Image Change Captioning by Learning from an Auxiliary Task

Mehrdad Hosseinzadeh[1] and Yang Wang[1,2]
University of Manitoba[1], Huawei Technologies Canada[2]
{mehrdad, ywang}@cs.umanitoba.ca

Abstract

We tackle the challenging task of image change captioning. The goal is to describe the subtle difference between two very similar images by generating a sentence caption. While the recent methods mainly focus on proposing new model architectures for this problem, we instead focus on an alternative training scheme. Inspired by the success of multi-task learning, we formulate a training scheme that uses an auxiliary task to improve the training of the change captioning network. We argue that the task of composed query image retrieval is a natural choice as the auxiliary task. Given two almost similar images as the input, the primary network generates a caption describing the fine change between those two images. Next, the auxiliary network is provided with the generated caption and one of those two images. It then tries to pick the second image among a set of candidates. This forces the primary network to generate detailed and precise captions via having an extra supervision loss by the auxiliary network. Furthermore, we propose a new scheme for selecting a negative set of candidates for the retrieval task that can effectively improve the performance. We show that the proposed training strategy performs well on the task of change captioning on benchmark datasets.

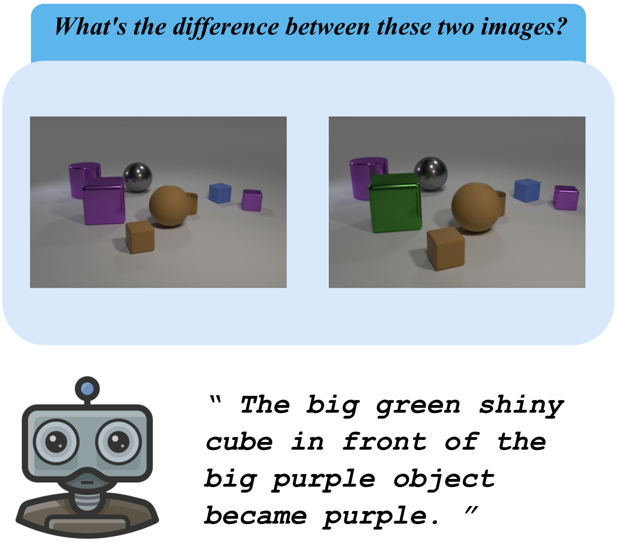

What's the difference between these two images?
" The big green shiny cube in front of the big purple object became purple. "

Figure 1: (Best viewed in color) Given two very similar images, the goal of change captioning is to describe the subtle difference between these two images. The difference can be in terms of objects' color, texture, position, addition or removal, etc.

1. Introduction

Change is an inevitable part of a dynamic environment. There has been much attention in the community for a variety of change detection tasks [10, 27, 28, 13, 22, 23, 31]. While localizing the change has been the cornerstone of these works, it requires a deeper level of understanding to be able to semantically refer to the change. From a user's perspective, describing (captioning) the change between two images provides a more meaningful way of understanding the difference between images. The task of change captioning aims to describe the change between two images by generating a detailed sentence about the change of objects in these images. Note that in this task, we are only interested in changes at the object level (*e.g.* changes in terms of object color, position, etc), *i.e.* we do not want to generate captions for changes in terms of viewpoints, light condition, etc. Some early works [13, 22, 23] in this area assume that there is always an object change between a pair of input images. This assumption does not always hold in a real-world application. In many cases, no object has been changed between the two images. Instead, it is only the viewpoint that is different (*e.g.* a moving robot can see a scene from two different viewpoints and should be able to distinguish that it is the same scene observed from different angles/lighting conditions). To address this limitation, recent work [25] proposes the task of robust changing captioning where not all pairs of images exhibit a change – some pairs can be distractors with similar scenes from different viewpoints.

In this paper, we also consider the problem of change captioning with distractors. Our goal is to detect a change between a pair of images and generate a sentence describing the change. While previous works mainly focus on proposing new network architectures to better tackle the problem [31, 25], we instead focus on improved learning strategies via multi-task learning. Our proposed method consists of networks for two complementary tasks, namely the primary task and the auxiliary task.

The *primary task* in our formulation is the task of change captioning. Given two images, the goal of the primary task is to describe the difference between these two images. The input images are similar to each other and only differ in very subtle ways (*e.g.* an object's color, shape, or position is changed).

The *auxiliary task* in our formulation is the composed query image retrieval [37, 11, 8, 3]. This is an extension of the image retrieval task. The input to this task consists of a reference image and a sentence that defines the user's desired modification on the reference image. The model should then pick a candidate among a set of images. The candidate should look like the reference image, but differ from it according to the desired modification.

We argue that the aforementioned two tasks are naturally complimentary to each other. As a result, we can use the auxiliary task to help the primary task during learning. We propose a joint learning scheme for these two tasks. Inspired by the works on cycle-consistency [42, 29, 9], our proposed learning scheme sequentially performs the primary and the auxiliary tasks in a way that the output of one task is the input to the other task. These two tasks can reinforce each other during training. For example, the learning scheme forces the primary task to generate a good caption to be used as the input to the auxiliary network. If the caption generated by the primary network is not reliable, the auxiliary network would fail to retrieve the correct image. This will produce an auxiliary loss which is then used to further train the primary network.

The contributions of this paper are manifold.

- First, we propose a new learning scheme for the task of change image captioning that involves using an auxiliary task to improve the performance of the primary task.
- Second, we further improve our proposed learning scheme by defining a new strategy for selecting hard negative samples for the auxiliary task of composed query image retrieval.
- Finally, we show that our proposed method can improve the performance of a change captioning task by performing empirical experiments on the CLEVR-Change dataset and the Spot-the-diff dataset.

2. Related Work

Our work is related to two lines of research, namely, image captioning and change detection. We briefly review relevant work on these topics.

Image Captioning: Image captioning [22, 39, 1, 40, 2, 12, 5, 4] has been studied extensively in recent years. Vinyals et al. [36] propose one of the earliest methods for image captioning which utilize LSTM modules. Xu et al. [39] extend the former model by introducing visual attention. Visual attention has been proved to be an effective technique for the image captioning models and has been used extensively since then [5, 4, 12, 24].

While early works on image captioning mainly use recurrent modules (specifically LSTM layers) for the caption generation, there has also been work on using the convolutional or self-attention mechanism. Anega et al. [2] replace the LSTM module with a fully convolutional module that noticeably improves the training time of these networks. With the introduction of transformers and self-attention layers, BERT-based models [35, 7] have been shown to be successful for a variety of language tasks. These model have also been applied in vision-language tasks as well [32, 4, 12, 17, 18, 38].

Our method is closely related to standard image captioning. But in our setting, the input is a pair of images instead of a single image. Our goal is to generate the caption to describe the difference between these two images rather than describing the content of a single image.

Change Detection: Detecting changes in an environment has been an active field of study in computer vision. There has been work on finding the change between two (or more) images. Change detection has been applied in aerial and satellite imagery [20, 34, 41] for applications such as natural disaster management [10], observing land dynamics [15], etc.

Most change detection tasks focus on finding changes in the pixel space. However, if we want the system to interact with users, it is more favorable to present the detected change in a human-readable form, such as describing the change using natural language. This has led to a new line of research called the image change captioning [13]. Early works in this area use the Spot-the-diff dataset [13] which has two major flaws: 1) it assumes that there is always a change between each pair of images and 2) it is a relatively small dataset (∼13K images in total). To overcome this issue, Park et al. [25] propose a new problem called the robust change captioning and introduce a new larger dataset called the CLEVR-Change that is based on the popular CLEVR engine [14]. This dataset better evaluates the performance of change captioning systems since some of the pairs in the dataset are the same image from different viewpoints. They also propose an attention-based model called DUDA. Recently, Shi et al. [31] propose a method to simulate human visual attention for better localization of the change. While these methods focus on proposing new network architectures for change captioning, we propose a new training technique that can be applied to any change captioning system.

3. Background

In this section, we provide some background on the image change captioning task and the composed query image retrieval task. Our proposed approach uses the composed query image retrieval as an auxiliary task to improve the performance of the primary image change captioning task.

Image Change Captioning: As mentioned in Sec. 1, the task of image change captioning is to generate a caption that describes the subtle but important change between two very similar images. Formally, given a pair of images (A, B), a model generates a caption describing what has been changed between A and B:

$$f(A, B; \theta_{\mathcal{P}}) \rightarrow \hat{C} \tag{1}$$

where $\theta_{\mathcal{P}}$ denotes the model parameters of the change captioning network and $\hat{C}$ represents the generated caption.

We use the Dual Dynamic Attention (DUDA) model [25] as the image change captioning network in our work. Although some recent work [31] has reported better performance, the code is not available yet. So we choose to build our approach based on DUDA. Here we briefly describe the network. DUDA consists of 3 major modules: a feature extraction module, a dual attention visual module, and a caption generator. The feature extraction module is a ResNet model trained offline and its weights are frozen. The feature extractor takes two images A and B as the input. It then produces 2D feature maps, $A_f, B_f \in \mathbb{R}^{d_v \times H \times W}$. A dual attention module is then applied on (A_f, B_f) and produces three feature maps capturing the visual information of the two images (A, B), and the difference between A and B. An LSTM-based captioning module then generates the words in the caption based on these three feature maps.

Composed Query Image Retrieval: This task can be seen as the opposite of the image change captioning task. Given an image and a caption describing some desired modification, the goal of this task is to retrieve the result of the modification among a set of image candidates [37, 11]. More formally, given an image A and a text sentence C describing the desired modification to be applied on A, the goal is to retrieve an image $\hat{B}$ from a candidate set S.

$$g(A, C|S; \theta_{\mathcal{A}}) \rightarrow \hat{B} \tag{2}$$

Let B be the resulting image of applying the modification C on the image A. Ideally, B and $\hat{B}$ should be the same image. We use a modified version of TIRG [37] model as our network for the composed query image retrieval. TIRG encodes the input image (A) and the modification text (C) using CNN and LSTM, respectively. Next, the textual feature is added to the visual feature to generate a single feature vector representing the composed query. The feature vector for the composed query and the visual feature vector of each candidate image $I \in S$ are then projected onto a common feature space. Using a nearest-neighbor strategy, the closest candidate image to the composed query is selected and retrieved.

4. Our Approach

Our proposed approach is based on the following key observation. Image change captioning and composed query image retrieval are two closely related problems. In this paper, we call them the primary task and the auxiliary task, respectively. If we have good models for both tasks, these two models should have the following cycle consistency [42]. Let (A, B, C) be a sample triplet in the training set where A and B are two images that are almost identical to each other but differ in very subtle details, and C is a sentence describing the subtle difference. Suppose we feed (A, B) to the primary network (image change captioning) and produce $\hat{C}$ as the output. We then feed $(A, \hat{C})$ to the auxiliary network (composed query image retrieval) and produces $\hat{B}$. We would expect B and $\hat{B}$ to be close. Similarly, if we first feed (A, C) to the auxiliary network to produce $\hat{B}$, then feed $(A, \hat{B})$ to obtain $\hat{C}$, we would expect C and $\hat{C}$ to be close (Fig. 2). Based on this observation, we jointly train these two networks. Note that the training dataset for the primary task can be easily re-purposed for the auxiliary task, so we do not need extra training data for the auxiliary task.

4.1. Joint Primary and Auxiliary Networks

The main novelty of this work is that we propose a new approach to train the primary network by coupling it with the auxiliary network. We use the DUDA and TIRG (see Sec. 3) as our primary and auxiliary networks, respectively. But our proposed method is not limited to these specific choices. It can be applied with any other network architectures for image chance captioning or composed query retrieval, respectively.

Based on recent advances in using auxiliary task learning to improve the primary task [19, 33, 21, 30], we propose to couple the primary and the auxiliary tasks, then train them jointly. Given a (A, B, C) triplet representing two images (A, B) and a caption C describing the difference between these two images, our proposed method involves two stages for training.

Primary → Auxiliary: The first stage involves feeding the image input (A, B) into the primary network and using the generated caption along with one of the images A as the

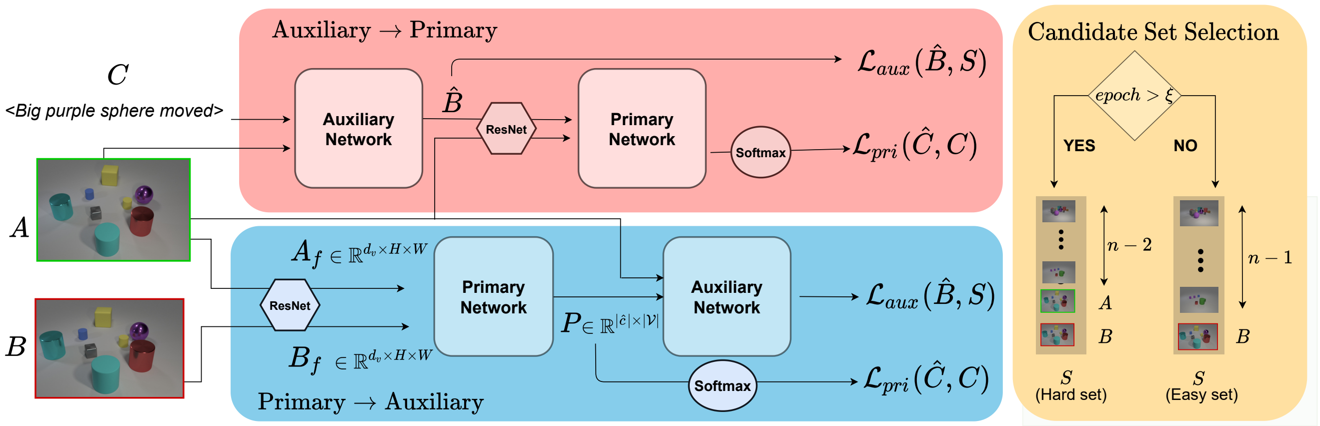

Auxiliary → Primary
C
<Big purple sphere moved>
Auxiliary Network
$\hat{B}$
ResNet
Primary Network
Softmax
$\mathcal{L}_{aux}(\hat{B}, S)$
$\mathcal{L}_{pri}(\hat{C}, C)$
A
B
$A_f \in \mathbb{R}^{d_v \times H \times W}$
ResNet
$B_f \in \mathbb{R}^{d_v \times H \times W}$
Primary Network
$P \in \mathbb{R}^{|\hat{c}| \times |\mathcal{V}|}$
Auxiliary Network
Softmax
$\mathcal{L}_{aux}(\hat{B}, S)$
$\mathcal{L}_{pri}(\hat{C}, C)$
Primary → Auxiliary
Candidate Set Selection
$epoch > \xi$
YES
NO
$n-2$
A
B
$n-1$
B
S (Hard set)
S (Easy set)

Figure 2: Overview of our approach. Given a triplet (A, B, C) from the training set, where (A, B) are image pairs and C is the caption describing their difference, our method involves jointly training two networks for the primary and the auxiliary tasks. The training involves two stages. In the first stage ("Primary → Auxiliary"), we feed (A, B) to the primary network to generate a caption $\hat{C}$, then feed $(A, \hat{C})$ as the input to the auxiliary network. In the second stage, we feed (A, C) to the auxiliary network to retrieve $\hat{B}$ from a set of S candidate images, then feed $(A, \hat{B})$ to the primary network. These two stages form a cycle consistency. The candidate set S is constructed differently depending on the training epoch. See the main text for details.

input to the auxiliary network. More specifically, consider $\theta_{\mathcal{P}}$ and $\theta_{\mathcal{A}}$ to be the parameters of primary and auxiliary networks, respectively. We first input A and B to the primary network. The primary network generates a caption $\hat{C} = \{\hat{w}^i\}_{i=0}^{|\hat{C}|}$ where $\hat{w}^i$ is the i-th word in the caption. Each word $\hat{w}^i$ is chosen from a vocabulary $\mathcal{V}$. We use P to denote the softmax scores for each word at each time step as the output of the primary network:

$$P = f(A, B; \theta_{\mathcal{P}}) \tag{3}$$

where $P \in \mathbb{R}^{|\hat{C}| \times |\mathcal{V}|}$ is a matrix representing the probability score of each word in the vocabulary being selected as the word at each time step in the generated caption. In other words, $\hat{C}$ is obtained by picking the word with the highest probability at each time step according to P.

Let $\mathcal{E}_{\hat{C}}$ be a matrix representing the embedding of each word in the caption $\hat{C}$. The input to the auxiliary network is then $(A, \mathcal{E}_{\hat{C}})$. The auxiliary network has a visual-textual module $k(A, \mathcal{E}_{\hat{C}}; \theta_{\mathcal{K}})$ which extracts a feature vector of $(A, \mathcal{E}_{\hat{C}})$. For a candidate image B for retrieval, the auxiliary network uses an image encoder $g(B; \theta_{\mathcal{G}})$ to extract a feature vector of B. Note that $k(\cdot, \cdot; \theta_{\mathcal{K}})$ and $g(\cdot; \theta_{\mathcal{G}})$ have the same dimensions. The similarity between $k(A, \mathcal{E}_{\hat{C}}; \theta_{\mathcal{K}})$ and $g(B; \theta_{\mathcal{G}})$ is used to measure how good B is as the retrieved result.

Note that if we naively feed discrete words in $\hat{C}$ to an LSTM module to get the embedding vector $\mathcal{E}_{\hat{C}}$, this will make the pipeline non-differentiable. As a result, the learning signal will not be able to propagate from the auxiliary network to the primary network. To address this issue, we propose to use a soft word selection based on their softmax scores (i.e. P) instead of words themselves. Let $W_{emb} \in \mathbb{R}^{|\mathcal{V}| \times d_e}$ be the word embedding matrix for the auxiliary network where d_e is the dimension of the word embedding, we define a soft word embedding layer as follows:

$$\mathcal{E}_{\hat{C}} = P \times W_{emb} \tag{4}$$

where $\mathcal{E}_{\hat{C}} \in \mathbb{R}^{|\hat{C}| \times d_e}$ is the soft embedding of words in the generated caption.

We then use the image A and the generated caption $\hat{C}$ (represented as $\mathcal{E}_{\hat{C}}$) as the input to the auxiliary network. The auxliary network also has access to a set of n candidate images $S = \{I_j\}_{j=1}^{n-1} \cup B$. The goal of the auxiliary network is to retrieve one image from S that best matches $(A, \hat{C})$. This can be achieved using a batch-based classification [37]. Let $\theta_{\mathcal{A}} = \{\theta_{\mathcal{G}}, \theta_{\mathcal{K}}\}$ be the parameters of the auxiliary network. We are given a batch β of training examples. Each sample i in the batch has the form $(A_i, \mathcal{E}_i, B_i)$ where

$$\mathcal{E}_i = f(A_i, B_i; \theta_{\mathcal{P}}) \times W_{emb}, \;\; i = 1, 2, ..., \beta \tag{5}$$

We can define the following batch-based classification loss:

$$\mathcal{L}_{aux} = \frac{1}{\beta} \sum_{i=1}^{\beta} -\log\left\{ \frac{e^{\left\langle k(A_i, \mathcal{E}_i; \theta_{\mathcal{K}}), g(B_i; \theta_{\mathcal{G}}) \right\rangle}}{\sum_{j=1}^{\beta} e^{\left\langle k(A_i, \mathcal{E}_i; \theta_{\mathcal{K}}), g(B_j; \theta_{\mathcal{G}}) \right\rangle}} \right\} \tag{6}$$

where $< \cdot, \cdot >$ is the dot-product of two vectors as the similarity measure.

Auxiliary → **Primary**: The second stage for the training starts with the auxiliary network. Given (A, C) where C is the ground-truth caption describing the difference between A and B, the auxiliary network tries to pick B among a set of candidate images $S = \{I_j\}_{j=1}^{n-1} \cup B$. However, since the hard selection operation is not differentiable, we adopt a soft selection strategy as follows. We first compute the joint representation of (A, C) using the multi-modal module in the auxiliary network:

$$R = k(A, \mathcal{E}_C; \theta_{\mathcal{K}}) \tag{7}$$

where $\mathcal{E}_C$ is a matrix representing the embedding of words in the caption C by applying an embedding layer with weight W_{emb}, and $R \in \mathbb{R}^{d_r}$ is the joint representation vector. We also encode each $I_j \in S$ using the visual module of the auxiliary network to obtain $\tilde{I}_j$:

$$\tilde{I}_j = g(I_j; \theta_{\mathcal{G}}) \quad \forall I_j \in S \tag{8}$$

where $\hat{I}_j \in \mathbb{R}^{d_r}$. We define $\tilde{S}$ as the matrix representing encoded images in the candidate set, i.e. $\tilde{S} = \{\tilde{I}\}_{j=1}^{n} \in \mathbb{R}^{n \times d_r}$. The soft selection is calculated by first computing a set of n weights denoted by ω:

$$\omega = Softmax(\tilde{S} \cdot R) \tag{9}$$

where $\omega \in \mathbb{R}^n$. We now softly select (generate) from the candidate set using weights calculated above:

$$\hat{B} = \sum_{j=1}^{n} \omega_j \tilde{I}_j \tag{10}$$

We now feed $(A, \hat{B})$ as input to the primary network. The output of the primary network at each time step i is a $|\mathcal{V}|$-sized vector p_i representing the Softmax scores for each word in the vocabulary. To generate the predicted caption, we can take the word that maximizes the score as the predicted word in i-th time step according to p_i. To calculate the primary network's loss function during training, we use the negative likelihood of the words in the ground-truth caption according to the Softmax output:

$$\mathcal{L}_{pri} = -\sum_{w^i \in C} log(p_i(w^i)) \tag{11}$$

$$C = \{w^1, \cdots, w^{C_n}\} \quad C_n = |C| \tag{12}$$

4.2. Model Training

We jointly train primary and auxiliary networks end-to-end for 60 epochs using the Adam optimizer [16]. The learning rate is set to $5e-3$. Following [25], we also use a pre-trained ResNet 101 trained on ImageNet [6] as our feature extractor for the primary network. Other modules are trained from scratch. We alternate between the two stages from one batch to another.

Negative Sample Selection: Selecting the right candidate set for the auxiliary network is crucial since it has a direct effect on the primary network. Selecting an easy-to-pick set of candidates will cause the auxiliary network to converge quickly. This will cause $\mathcal{L}_{aux}$ to diminish fast and provide little gradient for the primary network to train. On the other hand, a very hard set of candidates at the beginning of the training prevents the network from converging since $\mathcal{L}_{aux}$ will be high.

To circumvent this issue, we propose the following curriculum learning strategy. In the early epochs of training, we provide the auxiliary network with a relatively easy set of candidates to choose from. In the later epochs, we switch to a harder set of images. More specifically, during early epochs, we use $n-1$ random images plus B to form the candidate set S. This helps the model to first learn to distinguish between B and other images in S, *i.e.* $S = \{I_j\}_{j=1}^{n-1} \cup B$. Once the model has learned this task ($\mathcal{L}_{aux}$ becomes small), we construct S differently. We randomly select $n-2$ and add to them both A and B to form S, *i.e.* $S = \{I_j\}_{j=1}^{n-2} \cup \{A, B\}$. So we have:

$$\begin{cases} S = \{I_j\}_{j=1}^{n-1} \cup B & epoch < \xi \\ \\ S = \{I_j\}_{j=1}^{n-2} \cup \{A, B\} & epoch \geq \xi \end{cases} \tag{13}$$

where $epoch$ denotes the current training epoch and ξ is a predefined threshold which defines the epoch at which we start using the hard negative sampling strategy.

From the captioning perspective, the easy set helps the model to learn to produce generally good captions, while the hard samples push the model to focus on subtle details in the images and generate fine-detailed captions.

Putting everything together, the network is trained using a weighted loss function:

$$\mathcal{L}_{final} = (1-\gamma)\mathcal{L}_{pri} + \gamma\mathcal{L}_{aux} \tag{14}$$

where $0 \leq \gamma \leq 1$ determines the weight for the auxiliary loss function.

5. Experimental Results

We perform empirical experiments to evaluate the performance of the proposed method on the change captioning task.

5.1. Dataset and Setting

We use the CLEVR-Change [25] dataset to evaluate the performance of our approach. CLEVR-Change is a synthetic dataset that is generated using the CLEVR engine

Method	B4	C	M	R	S
Capt-Pix-Diff [25]	30.2	75.9	23.7	-	17.1
Capt-Rep-Diff [25]	33.5	87.9	26.7	-	19.0
Capt-At [25]	42.7	106.4	32.1	-	23.2
Capt-Dual-Att [25]	43.5	108.5	32.7	-	23.4
DUDA [25]	47.3	112.3	33.9	-	24.5
VAM [31]	50.3	114.9	37.0	69.7	30.5
Ours	**51.2**	**115.4**	**37.7**	**70.5**	**31.1**

Table 1: Performance of the proposed method on the entire CLEVR-Change dataset. Metrics indicated by "-" are not reported by the authors. Numbers are taken from respective papers. Our proposed method improves the performance of DUDA which uses the same base network. B4, C, M, and R, S are BLEU-4, CIDEr, METEOR, and ROUGE-L, Spice, respectively.

Method	B4	C	M	R	S
Changed Pairs Only					
DUDA	42.9	94.6	29.7	-	19.9
Ours	**49.9**	**101.3**	**34.3**	**65.4**	**27.9**
Distractor Pairs Only					
DUDA	59.8	110.8	45.2	-	29.1
Ours	**62.4**	**116.3**	**50.5**	**53.9**	**35.0**

Table 2: Performance of the proposed method evaluated only on the changed pairs (top) vs. the performance evaluated only on the distractor pairs (bottom) on the CLEVR-Change dataset. B4, C, M, and R, S are BLEU-4, CIDEr, METEOR, and ROUGE-L, Spice, respectively.

Method	B4	C	M	R
DDLA [13]	0.081	0.340	0.115	0.283
DUDA	0.081	0.325	0.118	0.291
Ours	0.081	0.345	0.125	0.299

Table 3: Performance of our method against DUDA and DDLA on the Spot-the-diff dataset. B4, C, M, and R are BLEU-4, CIDEr, METEOR, and ROUGE-L, respectively.

Method	B4	C	M	R	S
Ours (easy set only)	51.0	115.2	37.3	70.4	30.8
Ours (easy+hard set)	**51.2**	**115.4**	**37.7**	**70.5**	**31.1**

Table 4: Performance of the proposed method when only providing the auxiliary network with the easy set of candidates (first row) vs. the performance when using a dynamic strategy and switching to hard sample sets after a certain epochs. Other settings remain identical in both cases. Our method benefits from the dynamic strategy and the result has been improved. B4, C, M, and R, S are BLEU-4, CIDEr, METEOR, and ROUGE-L, Spice, respectively.

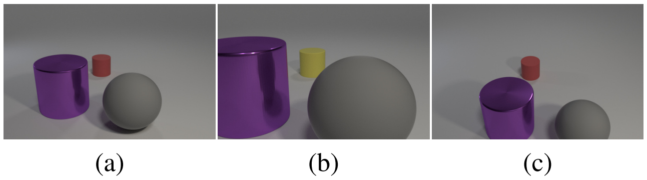


Figure 3: (Best viewed in color) Examples of two types of pairs in the dataset. (a) and (b) form a "changed pair" since there is a change at the object level (red → yellow). (a) and (c) form a "distractor" pair since there is only viewpoint change.

[14]. Due to its flexibility in generating different scenarios, CLEVR has become a standard tool to create diagnostic datasets for a variety of vision-language applications.

CLEVR-Change has 67660, 3976, and 7970 samples for training, validation, and test splits, respectively. The image pairs are categorized into two scenarios: distractor pairs, and changed pairs. Distractor pairs are those in which no object has been changed between two images. However, the camera view has changed from one image to another. A successful model should predict that there has been no change for distractor pairs. Changed pairs are those samples in which there is a change at the object level from one image to the other. Changes can be 1) changing an object's color; 2) changing an object's texture; 3) adding a new object to the scene; 4) removing one object from the image; and 5) moving an object to a new position. Each image pair in the dataset is accompanied by a sentence describing the change or expressing the absence of any changes depending on the scenario. Fig. 3 shows a sample of each of these two scenarios.

To compare with other methods, we also report the performance of our method on Spot-the-difference dataset [13]. This dataset contains 13,192 images of mostly parking lots. Each pair of images differ in a subtle change. There are two key differences between Spot-the-diff and CLEVR-change: 1) the camera in Spot-the-diff is fixed while in CLEVR-change the viewing changes from one image to the other in the pair, and 2) there are no distrators in Spot-the-diff, i.e. one can assume that there is always a change between two images. These differences make CLEVR-change a more challenging benchmark for the task of image change captioning.

We use PyTorch [26] for implementation and jointly train the primary and auxiliary networks end-to-end for 60 epochs using Adam optimizer [16]. The learning rate is set to $5e-3$. We report our experimental results in terms of BLEU@1, BLEU@2, BLEU@3, BLEU@4, CIDEr, SPICE, METEOR, Rouge-L [25].

5.2. Results

Results on CLEVR-Change: We present the results of our method on the entire CLEVER-Change dataset in Table 1. We compare our method against other state-of-the-art methods. Our approach outperforms DUDA [25] which uses the same base network. The performance of our method is also on par with VAM [31] which uses a different network architecture. Since the code of VAM is not released yet, we cannot build our approach based on VAM. Also, note that [31] has reported improved results using reinforcement learning as postprocessing. In order to keep the comparison fair, we report the results of the VAM version without this extra postprocessing in Table 1. Note that the focus of this work is not on proposing a new architecture for change captioning, but proposing a training scheme that can improve the performance of any given change captioning network including VAM.

Results on Changed Pairs: Table 2 (top rows) provides the results of evaluating our proposed method only on pairs of images that have a changed object. Our method outperforms the DUDA method which uses the same change captioning network as our primary network.

Results on Distractor Pairs: Finally we present our result on evaluating only on distractor pairs in Table 2 (bottom rows). These image pairs only have the camera angle/scene lighting change. Again we see a similar trend. Our method significantly outperforms DUDA.

Results on Spot-the-diff: We report the performance of our method on the Spot-the-diff dataset in Table 3. Again, our method outperforms other alternative approaches.

Qualitative Examples: We present some qualitative examples in Fig. 4. The proposed method generates captions that are semantically similar to the ground-truth captions. Note that in the last example, there is no change at the object level between the two input images. Instead, these two images only have a viewpoint change. The caption generated by our method correctly indicates that there is no change between these two images.

5.3. Ablation Study

We perform additional ablation studies to further analyze various aspects of the proposed approach. Specifically, we are interested to measure the effect of dynamic negative set selection on the overall performance. Also, we provide a break-down performance of our method on various types of change for the semantically changed pairs. All ablation studies are performed on the CLEVR-Change dataset.

Effect of Using Hard Negative Samples: To identify the effect of easy vs. hard candidate set, we perform two experiments and report the result in Table 4. For the first experiment, we train the networks using only the easy candidate sets, *i.e.* S only contains one image from (A, B). In the second experiment, we start our training by constructing S as an easy set. After certain epochs (30 in our case), we start to provide the hard set for the auxiliary network, namely, put both A and B among the n candidate images. Table 4 clearly demonstrates that the later strategy is superior to the former strategy which only uses easy candidate sets.

The result of this experiment make intuitive sense. Consider the case where the generated caption from the primary network ($\hat{C}$) along with image A is the input to the auxiliary network. When the easy set is solely used to train the auxiliary network, eventually the auxiliary network learns to distinguish the right image from the candidate set even if the generated caption $\hat{C}$ is not very accurate. This is because A and B are very similar to each other and only differ in one change. So if only easy candidate sets are provided to the auxiliary network, it eventually learns to ignore the input caption and picks the image in the candidate set that is most similar to the input image. This causes the auxiliary loss to become extremely low and does not provide much gradient flow for the primary network supervision.

To avoid this issue, it is essential to dynamically increase the task difficulty for the auxiliary task so that it does not converge prematurely or learn to ignore $\hat{C}$. Using this dynamic method, the auxiliary loss provides a much more reasonable gradient flow throughout the training process and improves the performance of the primary network as seen in Table 4.

Result per Change Category: We also provide the breakdown results of our method for different change categories in Table 5. The proposed method effectively improves the performance of the DUDA network by a large margin in almost every category.

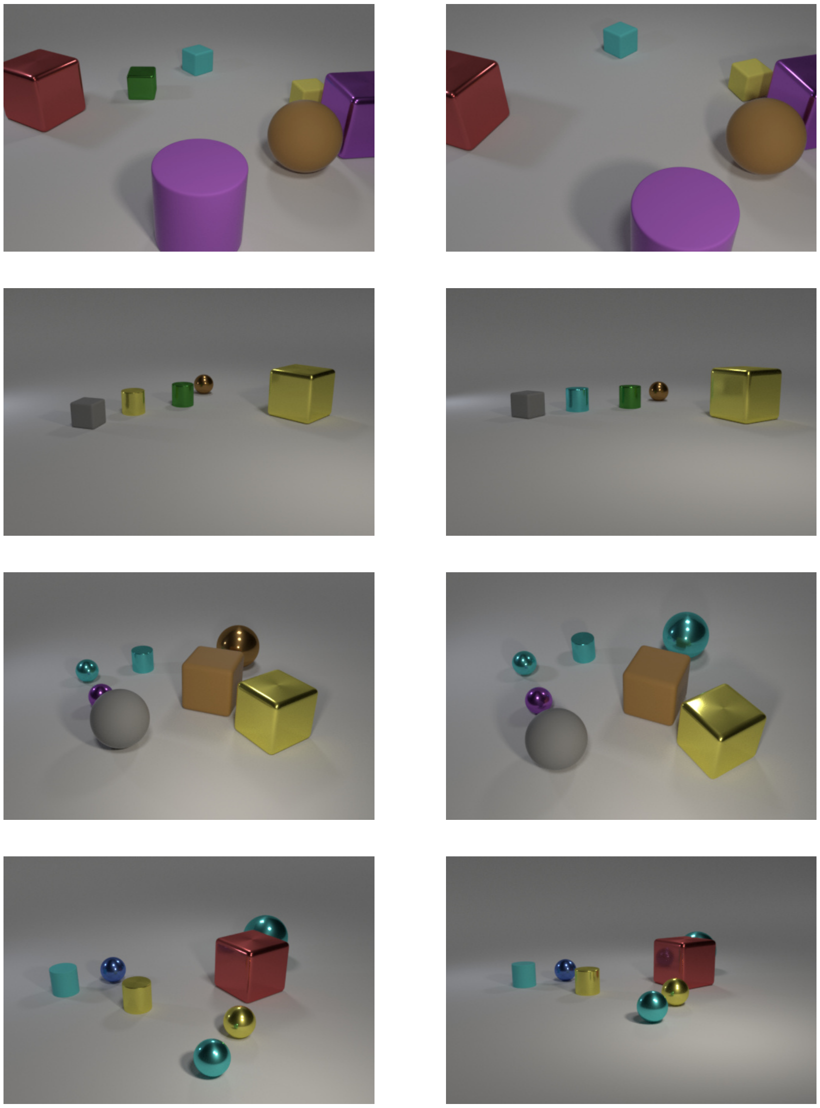
6. Conclusion

We have proposed a new training scheme for the task of image change captioning. Our proposed scheme uses the composed query image retrieval as an auxiliary task to improve the primary task of image change captioning. The two networks of these tasks are jointly trained in a sequential fashion. Our learning scheme enables the auxiliary task to provide an extra level of supervision for the primary task. This scheme along with a proposed candidate set selection for the image retrieval task proves to be effective for improving the performance of primary network. Our experimental results demonstrate the effectiveness of our proposed approach.

Acknowledgement: This work was supported in part by NSERC, and in part by the FoS research chair program and the GETS program at the University of Manitoba. We thank NVIDIA for donating some of the GPUs used in this work.

	CIDEr					METEOR					SPICE				
Method	C	T	A	D	M	C	T	A	D	M	C	T	A	D	M
Capt-Pix-Diff [25]	4.2	16.1	30.1	27.1	18.0	7.4	16.0	24.4	20.9	18.2	1.3	6.8	11.4	10.6	9.2
Capt-Rep-Diff [25]	44.5	21.9	50.1	49.7	26.5	19.2	18.2	25.7	23.5	18.9	8.2	8.8	12.1	12.0	9.6
Capt-At [25]	112.1	75.9	91.5	98.4	49.6	30.5	25.4	30.2	31.2	22.2	17.9	16.3	19.0	22.3	14.5
Capt-Dual-Att [25]	115.8	82.7	85.7	103.0	52.6	32.1	26.7	29.5	31.7	22.4	19.8	17.6	16.9	21.9	14.7
DUDA [25]	120.4	86.7	108.2	103.4	56.4	32.8	27.3	33.4	31.4	23.5	21.2	18.3	22.4	22.2	15.4
Ours	**120.8**	**89.9**	**119.8**	**123.4**	**62.1**	**36.1**	**30.4**	**37.8**	**36.7**	**27.0**	**29.7**	**27.4**	**31.4**	**30.8**	**23.5**

Table 5: Performance of the proposed method compared with other state-of-the-art approaches on each category of change. The changes categories are : Color Change (C), Texture Change (T), Adding an object (A), Deleting an Object (D), and Moving an Object (M).



Ours: *the tiny green metal block that is behind the big purple matte thing is no longer there*
GT: the small green metallic block that is behind the large red metallic object has disappeared

Ours: *the small yellow shiny cylinder that is behind the small cyan rubber thing turned cyan*
GT: the small yellow shiny cylinder that is to the left of the small brown shiny thing became cyan

Ours: *he large brown metallic sphere that is behind the big cyan matte thing turned cyan*
GT: the big brown metal sphere behind the yellow metallic thing turned cyan

Ours: *the scene remains the same*
GT: there is no change in the scene

Figure 4: (Best viewed in color) Qualitative examples of our method. The first three rows depict the cases where there is a change at the object level between the two input images, such as object removal, change in object texture, and change in object color, respectively. The last row shows a case in which there is no semantic change between two images.

References

[1] 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[2] Jyoti Aneja, Aditya Deshpande, and Alexander G Schwing. Convolutional image captioning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[3] Yanbei Chen and Loris Bazzani. Learning joint visual semantic matching embeddings for language-guided retrieval. In *European Conference on Computer Vision (ECCV)*, 2020. 2

[4] Marcella Cornia, Matteo Stefanini, Lorenzo Baraldi, and Rita Cucchiara. Meshed-memory transformer for image captioning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[5] Riccardo Del Chiaro, Bartłomiej Twardowski, Andrew Bagdanov, and Joost van de Weijer. Ratt: Recurrent attention to transient tasks for continual image captioning. *Advances in Neural Information Processing Systems*, 2020. 2

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009. 5

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*, 2019. 2

[8] Eric Dodds, Jack Culpepper, Simao Herdade, Yang Zhang, and Kofi Boakye. Modality-agnostic attention fusion for visual search with text feedback. *arXiv:2007.00145*, 2020. 2

[9] Xuguang Duan, Wenbing Huang, Chuang Gan, Jingdong Wang, Wenwu Zhu, and Junzhou Huang. Weakly supervised dense event captioning in videos. In *Advances in Neural Information Processing Systems*, 2018. 2

[10] Lionel Gueguen and Raffay Hamid. Large-scale damage detection using satellite imagery. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 1, 2

[11] Mehrdad Hosseinzadeh and Yang Wang. Composed query image retrieval using locally bounded features. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020. 2, 3

[12] Lun Huang, Wenmin Wang, Jie Chen, and Xiao-Yong Wei. Attention on attention for image captioning. In *IEEE International Conference on Computer Vision (ICCV)*, 2019. 2

[13] Harsh Jhamtani and Taylor Berg-Kirkpatrick. Learning to describe differences between pairs of similar images. *arXiv:1808.10584*, 2018. 1, 2, 6

[14] 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2, 6

[15] 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. 2

[16] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015. 5, 6

[17] Guang Li, Linchao Zhu, Ping Liu, and Yi Yang. Entangled transformer for image captioning. In *IEEE International Conference on Computer Vision (ICCV)*, 2019. 2

[18] Zhuowan Li, Quan Tran, Long Mai, Zhe Lin, and Alan L Yuille. Context-aware group captioning via self-attention and contrastive features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3440–3450, 2020. 2

[19] Shikun Liu, Andrew Davison, and Edward Johns. Self-supervised generalisation with meta auxiliary learning. In *Advances in Neural Information Processing Systems*, 2019. 3

[20] Zhunga Liu, Gang Li, Gregoire Mercier, You He, and Quan Pan. Change detection in heterogenous remote sensing images via homogeneous pixel transformation. *IEEE Transactions on Image Processing*, 27(4):1822–1834, 2017. 2

[21] Avital Oliver, Augustus Odena, Colin A Raffel, Ekin Dogus Cubuk, and Ian Goodfellow. Realistic evaluation of deep semi-supervised learning algorithms. In *Advances in neural information processing systems*, pages 3235–3246, 2018. 3

[22] Ariyo Oluwasanmi, Muhammad Umar Aftab, Eatedal Alabdulkreem, Bulbula Kumeda, Edward Y Baagyere, and Zhiquang Qin. Captionnet: Automatic end-to-end siamese difference captioning model with attention. *IEEE Access*, 7:106773–106783, 2019. 1, 2

[23] Ariyo Oluwasanmi, Enoch Frimpong, Muhammad Umar Aftab, Edward Y Baagyere, Zhiguang Qin, and Kifayat Ullah. Fully convolutional captionnet: Siamese difference captioning attention model. *IEEE Access*, 7:175929–175939, 2019. 1

[24] Yingwei Pan, Ting Yao, Yehao Li, and Tao Mei. X-linear attention networks for image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10971–10980, 2020. 2

[25] Dong Huk Park, Trevor Darrell, and Anna Rohrbach. Robust change captioning. In *IEEE International Conference on Computer Vision*, 2019. 1, 2, 3, 5, 6, 7, 8

[26] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, 2019. 6

[27] Julia Patriarche and Bradley Erickson. A review of the automated detection of change in serial imaging studies of the brain. *Journal of Digital Imaging*, 17(3):158–174, 2004. 1

[28] Ken Sakurada and Takayuki Okatani. Change detection from a street image pair using cnn features and superpixel segmentation. In *British Machine Vision Conference*, 2015. 1

[29] Meet Shah, Xinlei Chen, Marcus Rohrbach, and Devi Parikh. Cycle-consistency for robust visual question answering. In

IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019. 2

[30] Baifeng Shi, Judy Hoffman, Kate Saenko, Trevor Darrell, and Huijuan Xu. Auxiliary task reweighting for minimum-data learning. *Advances in Neural Information Processing Systems*, 33, 2020. 3

[31] Xiangxi Shi, Xu Yang, Jiuxiang Gu, Shafiq Joty, and Jianfei Cai. Finding it at another side: A viewpoint-adapted matching encoder for change captioning. 2020. 1, 2, 3, 6, 7

[32] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. In *Conference on Empirical Methods in Natural Language Processing and International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019. 2

[33] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in neural information processing systems*, pages 1195–1204, 2017. 3

[34] Jiaojiao Tian, Shiyong Cui, and Peter Reinartz. Building change detection based on satellite stereo imagery and digital surface models. *IEEE Transactions on Geoscience and Remote Sensing*, 52(1):406–417, 2013. 2

[35] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017. 2

[36] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: A neural image caption generator. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 2

[37] Nam Vo, Lu Jiang, Chen Sun, Kevin Murphy, Li-Jia Li, Li Fei-Fei, and James Hays. Composing text and image for image retrieval-an empirical odyssey. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2, 3, 4

[38] Junbo Wang, Wei Wang, Liang Wang, Zhiyong Wang, David Dagan Feng, and Tieniu Tan. Learning visual relationship and context-aware attention for image captioning. *Pattern Recognition*, 98:107075, 2020. 2

[39] 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 *International Conference on Machine Learning (ICML)*, 2015. 2

[40] Quanzeng You, Hailin Jin, Zhaowen Wang, Chen Fang, and Jiebo Luo. Image captioning with semantic attention. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2

[41] Massimo Zanetti and Lorenzo Bruzzone. A generalized statistical model for binary change detection in multispectral images. In *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 3378–3381, 2016. 2

[42] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *IEEE International Conference on Computer Vision (ICCV)*, 2017. 2, 3