
CLIP4IDC	149.1	135.3	91.0	132.4	135.5	133.4

Table 2: The breakdown of CIDEr score on different types of changes on CLEVR-Change test split. The columns C, T, M, A, D, DI stand for change types of Color, Texture, Move, Add, Drop and Distractor, i.e. no changes in the image pairs.

Model	Input	PT	B	M	C	R
DDLA (2018)	ResNet	–	8.5	12.0	32.8	28.6
DUDA (2019)	ResNet	–	8.1	11.5	34.0	28.3
VAM (2020)	ResNet	–	10.1	12.4	38.1	31.3
IFDC (2021)	F-RCNN	–	8.7	11.7	37.0	30.2
DUDA+Aux (2021)	ResNet	–	8.1	12.5	34.5	29.9
VACC (2021)	ResNet	–	9.7	12.6	41.5	32.1
CLIP4IDC	Raw	✓	11.6	14.2	47.4	35.0
CC-Full (2022)	Raw,ResNet	✓	8.3	13.0	33.0	30.0

Table 3: Results of IDC on Spot-the-Diff test split.

Model	Input	PT	B	M	C	R
Rel-Att (2019)	ResNet	–	6.7	12.8	26.4	37.4
DUDA (2019)	ResNet	–	6.5	12.4	22.8	37.3
BiDiff (2022)	ResNet	–	6.9	14.6	27.7	38.5
CLIP4IDC	Raw	✓	8.2	14.6	32.2	40.4

Table 4: Results on Image-Editing-Request test split.

Model	$\mathcal{L}$	Params	CLEVR-Change				Spot-the-Diff			
			B	M	C	R	B	M	C	R
CLIP-FT	–	135.57M	49.9	34.8	133.9	70.8	11.0	12.8	43.3	33.5
CLIP4IDC	–	135.65M	54.2	37.9	147.5	75.4	11.0	12.9	43.0	33.4
CLIP4IDC	✓	135.65M	56.9	38.4	150.7	76.4	11.6	14.2	47.4	35.0

Table 5: Ablation results of IDC on the two datasets.

We also assess the models by different types of changes on CLEVR-Change, as seen in Table 2. CLIP4IDC outperforms IDC-PCL on Color, Texture, Move and Drop types.

Results on Spot-the-Diff and Image-Editing-Request. Tables 3 and 4 show that CLIP4IDC achieves higher accuracy than the baselines on all the metrics on the two real datasets.

Ablations. We conduct ablation studies on different CLIP architectures and adaptation strategies. Table 5 shows that CLIP4IDC without the adaptation stage (without $\mathcal{L}$ in Eq. 8) outperforms the direct CLIP finetuning ("CLIP-FT") on CLEVR-Change. On the more challenging real-world dataset, Spot-the-Diff, we observe the same trend. Having the adaptation stage with $\mathcal{L}$ thus further enhances the performances. This confirms that learning to capture more fine-grained visual differences in the adaptation stage is beneficial.

	CLEVR-Change						Spot-the-Diff						Editing-Request					
	Image Pair ⇔ Text			Text ⇔ Image Pair			Image Pair ⇔ Text			Text ⇔ Image Pair			Image Pair ⇔ Text			Text ⇔ Image Pair		
Model	R@1	R@5	R@10	R@1	R@5	R@10	R@10	R@20	R@50	R@10	R@20	R@50	R@1	R@5	R@10	R@1	R@5	R@10
CLIP4IDC	46.4	83.0	86.6	26.8	58.7	70.0	3.7	7.3	16.8	6.2	10.5	20.0	17.1	28.4	33.8	17.3	33.7	41.9

Table 6: Results of IP-T and T-IP retrieval on the three datasets.

Model	N_{intra}	N_{inter}	Image Pair ⇒ Text					Text ⇒ Image Pair					Captioning			
			R@1	R@5	R@10	MdR↓	MnR↓	R@1	R@5	R@10	MdR↓	MnR↓	B	M	C	R
	6	6	46.1	79.8	83.9	2.0	49.6	26.4	57.1	68.4	4.0	29.4	54.0	37.4	146.5	75.2
	7	5	46.1	80.8	84.5	2.0	45.5	27.0	57.8	69.0	4.0	28.2	54.5	37.5	148.4	75.5
	8	4	47.2	80.7	84.4	2.0	46.3	27.7	58.7	69.7	4.0	29.9	54.1	37.4	147.3	75.4
CLIP4IDC	9	3	46.4	83.0	86.6	2.0	39.2	26.8	58.6	70.0	4.0	25.6	54.8	37.8	148.6	75.8
	10	2	37.5	68.5	73.9	2.0	88.8	22.9	52.3	63.9	5.0	54.4	51.5	35.4	134.6	71.5
	11	1	24.7	47.2	53.3	7.0	143.6	17.8	40.2	50.9	10.0	84.8	45.0	32.7	122.8	67.9
	12	0	2.3	7.0	11.8	182.0	459.9	1.1	3.9	5.9	419.0	716.5	38.8	29.5	90.9	60.6

Table 7: Results of setting different number of layers in CLIP4IDC on the IP-T, T-IP retrieval and IDC tasks on CLEVR-Change test split.

3.3 Adaptation Results

We report the results in the retrieval tasks used for adaptation in Table 6 on the test splits of the three datasets. These results from the image-pair and text retrieval tasks are simply to testify the model's capability of capturing details in the image pairs. The effects brought by the retrieval tasks on the captioning accuracy are assessed in the following.

4 Assessments of IDC Adaptation

We study how the retrieval accuracy is affected by different architectural options in CLIP4IDC on CLEVR-Change test split. Table 7 shows the effect of setting different numbers of layers in the *intra* and *inter* modules. It can be seen that the improvement is achieved by allocating a large number of layers to the *intra* module. However, it does not mean that *inter* layers are not required, as shown in the decreased accuracy when cutting the number of inter layers. In addition, when the *inter* layers are removed, i.e. $N_{inter} = 0$, the architecture is similar to Luo et al. (2021) and its accuracy is greatly reduced. We owe it to the fact that the global information represented by two separate image embeddings fails to localize the changes between them.

To further study the relationships between the retrieval-based adaptation and the captioning accuracy, we fine-tune the models from the adaptation stage on the captioning task with the frozen image encoder. It can be observed in Table 7 that, in general, better adaptation with higher recall values on the retrieval tasks translates to better captioning. The observation suggests that the introduced retrieval tasks and the metrics used for retrieval serve as a strong indicator of the IDC performance.

5 Conclusion and Future Work

In this work, we studied how to fine-tune CLIP for image difference captioning. Retrieval-based adaptation was introduced to improve the visual representations for captioning and to narrow the gap between the purposes and data domains of CLIP pre-training and IDC. Experimental results demonstrated the effectiveness of the CLIP4IDC model and the applied domain adaptation.

In the future work, we will further explore enhancing the relationships between the vision and language domains. Specifically, CLIP4IDC adapts CLIP which does not involve cross-modal interactions as early as other pre-trained VL models (Lu et al., 2019; Su et al., 2019; Li et al., 2019) that allow the interactions from the ground up. Adapting other VL models for IDC is naturally one interesting future direction. Moreover, exploring other means than our contrastive approach, such as domain confusion (Tzeng et al., 2014), to bridge vision and language domains is another plausible direction.

References

Kenan Emir Ak, Ying Sun, and Joo Hwee Lim. 2022. Learning by imagination: A joint framework for text-based image manipulation and change captioning. *IEEE Transactions on Multimedia*.

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Mehrdad Hosseinzadeh and Yang Wang. 2021. Image change captioning by learning from an auxiliary task. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2725–2734.

Qingbao Huang, Yu Liang, Jielong Wei, Cai Yi, Hanyu Liang, Ho-fung Leung, and Qing Li. 2021. Image difference captioning with instance-level fine-grained feature representation. *IEEE Transactions on Multimedia*.

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

Hoeseong Kim, Jongseok Kim, Hyungseok Lee, Hyunsung Park, and Gunhee Kim. 2021. Agnostic change captioning with cycle consistency. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2095–2104.

Jie Lei, Linjie Li, Luowei Zhou, Zhe Gan, Tamara L Berg, Mohit Bansal, and Jingjing Liu. 2021. Less is more: Clipbert for video-and-language learning via sparse sampling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7331–7341.

Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. 2021. Align before fuse: Vision and language representation learning with momentum distillation. *Advances in neural information processing systems*, 34:9694–9705.

Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. 2019. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Advances in neural information processing systems*, 32.

Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. 2021. Clip4clip: An empirical study of clip for end to end video clip retrieval. *arXiv preprint arXiv:2104.08860*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Dong Huk Park, Trevor Darrell, and Anna Rohrbach. 2019. Robust change captioning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4624–4633.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28.

Xiangxi Shi, Xu Yang, Jiuxiang Gu, Shafiq Joty, and Jianfei Cai. 2020. Finding it at another side: A viewpoint-adapted matching encoder for change captioning. In *European Conference on Computer Vision*, pages 574–590. Springer.

Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. 2019. Vl-bert: Pre-training of generic visual-linguistic representations. In *International Conference on Learning Representations*.

Yaoqi Sun, Liang Li, Tingting Yao, Tongyv Lu, Bolun Zheng, Chenggang Yan, Hua Zhang, Yongjun Bao, Guiguang Ding, and Gregory Slabaugh. 2022. Bidirectional difference locating and semantic consistency reasoning for change captioning. *International Journal of Intelligent Systems*, 37(5):2969–2987.

Hao Tan, Franck Dernoncourt, Zhe Lin, Trung Bui, and Mohit Bansal. 2019. Expressing visual relationships via language. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1873–1883.

Mingkang Tang, Zhanyu Wang, Zhenhua Liu, Fengyun Rao, Dian Li, and Xiu Li. 2021. Clip4caption: Clip for video caption. In *Proceedings of the 29th ACM*

38

International Conference on Multimedia, pages 4858–4862.

Eric Tzeng, Judy Hoffman, Ning Zhang, Kate Saenko, and Trevor Darrell. 2014. Deep domain confusion: Maximizing for domain invariance. *arXiv preprint arXiv:1412.3474*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. 2015. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4566–4575.

Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. 2015. Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3156–3164.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057. PMLR.

Linli Yao, Weiying Wang, and Qin Jin. 2022. Image difference captioning with pre-training and contrastive learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3108–3116.

Luowei Zhou, Hamid Palangi, Lei Zhang, Houdong Hu, Jason Corso, and Jianfeng Gao. 2020. Unified vision-language pre-training for image captioning and vqa. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 13041–13049.

A Dataset

CLEVR-Change (Park et al., 2019) is a synthetic dataset generated by CLEVR engine. Geometric differences between the objects in the images are annotated. It is divided into the training, validation and test splits which have 67,660, 3,976 and 7,970 image pairs, respectively. Spot-the-Diff (Jhamtani and Berg-Kirkpatrick, 2018) describes multiple scene changes in the real 13,192 image pairs sampled from the VIRAT Ground Video Dataset with human-annotated captions. On an average, there are 1.86 sentences to describe the differences for each image pair. Two decoding strategies containing single-sentence decoding and multi-sentence decoding are set for captioning. Following Jhamtani and Berg-Kirkpatrick (2018), we evaluate models in the single-sentence decoding by setting the ground truth description as multiple reference captions. Image-Editing-Request (Tan et al., 2019) is a dataset consisted of camera shots, paintings and animations, and most of the images are realistic. It contains 3,939 image pairs with instructions written by human annotators.

B Implementation Details

IDC Adaptation Settings. The vision and language encoders are initialized with CLIP ViT-B/32 (Dosovitskiy et al., 2020). The sentence length is 32 and the number of layers in the language encoder $N_G = 12$. The dimension of the text embedding $d_T = 512$. The size of an image is 224×224 and each image is processed by a 2D convolution network with kernel size 32, stride 32 and 768 channels. The number of image patches $n = 49$ and the dimension of image patches $d_I = 768$. The number of layers in the intra- and inter-Transformer modules are $N_{intra} = 9$ and $N_{inter} = 3$, respectively. Adam optimizer is applied with initial learning rate 10^{-7}. The models are trained for 12 epochs by fixing all the random seeds to 42 on two NVIDIA Tesla V100 GPUs.

IDC Fine-tuning Settings. We initialize the vision encoder with the model from IDC adaptation and set the dimensionality of the word embedding $d_T = 512$. The captioning model is learned from scratch. The number of Transformer layers in both captioning encoder and decoder is 3 on all the datasets. The attention layer in the Transformer has 8 heads and 10% dropout probability, and its hidden size is 512.

For the direct CLIP fine-tuning, the parameters of its vision encoder are initialized with CLIP ViT-B/32. The settings of its captioning model are the same as those in CLIP4IDC.

Adam is used with initial learning rate 10^{-7} for the vision encoder and 10^{-4} for the captioning model. The model is trained for at most 50 epochs and the batch size is 16. Greedy decoding with maximum 32 steps is applied for generating sentences in inference. The experiments are carried out on a NVIDIA Tesla V100 GPU.

C Qualitative Results

To understand the effect of IDC adaptation, some cases on CLEVR-Change, Spot-the-Diff and Image-Editting-Request datasets are visualized in Figures 3, 4 and 5, respectively.

GT: the blue ball changed to yellow
CLIP4IDC: the blue ball became yellow

GT: the big purple metal block behind the green thing
changed to rubber
CLIP4IDC: the large purple metal block that is behind the
big purple metal sphere became rubber

GT: there is no difference
CLIP4IDC: there is no change

GT: there is no change
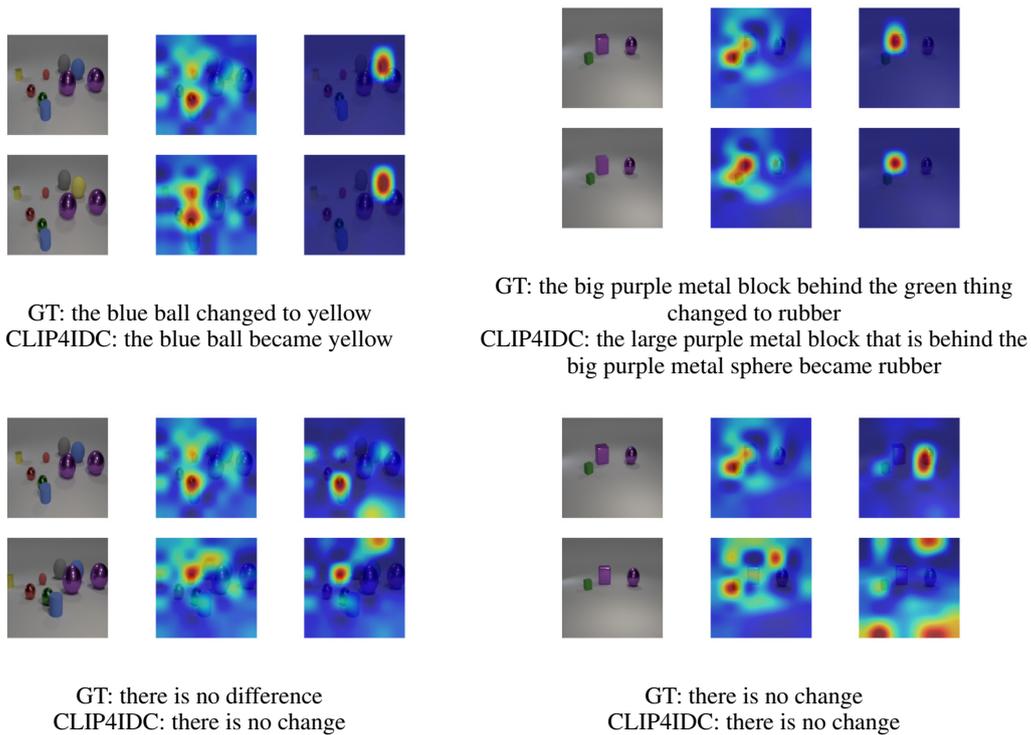CLIP4IDC: there is no change

Figure 3: Visualization of the vision encoder's output in CLIP4IDC on CLEVR-Change. Figures are arranged in three columns. The first column shows the first and the second raw images. The second column shows their attention maps in the intra-encoder's output. The last column shows their attention maps in the inter-encoder's output.
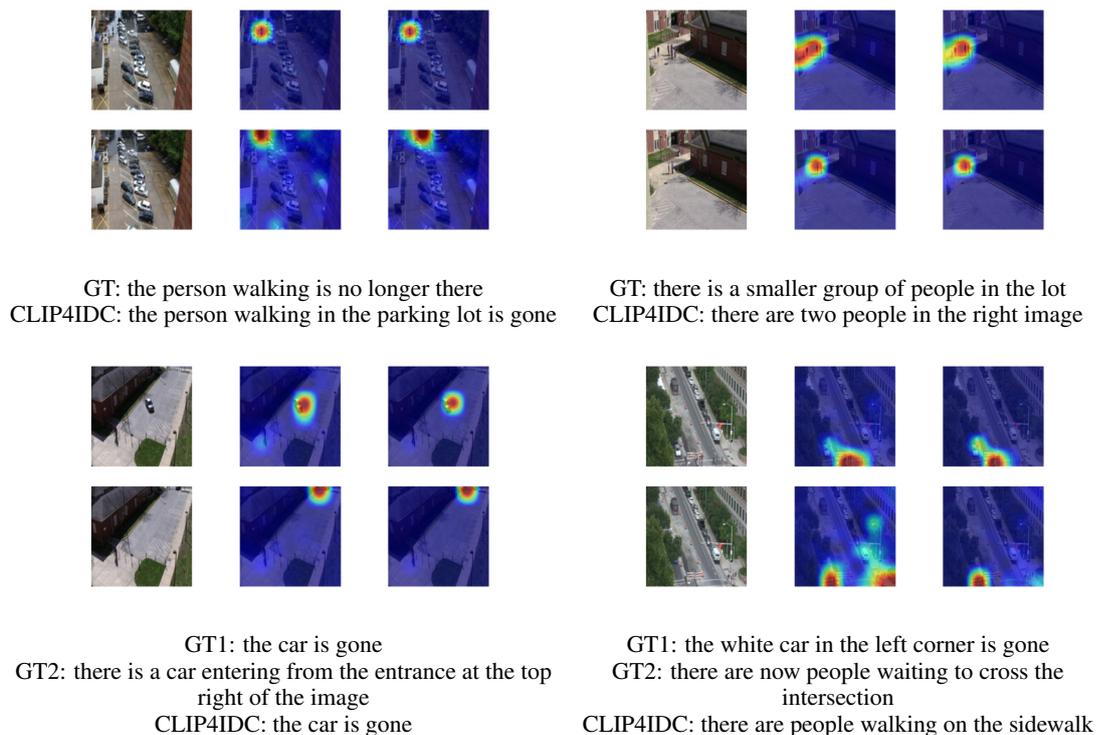

GT: the person walking is no longer there
CLIP4IDC: the person walking in the parking lot is gone

GT: there is a smaller group of people in the lot
CLIP4IDC: there are two people in the right image

GT1: the car is gone
GT2: there is a car entering from the entrance at the top
right of the image
CLIP4IDC: the car is gone

GT1: the white car in the left corner is gone
GT2: there are now people waiting to cross the
intersection
CLIP4IDC: there are people walking on the sidewalk

Figure 4: Visualization of the vision encoder's output in CLIP4IDC on Spot-the-Diff.

GT: color the sky blue
CLIP4IDC: make the image more blue

GT: brighten the entire photo
CLIP4IDC: brighten the photo

GT: remove girl in background
CLIP4IDC: remove the people from the background

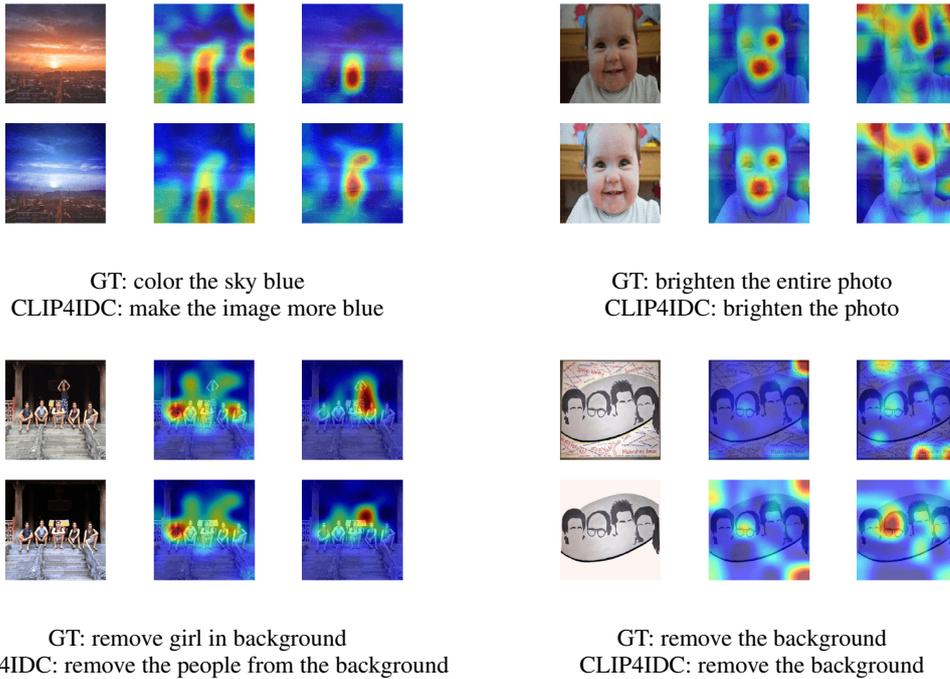GT: remove the background
CLIP4IDC: remove the background

Figure 5: Visualization of the vision encoder's output in CLIP4IDC on Image-Editting-Request.

Synthetic Dataset The four cases in Figure 3 are from CLEVR-Change. In the second column of each case, it can be seen that CLIP4IDC's *intra* encoder attends to regions where information is more likely to be needed for capturing the fine-grained difference in the second images. While in the third column of them, *inter* encoder filters the information uncorrelated to the difference and pay attention to the changes in the second image. However, the condition is different for the cases, shown in the bottom two sets of figures, without changes. The *inter* encoder appears to attend more uniformly across regions to seek for any change instead of getting fixated on one specific region.

Real-world Dataset Figures 4 and 5 show the cases from Spot-the-Diff and Image-Editing-Request, respectively. It can be seen that our CLIP4IDC capture the fine-grained differences in the real-world and complicated cases.

D Descriptions of the Baseline Methods

Some recent works have made great progress in the IDC task by devising a language model that describes the changes, given the visual features pre-extracted by the CNN backbones (He et al., 2016; Ren et al., 2015). We describe the baselines we compare against in the experiments as follows:

- **DUDA** (2019): A dual attention module is pro-

posed to distinguish distractors from semantic changes and localize the changes. A dynamic attention module is then used to describe the changes.

- **VAM** (2020): A novel visual encoder is proposed to distinguish viewpoint changes from semantic changes. Moreover, it fine-tunes the model directly with reinforcement learning in which the rewards coming from evaluating the generated captions.
- **IFDC** (2021): A language generator, which consists of a feature fusion module, a similarity-based difference finding module, and a difference captioning module, is introduced.
- **VACC** (2021): A difference encoder is devised to encode viewpoint information and model the difference.
- **BiDiff** (2022): A change captioning pipeline is introduced to localize the changes in the image pair and a decoder with spatial-channel attention to generate descriptions.

These methods consistently improve the model accuracy by refining or improving the visual features to better capture the fine-grained changes in the image pair. In addition, inspired by the success of multi-task learning, the following training schemes were also introduced.

- **VACC** (2021) and **DUDA+Aux** (2021): Both work proposed auxiliary modules to match the composite feature of the generated caption and before image with the after image feature.
- **IDC-PCL** (2022): A "pretrain-and-finetune" paradigm is proposed and contains three pretraining tasks as follows. Given visual-linguistic contexts, the Masked Language Modelling (MLM) and Masked Visual Contrastive Learning (MVCL) tasks were applied to map the visual context to language and to reconstruct the masked image features, respectively. Fine-grained Difference Aligning (FDA) was introduced to rewrite the captions as the hard samples to maximize the connections in the joint representation of the text and the image pair.
- **CC-Full** (2022): The work proposed to co-train text-based image manipulation (TIM) with change captioning (CC) modules. The CC module generates captions evaluated with the TIM module with a reinforcement learning framework. The TIM module generates images that are evaluated with the CC module with a generative adversarial network.