IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 1 July 2023, accepted 7 August 2023, date of publication 10 August 2023, date of current version 7 September 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3303891

RESEARCH ARTICLE

Counterfactual Mix-Up for Visual Question Answering

JAE WON CHO[1], (Student Member, IEEE), DONG-JIN KIM[2], (Member, IEEE), YUNJAE JUNG[1], (Student Member, IEEE), AND IN SO KWEON[1], (Member, IEEE)

[1]Department of Electrical Engineering, Korea Advanced Institute of Science and Technology (KAIST), Daejeon 34141, South Korea
[2]Department of Data Science, Hanyang University, Seoul 04763, South Korea

Corresponding author: In So Kweon (iskweon77@kaist.ac.kr)

This work was supported in part by the Police-Laboratory 2.0 Program (www.kipot.or.kr) funded by the Ministry of Science and Information and Communication Technology (MSIT), South Korea, and Korean National Police Agency (KNPA), South Korea [Project Name: Artificial Intelligence (AI) System Development for a Image Processing Based on Multi-Band (Visible, Near-Infrared (NIR), Long-Wave Infrared (LWIR)) Fusion Sensing under Project 220122M0500]; in part by the Institute of ICT Planning and Evaluation (IITP) funded by the Korean Government through MSIT under Grant 2020-0-01373; in part by the AI Graduate School Program (Hanyang University); and in part by the National Research Foundation of Korea (NRF) funded by the Korean Government through MSIT under Grant RS-2023-00245661.

ABSTRACT Counterfactuals have been shown to be a powerful method in Visual Question Answering in the alleviation of Visual Question Answering's unimodal bias. However, existing counterfactual methods tend to generate samples that are not diverse or require auxiliary models to synthesize additional data. In this regard, we propose a more diverse and simple counterfactual sample synthesis method called Counterfactual Mix-Up (CoMiU), which generates counterfactual image features and questions through batch-wise swapping in local object- and word-level. This method efficiently facilitates the generation of more abundant and diverse counterfactual samples, which help improve the robustness of Visual Question Answering models. Moreover, with the creation of diverse counterfactual samples, we introduce two more robust and stable contrastive loss functions, namely Batch-Contrastive loss and Answer-Contrastive loss. We test our method on various challenging Visual Question Answering robustness testing setups to show the advantages of the proposed method compared with the current state-of-the-art methods.

INDEX TERMS Computer vision, counterfactuals, visual question answering, unimodal bias.

I. INTRODUCTION

Visual Question Answering (VQA) [1], a task that requires a model to correctly predict an answer given an image-question pair, has been actively studied for several years [2], [3], [4], [5], [6], [7]. However, VQA models commonly suffer from unimodal bias [8], [9], [10], where a model predicts answers by simply relying on language priors. These models are able to obtain the correct answer with only the given question and without actually "seeing" the visual information. However, a desirable visual system should give informative hints for the right answer based on the evidence of visual information present. In other words, removing the essential information of an image or question should lead to uncertainty or even the wrong answer [11].

To this end, VQA models have been using counterfactuals to combat the unimodal bias problem [11], [12], [13] as using counterfactuals enforces a model to answer through logical coherency or through more exposure to data. Within counterfactual methods, several methods have been studied where they try to balance the dataset with counterfactual samples. Although such methods have shown favorable performance on the VQA-CP dataset [9], such methods are either not diverse enough (creating a single counterfactual by masking by Chen et al. [12]) or not efficient (requiring manual human annotations by Selvaraju et al. [14] or a large amount of knowledge to pre-train an inpainting model by Gokhale et al. [15]). In response, we propose a simple and more efficient method of creating more diverse and abundant counterfactual samples that do not require any external knowledge. We name our counterfactual sample generation method as *Counterfactual Mix-Up* (CoMiU), where

The associate editor coordinating the review of this manuscript and approving it for publication was Szidonia Lefkovits.

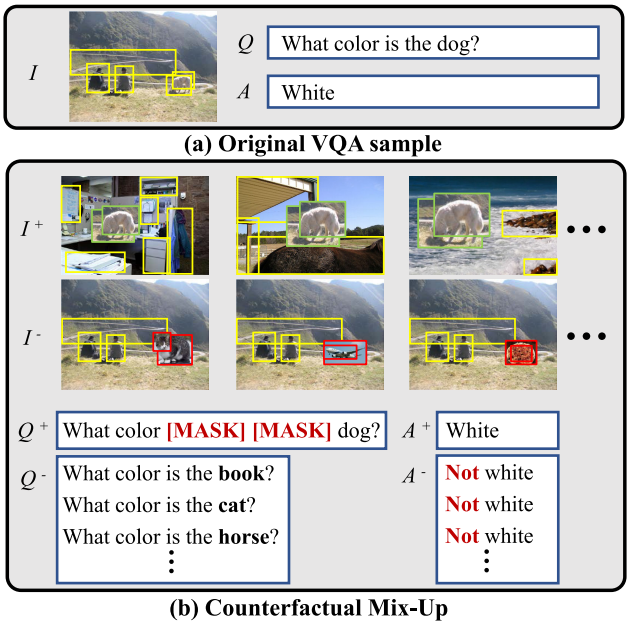

FIGURE 1. (a) Original sample. (b) Counterfactual samples that can be generated from CoMiU with various distractor images and questions. Note that the image is a representation as the object detector in VQA sometimes has overlapping images. I^+ is an image with the original object and one of the random backgrounds, I^- is an image with one of the random objects and the original background. Q^+ keeps question-type words and the salient word while the rest are masked. Q^- is the question of the salient word is switched with a different word. A^+ is an answer for (I^+, Q) or (I, Q^+), and A^- is an answer for (I^-, Q) or (I, Q^-). CoMiU is able to generate diverse and abundant counterfactual samples without any external knowledge.

counterfactual images are generated by *swapping local* object bounding boxes with random backgrounds, creating multiple and diverse counterfactual images at every training iteration. While selecting random object bounding boxes to swap is already helpful, we improve our counterfactual image mix-up by swapping the object bounding boxes that are *similar* based on GradCAM [16]. Counterfactual questions are generated by finding semantically similar salient *words* within its batch to swap with the original salient *words*, creating a new sentence every iteration that is seemingly similar, but in actuality a counterfactual to the image-question pair. Ultimately, these diverse counterfactual images we generate aid in unimodal bias by exposing the model to a vast number of counterfactual samples and making our model more robust by making the model focus on image to predict a correct answer.

In contrast to CSS [12], CoMiU is able to generate more abundant and diverse counterfactual samples, making the model exposed to more data, and improving its performance. For example, with the VQA-CP2 dataset [9] containing $121K$ images for the train set, CSS [12] creates $121K \times 2$ potential counterfactual samples, while CoMiU can potentially create $(121K)^2$ images. Fig. 1 illustrates the number of counterfactual images and questions that can be generated from a single image-question pair. On the other hand, MUTANT [15], using an external knowledge pre-trained inpainting model, generates a total of 679K samples. In short, our method is significantly efficient while being able to generate diverse counterfactuals without external knowledge.

Moreover, in order to best exploit our vastly generated diverse counterfactual samples, we utilize two contrastive based loss functions which help learn the feature representations of our counterfactual samples. In particular, we introduce *Batch-Contrastive loss* (BC) to learn the relationship between the original and the positive/negative counterfactual samples within the batch and *Answer-Contrastive loss* (AC) to optimize the distance between the answer and the counterfactual samples' representation. Through generating more abundant and diverse counterfactual samples, our contrastive losses can be more effectively utilized, which shows one of the significant benefits of our novel counterfactual sample generation method. We empirically show the effectiveness of our approach by showing state-of-the-art results on the VQA-CP2, VQA-CP1 [9], GQA-OOD [17] datasets, and favorable performances on other robustness testing VQA datasets such as VQA-CE [18] among the approaches without external networks or data [15].

Our contributions are summarized as follows: (1) We devise a novel method for generating counterfactuals, CoMiU, to improve the generalizability and robustness of VQA models. (2) We show the benefit of our diverse and abundant counterfactual samples with our two contrastive learning based loss functions. (3) We show the efficacy of our approach through extensive experiments on VQA datasets compared to recent state-of-the-art methods.

II. RELATED WORK

A. BIAS IN VQA

Even with much progress in the field of Visual Question Answering (VQA) [2], [6], [19], [20], VQA models have been known for their unimodal bias, more specifically language bias, and various VQA datasets that aid diagnose this problem have been proposed [9], [17], [18]. Several different techniques have been proposed to target this issue, such as ensemble based methods [13], [21], [22], [23], [24], balancing based methods with counterfactual examples [11], [12], [25], dataset shuffling methods [26], [27], [28], [29], [30], data augmentation methods [14], [15], [31], loss weight balancing [32], or even using a meta task like Visual Entailment [33]. Although data augmentation and balancing methods are powerful, such methods require either more data or extra steps that are difficult to procure or recreate. Chen et al. [12], where we gain inspiration from, only mask certain parts of an image or question, allowing the framework to be end-to-end without the need for additional data or models. However, this method shows limited diversity in the generated samples compared to data augmentation approaches, whereas our method automatically generates diverse *mixed* image and question pairs to reduce bias. Large scale vision and language transformer models [20], [34], [35], [36] also employ masking as a pre-training task, but they do not employ any form of mix-up in their pre-training. In our work, we do

TABLE 1. The list of notations used in our method along with their descriptions.

Notation	Description
$I = \{v_i\}_{i=1}^{N_v}$	Image. A set of N_v number of bounding boxes.
$Q = \{w_i\}_{i=1}^{N_q}$	Question. A sequence of N_q number of words
$A \subset \mathcal{A}$	A set of answers among the whole answer set $\mathcal{A}$
$I^+ = \{I^+, I^-\}$	Positive / Negative counterfactual image
$Q^+ = \{Q^+, Q^-\}$	Positive / Negative counterfactual question
A^+, A^-	Answer for Positive / Negative counterfactuals
O	Foreground objects of the image
B	Background region part of the image
$C(\cdot)$	The criterion to find foreground object
N	Number of data-points in a training batch
$N_{obj} \leq N_v$	# of objects to select for image mix-up
$g(\cdot)$	Pre-trained word embedding model
$s(\cdot, \cdot)$	Cosine similarity
$f(\cdot, \cdot)$	VQA feat generator + auxiliary projection function
z, z^+, z^-	Projected feat for original and counterfactuals

not directly compare to MUTANT [15] as this is out of scope and comparison to a newly generated dataset is unfair in testing debiasing learning techniques.

B. CONTRASTIVE LOSS

Motivated from the Noise Contrastive Estimation [37], contrastive learning [38], [39] has been recently adopted in various state-of-the-art self-supervised representation learning methods [40], [41]. Recently, contrastive learning has shown to be also effective for various tasks that require mutual information maximization other than self-supervised representation learning such as visual grounding (between image and caption) [42] and image-to-image translation (between source and target images) [43] with varying degrees of success. In addition, several existing VQA methods adopt the contrastive learning based loss function [42], [44] in order to maximize the mutual information (1) between visual-question joint feature and answer [15] or (2) between the original and the counterfactual representations without considering batch-wise representations. In contrast to the existing contrastive loss functions for the counterfactual VQA setting that show marginal improvements, by virtue of the more dense and diverse counterfactual samples generated via our method, our reformulated contrastive loss function shows more favorable performance.

III. PROPOSED METHOD

In this section, we give a simple explanation of Visual Question Answering (VQA) and explain in detail our methods of creating counterfactual samples in addition to our newly formulated losses. We also list all the notations used in our method along with their descriptions in Table 1.

A. VQA BASELINE

Given a pair of an image I and a question as a sequence of N_q number of words $Q = \{w_i\}_{i=1}^{N_q}$, the goal of the VQA task is to correctly predict a set of answers from the whole answer set $\mathcal{A}$, $A \subset \mathcal{A}$. We adopt one of the famous VQA state-of-the-art models as our base architecture, Bottom-Up Top-Down **(UpDn)** [2], where the object features are pre-extracted using a Faster-RCNN [45] network, the questions are embedded using a Glove Embedding [46], then both of these features are combined using a combination of convolutional and recurrent neural networks. Since the introduction of **UpDn** [2], it is commonplace in the VQA community to use the pre-extracted VQA object features. For the VQA-CP v2 dataset, among fusion based ensemble models that mitigate bias, we employ the Learned Mixin (**LMH**) debiasing method [22] in our work. In the training stage, the loss function in [22] is calculated based on the fusion of the ensemble of bias and plain model. During testing, only the plain model is used. With these in mind, we define our counterfactual sample generation for VQA under this formulation.

B. GENERATING COUNTERFACTUAL SAMPLES

Counterfactuals is defined as follows: "imagine something were to happen differently, how would it affect the outcome?" Given this definition, we devise a method of synthesizing counterfactual samples by changing the foreground object bounding boxes in an image or words in a question that may affect the answer prediction. As many data augmentation works [47] show that the diversity of the training samples improves the generalizability of the model on a test dataset, our goal is to efficiently generate *diverse and abundant* counterfactual samples. To this end, we propose to *swap* the *local* objects/words with another image/question; we call this method, Counterfactual Mix-Up (CoMiU). With the synthesized counterfactual image question pairs, the model is required to give an answer that is counterfactual to the input. To do so, we utilize the answer assigning method [12] to assign positive answer A^+ given positive counterfactual pairs (I^+, Q) or (I, Q^+) where A^+ is the set of top-N answers with the highest probabilities, and negative answer A^- given negative counterfactual pairs (I^-, Q) or (I, Q^-) where $A^- = \{a_i | a_i \in A, a_i \notin A^+\}$. Counterfactual questions and images are generated separately with a proportion hyper-parameter. With these preliminaries, we explain our method of creating counterfactual samples, $I^\pm = \{I^+, I^-\}$ and $Q^\pm = \{Q^+, Q^-\}$ for each training iteration, in detail as follows.

Counterfactual Image Mix-Up. Since the introduction of UpDn [2], pre-extracted Faster-RCNN [45] features have been used as visual features in VQA. In this regime, each image I is represented as a set of N_v number of pre-extracted object bounding boxes, *i.e.*, $I = \{v_i\}_{i=1}^{N_v}$. In our work, we first define I as a union of two different sets of foreground object bounding boxes O and background object bounding boxes B, *i.e.*, $I = O \subset B$. Although we empirically find that randomly selecting the foreground bounding boxes is already helpful for our method, we define *salient* bounding boxes as the foreground in order to further improve the effectiveness of our method. In particular, we use the modified version of Grad-CAM [16] to define a set of salient foreground bounding boxes O and label the rest as background B.

To generate counterfactual images, I^+ and I^-, we propose to swap salient object bounding boxes with another random image, generating more abundant and diverse counterfactuals every training iteration instead of simply masking either O or B. In particular, given a target image I_i and another random image that we consider a "distractor" image I_j from a training batch, we define the positive counterfactual to include the foreground object O_i from I_i and random background B_j from I_j. Combining the two, we create a new image: $I_i^{j+} = O_i \cup B_j$. Then, to create the negative counterfactual, similarly, we do the opposite, *i.e.*, $I_i^{j-} = O_j \cup B_i$. Note that even though I_i^{j+} and I_j^{i-} are identical, both are utilized for training with different labels (A_i for I_i^{j+} and A_j for I_j^{i-}).

In practice, considering all possible N^2 counterfactuals from a training batch with the size of N is computationally inefficient. Therefore, we assign one random distractor image for each of N target images in a training batch by shuffling the batch. As a result, we can abuse the notation of I_i^{j+} and I_i^{j-} as I_i^+ and I_i^- from here for simplicity. This means that as the batches are randomized at every iteration, a new set of random counterfactual positive and negative samples are generated, signifying that it is not a single positive and negative counterfactual per sample. In other words, there is only one I_i^+ per batch, but I_i^+ and I_i^- are different at every iteration. This ensures computational efficiency while still being able to generate diverse samples. Due to the large number of data points in the dataset and the randomness of our method, much like [28], we assume the chances of ambiguities and distractors to be sufficiently low.

To determine the number of object bounding boxes that we define as foreground object bounding boxes and background, we utilize the criterion $C(\cdot)$ defined by [12], which measures the relative magnitude of the Grad-CAM over the object bounding boxes in an image. The number of foreground object bounding boxes is defined as the number of object bounding boxes that have a Grad-CAM magnitude that surpasses a pre-defined threshold η. Unlike simple masking [12] or inpainting [15], as we utilize a *pair* of images to generate counterfactual samples, we need to reconcile the mismatch in the number of foreground object bounding boxes between the two images. As a result, we propose to define the number of object bounding boxes to swap $N_{obj} \leq N_v$ to be the floor of the batch-wise mean of the number of object bounding boxes that have large enough $C(\cdot)$:

$$N_{obj} = \left\lfloor \frac{1}{N} \sum_{i=1}^{N} \sum_{v \in I_i} \mathbb{1}\big[C(v) > \eta\big] \right\rfloor, \tag{1}$$

where N is the batch size, $\mathbb{1}[\cdot]$ is the indicator function, $v \in I_i$ is the individual object bounding box in an image, and η is the threshold hyper-parameter. We conjecture that by creating counterfactual images by swapping with randomly selected images, we are able to create that many more samples and give increased exposure to the model [48]. In addition to this, instead of simple masking, changing up the foreground object bounding boxes and backgrounds may lead to a more natural image in comparison to masked values.

Counterfactual Question Mix-Up. Similar to images, we determine the local contribution of each word in the question through the use of a modified Grad-CAM [16]. To create Q^+, we keep the question-type words (*e.g.*, "how many") and the salient word, and we mask the rest with the token "[MASK]." Then for Q^-, instead of simply masking the salient word, we propose to find different words and switch them with the salient word in the original question to generate a new question. Unlike random mix-up for images, randomly shuffling words does not guarantee the grammatical correctness of a generated sentence. Therefore, we propose to generate counterfactual questions by switching salient words with *semantically similar* words. In particular, we project all the salient words within the training batch into a Glove [46] pre-trained embedding space $g(\cdot)$. Then, to generate a counterfactual question Q^- from a question Q, we switch the current salient word $w_i \in Q$ with the most similar (but not same) salient word w_j^* from the entire set of salient words in a training batch, $\{w_j\}_{j=1}^N$:

$$w_i \leftarrow w_{j^*}, \text{ where } j^* = \underset{\substack{1 \leq j \leq N, \\ j \neq i, w_j \neq w_i}}{\operatorname{argmax}} s\Big(g(w_i), g(w_j)\Big), \tag{2}$$

where $s(\cdot, \cdot)$ is the cosine similarity. To avoid the phenomena where the same word is chosen, we also include additional criteria to the argmax setting to remove all instances where the words are the same ($w_j = w_i$). Note that, instead of finding a single optimum pair for a data point from the whole training data, as we find w_{j^*} every iteration from the training *batch*, a randomness factor is kept, improving the diversity of the generated counterfactual questions. As it is our motivation, we also empirically find that improved diversity leads to better performance. If the batch is too small, semantically similar replacements may not always be found, in this case, it can be seen as random word switching. Note that similar to the image counterfactuals, at every iteration, a different counterfactual Q^+ and Q^- are generated.

C. CONTRASTIVE LEARNING OBJECTIVES

To verify the effectiveness of our abundant counterfactual samples, we introduce two different contrastive learning based loss functions: (1) Batch-Contrastive loss and (2) Answer-Contrastive loss.

1) BATCH-CONTRASTIVE LOSS

With the causal triplets (I, I^+, I^-) and (Q, Q^+, Q^-) obtained from CoMiU at every training iteration, our goal is to effectively train our VQA model with these samples by training the relationship between the original sample and the positive/negative counterfactual samples, similar to the self-supervised representation learning studies [40], [41]. However, unlike self-supervised methods that require a large-scale dataset for pre-training tasks, we only utilize the

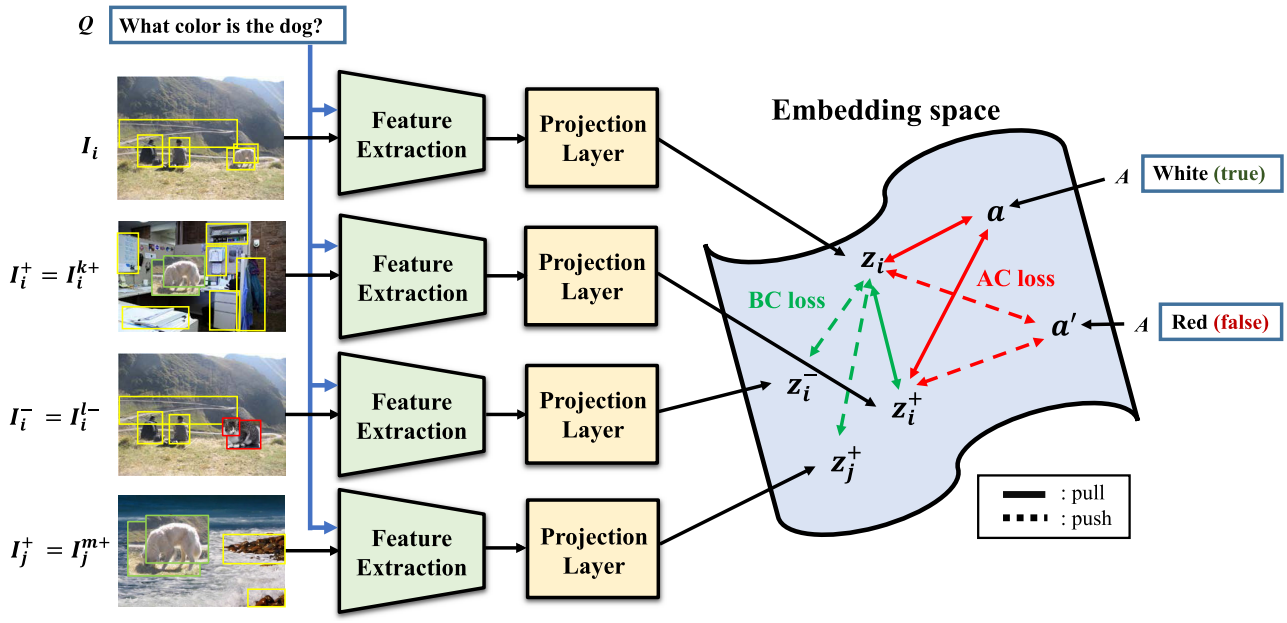

FIGURE 2. Illustration of Batch-Contrastive (BC) loss and Answer-Contrastive (AC) loss. Given a question (Q) with the original image (I_i), we show an example of our positive and negative counterfactual samples that can be generated. Given the features computed in the batch, the BC loss maximizes the similarity between the positive counterfactual sample and the original sample ($s(z_i, z_i^+)$) while minimizing the similarity between the original sample and the other samples including the negative counterfactual samples ($s(z_i, z_i^-)$) or the counterfactual samples from the other images ($s(z_i, z_j^+)$). In addition, the AC loss maximizes the similarity between the joint features and the ground truth answer ($s(z_i, g(a))$ and $s(z_i^+, g(a))$) while minimizing the similarity between the joint features and the other answers ($s(z_i, g(a'))$ and $s(z_i^+, g(a'))$).

samples that are generated at training time. Here, we introduce the contrastive learning based loss function, which we call Batch-Contrastive (BC) loss. In particular, for the case of counterfactual images as shown in Fig. 2, the I_i, I_i^+, and I_i^- paired with Q_i are fed into the multi-modal feature extractor of the VQA model, the last layer before the answer classifier. Then, passing through an auxiliary projection function [40] $f(\cdot, \cdot)$, the joint embeddings are projected as $z_i = f(I_i, Q_i)$, $z_i^+ = f(I_i^+, Q_i)$, and $z_i^- = f(I_i^-, Q_i)$, respectively which are used to compute the loss function as follows:

$$\mathcal{L}_i^{BC} = -\log\left(\frac{e^{s(z_i, z_i^+)}}{\sum_{j=1}^{N} e^{s(z_i, z_j^+)} + e^{s(z_i, z_j^-)}}\right), \tag{3}$$

where we use cosine similarity for the similarity scoring function $s(\cdot, \cdot)$. In other words, with the feature from the original image z_i as an anchor and the positive and negative counterfactuals (z_i^+ and z_i^-), we apply this BC loss overall $2 \times N$ counterfactual features in the batch of size N in order that the similarity between the positive counterfactual and the anchor features $s(z_i, z_i^+)$ is maximized while the similarity between the negative and the anchor $s(z_i, z_i^-)$ is minimized. Note that we also decrease the similarity between the anchor and all counterfactual features within the batch ($j \neq i$), further improving the stability and boosting the performance. As mentioned in Sec. III-B, note that we create only one pair of positive/negative counterfactual features for each of the N data points in the training batch at each training iteration. Therefore, there are $2 \times N - 1$ negative samples when we put all (both positive and negative) counterfactual samples $z_j^{\pm}$ (excluding z_i^+) in the batch as negative samples. Note that the same is applied for the counterfactual question where Q, Q^+, and Q^- are paired with I. Also, note that at each iteration, a different pair of positive/negative counterfactual features are generated and used to measure the contrastive loss.

We also find that under intuitive assumptions, the mutual information between the representations of the original sample and positive counterfactual sample $\mathcal{I}(z; z^+)$ is bounded by the contrastive learning formulation as follows:

$$\mathcal{I}(z; z^+) \geq \log(2N) - \mathbb{E}_{(z_i, z_i^+)}\left[\mathcal{L}_i^{BC}\right], \tag{4}$$

which becomes tighter as N becomes larger [38]. The proof of this inequality can be found in the supplementary material. Therefore, minimizing our BC loss function can maximize the lower bound of the mutual information, enabling the model to learn the relationship between the counterfactual and the original sample to predict the right answer from a more causal aspect.

If we only consider the same image ($N = 1$), the denominator of Eq. (3) becomes small, meaning that it is not large enough or the negative samples are not diverse enough. Then, the lower bound of the mutual information in Eq. (4) becomes loose; thus, degrades the performance [41], [49]. In contrast, by virtue of generating more abundant and diverse counterfactual samples, we can make Eq. (4) more effective, which shows one of the key significance of our novel counterfactual sample generation.

2) ANSWER-CONTRASTIVE LOSS

In addition, to aid the model distinguish the answers given a visual-question pair, we also introduce an additional contrastive learning based loss function, Answer-Contrastive (AC) loss, that operates in the space of answer embeddings by computing the similarity between the joint embeddings z and the embedding of all possible answers in the ground truth answer set, *i.e.*, $g(a), a \in A_i \subset \mathcal{A}$ given a joint feature z_i and one pair of positive/negative counterfactual samples z_i^+ and z_i^-:

$$\mathcal{L}_i^{AC} = -\log\left(\frac{\sum_{a\in A_i} e^{s\left(z_i,\ g(a)\right)} + e^{s\left(z_i^+,\ g(a)\right)}}{\sum_{z'\in\{z_i,z_i^+,z_i^-\}}\sum_{a'\in\mathcal{A}} e^{s\left(z',\ g(a')\right)}}\right). \quad (5)$$

As answers are words, we project the answers using Glove embedding and try to match the joint embeddings on the same embedding space. This loss function aims to maximize the similarity between the projections of the multi-modal joint features of our model (either with the original image or positive counterfactual) and the projected Glove embedding vector of the ground truth answer for the given question-image pair, $s(z, g(a))$ or $s(z^+, g(a))$. Note that the similarity metric is between the projection of multi-modal features and ground truth answers, not between the predicted and ground truth answers.

By removing z and z^- in our AC loss function, the formulation becomes similar to that of MUTANT [15]. Unlike MUTANT, by considering the relation between z and $z^{\pm}$s with respect to the distance with the answer embedding, our AC loss creates larger hypothesis space for the loss function (three times larger); our loss function ultimately becomes more stable and effective. We empirically show the effectiveness of the loss function. Our final formulation is the sum of the typical supervised VQA loss from [22] and $\lambda_1\mathcal{L}^{BC}+\lambda_2\mathcal{L}^{AC}$, with the weights of λ_1 and λ_2 being 5 to 1 respectively.

IV. EXPERIMENTS

In this section, we show our experimental setting and the findings through quantitative and qualitative results.

A. EXPERIMENTAL SETTING

1) DATASET

As we consider the bias problem within VQA, we evaluate our models on the VQA-CP2 and VQA-CP1 datasets [9]. Following the release of the newer out-of-distribution evaluation dataset, we also report our results on the GQA-OOD test set. We also report our results on VQA-CE [18], which is a new evaluation protocol to determine how much a model exploits shortcuts within the VQA-v2 dataset.

2) EVALUATION METRIC

As the standard VQA evaluation metric, we evaluate all of our VQA models on the standard VQA evaluation metric as in [1].

TABLE 2. Comparison of our method on the VQA-CP2 test set. Best results are styled in this manner while second best results are styled in this manner. We mainly compare our results within the same architecture and method type. ∗ shows models that we run on our machines using publicly available code. We show that our model outperforms other counterfactual techniques while significantly improving the "Num" category.

Dataset	VQA-CP2 test			
Model	All	Yes/No	Num	Other
Vanilla baseline architectures				
SAN [51]	24.96	38.35	11.14	21.74
GVQA [9]	31.30	57.99	13.68	22.14
S-MRL [21]	38.46	42.85	12.81	43.20
UpDn* [2]	39.94	42.46	11.93	45.09
Methods that include human annotations				
HINT [14]	46.73	67.27	10.61	45.88
SCR [31]	49.45	72.36	10.93	48.02
Methods that use ensembles				
AReg [10]	41.17	65.49	15.48	35.48
RUBi [21]	44.23	67.05	17.48	39.61
LMH [22]	52.45	69.81	44.46	45.54
GGE [23]	57.32	87.04	27.75	49.59
GenB [24]	59.15	88.03	40.05	49.25
Methods based on counterfactuals:				
CVL [11]	42.12	45.72	12.45	48.34
CF-VQA [13]	53.55	**91.15**	13.03	44.97
CSS [12]	**58.95**	84.37	**49.42**	**48.21**
CSS* [12]	58.61	83.33	49.39	**48.18**
CoMiU + BC + AC (Ours)	**59.99**	**87.71**	**54.23**	47.04
Methods that change the distribution of the dataset:				
Unshuffling [27]	42.39	47.72	14.43	47.24
RandImg [26]	55.37	83.89	41.60	44.20
SSL [28]	57.59	86.53	29.87	50.03
D-VQA [29]	61.91	88.93	52.32	50.39
KDDAug [30]	60.24	86.13	55.08	48.08
Methods that use external data or network when training:				
MUTANT [15]	61.72	88.90	49.68	50.78
SAR [33]	62.51	76.40	59.40	56.09

3) BASELINE ARCHITECTURE

We choose a popular state-of-the-art VQA baseline architecture, UpDn [2],[1] which is a commonly utilized architecture as a testing platform for debiasing. We also base our debiasing model on the ensemble based debiasing baseline LMH [22].

B. IMPLEMENTATION DETAILS

We implement our model using PyTorch, and as mentioned, we use the Bottom-Up Top-Down (UpDn) [2] model as our baseline architecture with a pre-trained Faster-RCNN [45] visual features. All questions are embedded using a 300D space Glove embedding [46]. We also use the default question-type annotations in the VQA-CP2 dataset. All of our models are trained on a single Nvidia Titan Xp GPU for 30 epochs with a batch size of 256 using the Adamax optimizer, which is a variant of Adam optimizer [50], on default settings. All 30 epochs require about 12 hours to complete.

[1]we use the publicly available re-implementation from https://github.com/hengyuan-hu/bottom-up-attention-vqa

TABLE 3. Results on the VQA-CP1 test set compared with the state-of-the-art methods. All models are based on the UpDn architecture except for GVQA. Results in **Bold** show the best results in each column.

Dataset	VQA-CP1 test			
Model	All	Yes/No	Num	Other
GVQA [9]	39.23	64.72	11.87	24.86
UpDn [2]	39.94	42.46	11.93	45.09
AReg [10]	41.17	65.49	15.48	35.48
RUBi [21]	50.90	80.83	13.84	36.02
LMH [22]	55.27	76.47	26.66	45.68
CSS [12]	60.95	85.60	**40.57**	44.62
CoMiU + BC + AC (Ours)	**63.13**	**90.44**	38.25	**45.97**

TABLE 4. We show the experimental results on the GQA-OOD dataset. We compare to available debiasing methods and show that our method outperforms all debiasing methods by a large margin.

Method	GQA-OOD test			
	All	Tail	Head	Avg
UpDn [2]	46.87	42.13	49.16	45.65
RUBi [21]	45.85	43.37	47.37	45.37
LMH [22]	43.96	40.73	45.93	43.33
CSS [12]	44.24	41.20	46.11	43.66
Ours	**48.96**	**45.91**	**50.84**	**48.38**

Each image has up to 36 bounding box objects with a feature size of 2048 and each question length is limited to 14 words per question. The η for the image mix-up threshold is set to 0.65. All of the answers are projected using a 300 dimension space Glove embedding [46]. To project the multi-modal joint features, we use a two-layer multi-layer perceptron (MLP) and resize the features into a 300 dimension space for the Answer-Contrastive and Batch-Contrastive loss.

For our method of Question and Image mix-up as mentioned in the method section, following CSS [12], we apply image and question mix-up independently. The proportion hyper-parameter is set at 0.1, which translates to 10% of the time the model uses Question mix-up and 90% of the time using Image mix-up. In addition, as masking can be seen as another form of mix-up, we found empirically found that masking in addition to mix-up was best in aiding performance gains, so we also include masking as a form of mix-up and run our experiments accordingly. The AC loss and BC loss are added using a loss weight, with the weights of AC to BC loss being 5 to 1.

C. RESULTS ON VQA-CP2

In order to understand how our method in relation to other debiasing methods, we show our results in Table 2. In the first four rows, we first list the vanilla base architectures (SAN [51], GVQA [9], S-MRL [21]), and the architecture that the rest of the debiasing methods including ours is based on, UpDn [2]. Then we compare our methods to other debiasing methods such as HINT [14], SCR [31], AReg [10], RUBi [21], LMH [22], GGE [23], GenB [24], CVL [11], CF-VQA [13], CSS [12], Unshuffling [26], SSL [28], D-VQA [29], MUTANT [15], and SAR [33]. Among these, we compare our method to other methods that either include external knowledge, datasets, during or before pre-training, or use a combination of techniques such as ensemble and dataset shuffling for fairness.

We show our Counterfactual Mix-Up (CoMiU) combined with our losses, AC loss, and BC loss, in comparison to other state-of-the-art VQA-CP2 test set results. We list CSS [12]* as the baseline that we run on our machine for fair comparison. We find that CoMiU + BC + AC (Ours) achieves 59.99% overall accuracy which is 7.54% higher than that of LMH, where our model is based (relative 14.38% improvement to that of LMH). In addition, our final method surpasses other methods significantly in the "Num" category with "Yes/No" coming in at a close second while also being competitive in the "Other" category. Interestingly, our final method shows especially significant performance improvement from our baseline, LMH, for the "Yes/No" category (relative 25.64% improvement). Also, our final method with our contrastive losses scores significantly higher in the "Num" category, beating all existing methods by a large margin of 4.81% accuracy compared to the next best method of CSS (a relative 9.70% improvement), and is even higher than MUTANT, which requires large-scale data to pre-train the data generator.

TABLE 5. VQA-CE evaluation compared with the state-of-the-art methods. Ours shows the best performance in counterexamples (Counter). Also, note that our method surpasses the other powerful counterfactual method CSS by a large margin in all metrics.

Dataset	VQA-CE		
Model	Overall	Counter	Easy
UpDn [2]	63.52	33.91	76.69
RUBi [21]	61.88	32.25	75.03
LMH [22]	61.15	34.26	73.13
RandImg [26]	63.34	34.41	76.21
CSS [12]	53.55	34.36	62.08
CoMiU + BC + AC (Ours)	57.84	34.62	68.18

D. RESULTS ON VQA-CP1

We also show our results on the VQA-CP1 test set against other available state-of-the-art approaches. Note that not all methods have the VQA-CP1 scores available, so we only list those that are available. Table 3 shows that Ours outperforms all existing state-of-the-art methods by a noticeable margin of 7.86% overall accuracy (relative 14.22% improvement) compared to our baseline (LMH) and 1.86% overall accuracy (relative 3.03% improvement), compared to the second best method, CSS+CL. For "Yes/No," our method is 13.97% accuracy higher (relative 18.27% improvement) than our baseline, LMH. Ours also scores among the highest for both the "Num" and "Other" categories with "Num" being second best and "Other" being the best.

TABLE 6. Ablation study of our method on Image (I) only. We include SSL as it swaps questions and images whole. We include CoMiU with and without BC loss and AC loss to show the individual effects.

Model	All	Yes/No	Num	Other
UpDn Baseline	39.94	42.46	11.93	45.09
V-CSS (region masking) [12]	58.23	80.53	52.48	48.13
SSL (shuffling) [28]	57.59	86.53	29.87	50.03
CoMiU (I Only)	58.72	85.17	51.45	46.86
CoMiU + BC + AC (I Only)	59.14	88.14	49.06	46.71

TABLE 7. Ablation study for our method on Question (Q) only. Again, we include SSL as it swaps questions and images whole. We include CoMiU with different shuffling methods with and without our BC + AC losses to show the individual effects.

Model	All	Yes/No	Num	Other
UpDn Baseline	39.94	42.46	11.93	45.09
Q-CSS (word masking) [12]	56.66	80.82	45.83	46.98
SSL (shuffling) [28]	57.59	86.53	29.87	50.03
CoMiU (Q Only)	57.24	82.68	49.35	46.07
CoMiU + BC + AC (Q Only)	58.48	84.47	52.25	46.57

E. RESULTS ON GQA-OOD

Recently, a new VQA out-of-distribution dataset that tests the robustness of VQA models has been introduced recently called the GQA-OOD [17] where the training data is manually balanced and the out of distribution results are tested through a manually biased test set. Due to space limitations of the main paper, we include the results of the GQA-OOD test set evaluation in this section. We compare our method **OURS** to recent, available state-of-the-art debiasing methods that are based on the UpDn [2] architecture such as RUBi [21], LMH [22], or CSS [12]. Unlike our setting on the VQA-CP2 and CP1, we do not use the base LMH [22] loss in our network, instead, we apply CoMiU + AC + BC loss on the base UpDn architecture with no debiasing losses. As seen from Table 4, **Ours** shows a significant improvement from other debiasing methods. Even when compared to the baseline architecture UpDn where all the methods are based, Ours outperforms it by 2.09% overall. In the tail category, Ours outperforms UpDn by 3.78%, which is the category that directly tests the OOD (Out-Of-Distribution) performance. Through these results, we show that our method is robust not simply on the VQA-CP sets but also on the newly formed GQA-OOD data.

F. RESULTS ON VQA-CE

Recently a new evaluation protocol has been introduced on the VQA-v2 dataset to measure how much a VQA model is dependent on shortcuts called the VQA-CounterExamples (VQA-CE) [18]. The evaluation protocol is split into Overall, Counter, and Easy. The Overall score simply lists the total score for the VQA-v2 validation set. Easy is a subset of samples where the shortcuts within the image/question pair give the correct answers and Counter is a counterexample where using the shortcuts leads to incorrect answers. We evaluate our model on VQA-CE as shown in Table 5. Although the overall score for VQA-CE is dependent on the VQA-v2 performance as presented in the evaluation metric [18], we show that our method shows the best performance on Counterexamples, which is the main point of interest in this dataset. Also note that our method surpasses CSS in all the metrics by a noticeable margin, which is our baseline method.

TABLE 8. Ablation study for our contrastive losses. Using BC and AC losses together rather than using individual losses shows noticeable improvements in performance. Also, applying the BC and AC losses on the CSS baseline without CoMiU gives negligible performance improvement.

Model	All	Yes/No	Num	Other
CoMiU	59.06	86.68	48.50	47.49
+ (Answer Projection loss) [15]	58.71	84.53	52.41	46.90
+ **BC loss**	58.80	86.60	47.91	47.22
+ **AC loss**	58.80	86.19	49.96	46.87
CoMiU + BC + AC (Ours)	59.99	87.71	54.23	47.04
CSS* [12]	58.61	83.33	49.39	48.18
CSS* [12] + **BC** + **AC**	58.62	84.87	49.71	47.30

TABLE 9. Evaluation of our method with different debiasing backbones as an add-on module. Our method shows consistent overall accuracy improvements for all the backbone architectures.

Model	All	Yes/No	Num	Other
UpDn [2]	39.74	**42.27**	11.93	46.05
+ CoMiU + BC + AC (Ours)	**39.94**	41.34	**12.80**	**46.43**
RUBi [21]	45.23	64.85	11.83	**44.11**
+ CoMiU + BC + AC(Ours)	**46.40**	**69.31**	**13.45**	43.43
LMH [22]	52.45	69.81	44.46	45.54
+ CoMiU + BC + AC (Ours)	**59.99**	**87.71**	**54.23**	**47.04**

G. ABLATION STUDY RESULTS

To further understand the effectiveness of our proposed methods, we perform ablation studies on the components of our methods compared with existing methods on the VQA-CP2 dataset. As our method comprises three parts, the Counterfactual Mix-Up (CoMiU), Batch-Contrastive loss (BC), and Answer-Contrastive loss (AC), we denote them separately as **CoMiU**, **BC**, and **AC** in our ablation study tables. In addition, as CoMiU comprises the Image mix-up and the Question mix-up, we show the individual effects of each component in Table 6 and Table 7 respectively.

1) INDIVIDUAL MODALITY RESULTS

Table 6 and Table 7 show that even without BC and AC losses, CoMiU already improves the performance of the models while adding BC and AC losses further boosts performance. In particular, in Table 6, our image mix-up shows the best overall accuracy compared to the simple region making [12] or pair shuffling [28]. In addition, in Table 7, our question mix-up shows favorable overall accuracy compared to simple word masking [12] and shows comparable

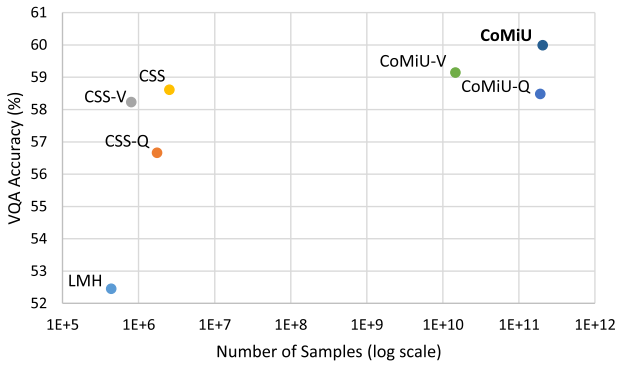

FIGURE 3. VQA performance vs diversity of the counterfactual samples during training shown on VQA-CP2. Compared to the baseline methods, LMH [22] and CSS [12], Our method (CoMiU), shown in bold, generates counterfactual samples with the largest diversity, which leads to the best VQA performance.

performance to that of pair shuffling [28]. We find that the Question mix-up improves the Num score while the Image mix-up improves the Yes/No score. We believe that each component forces the model to reduce bias on the specified modality and ultimately aids in the performance of the respective categories. Moreover, with our contrastive loss functions, our question mix-up performs significantly better than the competing methods.

2) CONTRASTIVE LOSS RESULTS

To understand the effects of BC and AC losses, we test CoMiU with different types of contrastive losses and show the results in Table 8. Note that CoMiU itself without the losses already gives a large performance gain, and using both of the losses together rather than using the individual contrastive losses shows noticeable improvements in the performance of CoMiU. We conclude that our BC and AC losses have complementary characteristics where the BC loss learns the feature level similarity while the AC loss learns the answer level similarity. Note that our AC loss without considering the original and negative counterfactual samples, which is similar to that of MUTANT [15] Answer Projection loss, which performs poorly when compared to BC or AC losses. In order to further understand the effects of our BC and AC losses, we also show how our losses perform when paired with CSS [12]. Adding our BC and AC losses gives CSS a negligible performance improvement, showing that the BC and AC losses are helpful when aided by our CoMiU which generates diverse and abundant negative and positive samples.

3) BACKBONE ABLATION

We test the general applicability of our method on other debiasing baselines (UpDn [2], RUBi [21], and LMH [22]) and show our results in Table 9. On the simple UpDn model, our CoMiU with BC and AC losses improves the performance in overall in addition to the "Num," and "Other" categories. On RUBi [21], CoMiU and the losses improve the performance by a fair margin, with both the "Yes/No" and "Num" category outperforming the baseline. As for LMH [22], which is the base debiasing loss for our model, CoMiu with the two losses improves the performance by a significant margin and improves the scores in every category by significant margins. Overall, our method is able to consistently improve the performance of various VQA backbone models.

TABLE 10. Evaluation of our method on the LXMERT [20] backbone architecture among reported counterfactual based baselines. Note that the availability of counterfactual based baselines is few so we list those available. We also include the ensemble based debiasing loss LMH which is the base debiasing baseline of our method and CSS. We find that among the counterfactual based baselines that do not include external data or network when training, we find our method outperforms previous baselines significantly.

Model	Overall	Yes/No	Num	Other
LXMERT [20]	44.14	43.12	17.07	51.66
+ LMH [22]	59.66	73.41	57.72	52.99
+ CSS [12]	63.63	**84.70**	62.12	53.00
+ CoMiU + BC + AC (Ours)	**68.82**	74.03	**73.51**	**64.80**

4) ARCHITECTURE ABLATION

We further test our method on the popular transformer based backbone the LXMERT [20] architecture. We show our results in Table 10. As many previous baselines lack performance on this backbone, we include the counterfactual based methods that do not use external data or external networks when training. We also include the LMH ensemble based debiasing technique on which our method is based. We find that on the LXMERT baseline, our method performs significantly better than CSS in the Overall category. In addition, our method improves upon the LMH baseline significantly in all categories.

5) DIVERSITY AND PERFORMANCE

In order to demonstrate the effect of the diverse counterfactual samples on the resulting VQA performance, we show a plot between the VQA accuracy and the total number of training samples for several baselines in Fig. 3. Compared to the baseline method, LMH [22], CSS [12] generates two counterfactual samples (one positive and one negative) which limits the diversity gap. In contrast, our method, CoMiU utilizes random distractor samples for every training iteration, which leads to significant diversity improvements compared to the baseline methods. Note that our method with the largest sample diversity shows the best VQA performance, showing the correlation between the counterfactual sample diversity and the model performance.

H. QUALITATIVE RESULTS

We show the qualitative results of our method with the attention masks and the top-1 answer from the baseline and Ours in Fig. 4. The examples in the first and the second columns show that although both models answer correctly,

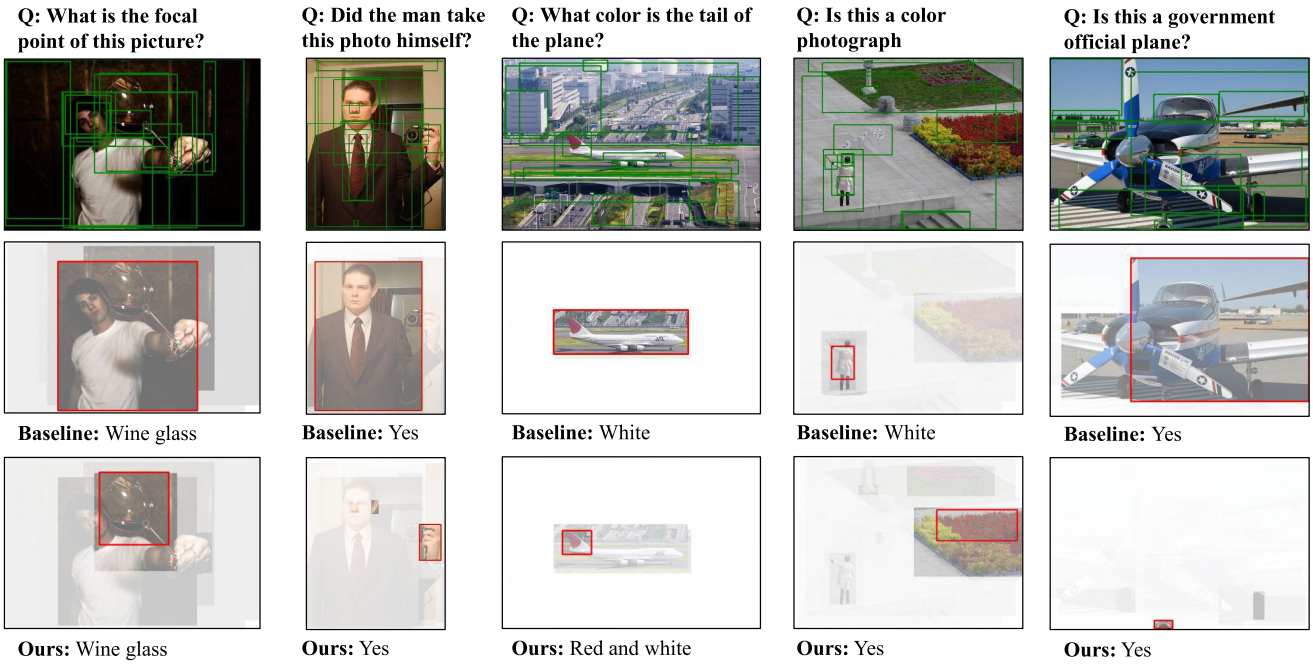

FIGURE 4. Qualitative results of our method. The top row shows the input image with object bounding boxes, the second row shows the baseline (LMH) with the baseline's answers. The last row shows our result. The translucent image regions show where the model attends to and the red boxes show the highest attention. Our model is able to attend more directly and accurately to the objects that cause the answers. In the last row, we show a negative case where the baseline attends relatively correctly, but our model is unable to attend correctly. We find that for questions with knowledge based reasoning questions, just like the baseline, our method also struggles.

our model is able to attend more distinctly to the region that causes the answers. The example in the third column shows that our model really focuses on the region specified by the question (''the tail of the plane''), and answers accordingly. The fourth column shows our model does not simply looks at the question priors, rather it shows that our model understands the semantic meaning of the question properly and answers instead of answering with color because there is a word ''color'' in the question. The last column shows a failure case where our model is unable to attend correctly or answer correctly to the question. We conjecture that for knowledge based reasoning questions, like many other VQA models including the baseline, our model also suffers. These qualitative examples verify that our model is able to attend in a more focused and accurate manner to the objects that cause the answers.

V. CONCLUSION

We propose a new efficient method of generating diverse and abundant counterfactuals called Counterfactual Mix-Up and two loss functions, Batch-Contrastive and Answer-Contrastive losses, to improve the generalizability and robustness of VQA models. Instead of simple masking, we show that using a dynamic way to create counterfactual samples leads to better VQA performance. We also show a way to leverage all of the newly generated samples in a contrastive manner during training. With these proposed methods, we show state-of-the-art results on the VQA-CP2, VQA-CP1, and GQA-OOD test sets without any additional data and show through qualitative results the impact of our method. In addition, we also show state-of-the-art results on the VQA-CE evaluation metric. In this work, we strive to automatically generate a large number of samples, and we do not specifically focus in depth on the quality of generated images. We believe that a potential scope for this type of work is generating high fidelity images automatically for counterfactual reasoning. To go further, with the current developments of large foundational models, we hope our work can inspire further works to automatically generate not just images but image-question pairs that can be more helpful for, but not limited to, the VQA task.

ACKNOWLEDGMENT

(Jae Won Cho and Dong-Jin Kim contributed equally to this work.)

REFERENCES

[1] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, and D. Parikh, ''VQA: Visual question answering,'' in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 2425–2433.
[2] P. Anderson, X. He, C. Buehler, D. Teney, M. Johnson, S. Gould, and L. Zhang, ''Bottom-up and top-down attention for image captioning and visual question answering,'' in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6077–6086.
[3] H. Ben-younes, R. Cadene, M. Cord, and N. Thome, ''MUTAN: Multimodal Tucker fusion for visual question answering,'' in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2631–2639.
[4] Y. Goyal, T. Khot, D. Summers-Stay, D. Batra, and D. Parikh, ''Making the V in VQA matter: Elevating the role of image understanding in visual question answering,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6325–6334.

[5] D. Gurari, Q. Li, A. J. Stangl, A. Guo, C. Lin, K. Grauman, J. Luo, and J. P. Bigham, "VizWiz grand challenge: Answering visual questions from blind people," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3608–3617.

[6] J.-H. Kim, J. Jun, and B.-T. Zhang, "Bilinear attention networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2018, pp. 1–12.

[7] Z. Yu, J. Yu, Y. Cui, D. Tao, and Q. Tian, "Deep modular co-attention networks for visual question answering," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 6274–6283.

[8] A. Agrawal, D. Batra, and D. Parikh, "Analyzing the behavior of visual question answering models," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2016, pp. 1955–1960.

[9] A. Agrawal, D. Batra, D. Parikh, and A. Kembhavi, "Don't just assume; look and answer: Overcoming priors for visual question answering," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4971–4980.

[10] S. Ramakrishnan, A. Agrawal, and S. Lee, "Overcoming language priors in visual question answering with adversarial regularization," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2018, pp. 1548–1558.

[11] E. Abbasnejad, D. Teney, A. Parvaneh, J. Shi, and A. van den Hengel, "Counterfactual vision and language learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10041–10051.

[12] L. Chen, X. Yan, J. Xiao, H. Zhang, S. Pu, and Y. Zhuang, "Counterfactual samples synthesizing for robust visual question answering," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10797–10806.

[13] Y. Niu, K. Tang, H. Zhang, Z. Lu, X.-S. Hua, and J.-R. Wen, "Counterfactual VQA: A cause-effect look at language bias," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 12695–12705.

[14] R. R. Selvaraju, S. Lee, Y. Shen, H. Jin, S. Ghosh, L. Heck, D. Batra, and D. Parikh, "Taking a HINT: Leveraging explanations to make vision and language models more grounded," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 2591–2600.

[15] T. Gokhale, P. Banerjee, C. Baral, and Y. Yang, "MUTANT: A training paradigm for out-of-distribution generalization in visual question answering," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2020, pp. 878–892.

[16] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 618–626.

[17] C. Kervadec, G. Antipov, M. Baccouche, and C. Wolf, "Roses are red, violets are blue…But should VQA expect them to?" in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 2775–2784.

[18] C. Dancette, R. Cadène, D. Teney, and M. Cord, "Beyond question-based biases: Assessing multimodal shortcut learning in visual question answering," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 1554–1563.

[19] Y. Kant, A. Moudgil, D. Batra, D. Parikh, and H. Agrawal, "Contrast and classify: Training robust VQA models," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 1584–1593.

[20] H. Tan and M. Bansal, "LXMERT: Learning cross-modality encoder representations from transformers," in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process.*, 2019, pp. 5100–5111, doi: 10.18653/V1/D19-1514.

[21] R. Cadene, C. Dancette, H. Ben-younes, M. Cord, and D. Parikh, "RUBi: Reducing unimodal biases in visual question answering," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2019, pp. 841–852.

[22] C. Clark, M. Yatskar, and L. Zettlemoyer, "Don't take the easy way out: Ensemble based methods for avoiding known dataset biases," in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)*, 2019, pp. 4069–4082.

[23] X. Han, S. Wang, C. Su, Q. Huang, and Q. Tian, "Greedy gradient ensemble for robust visual question answering," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 1564–1573.

[24] J. W. Cho, D.-J. Kim, H. Ryu, and I. S. Kweon, "Generative bias for robust visual question answering," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 1–6.

[25] V. Agarwal, R. Shetty, and M. Fritz, "Towards causal VQA: Revealing and reducing spurious correlations by invariant and covariant semantic editing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9687–9695.

[26] D. Teney, K. Kafle, R. Shrestha, E. Abbasnejad, C. Kanan, and A. van den Hengel, "On the value of out-of-distribution testing: An example of Goodhart's law," 2020, *arXiv:2005.09241*.

[27] D. Teney, E. Abbasnejad, and A. van den Hengel, "Unshuffling data for improved generalization," 2020, *arXiv:2002.11894*.

[28] X. Zhu, Z. Mao, C. Liu, P. Zhang, B. Wang, and Y. Zhang, "Overcoming language priors with self-supervised learning for visual question answering," in *Proc. 29th Int. Joint Conf. Artif. Intell.*, Jul. 2020, pp. 1083–1089.

[29] Z. Wen, G. Xu, M. Tan, Q. Wu, and Q. Wu, "Debiased visual question answering from feature and sample perspectives," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 3784–3796.

[30] L. Chen, Y. Zheng, and J. Xiao, "Rethinking data augmentation for robust visual question answering," 2022, *arXiv:2207.08739*.

[31] J. Wu and R. J. Mooney, "Self-critical reasoning for robust visual question answering," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2019, pp. 8604–8614.

[32] Y. Guo, L. Nie, Z. Cheng, Q. Tian, and M. Zhang, "Loss re-scaling VQA: Revisiting the language prior problem from a class-imbalance view," *IEEE Trans. Image Process.*, vol. 31, pp. 227–238, 2022.

[33] Q. Si, Z. Lin, M. Y. Zheng, P. Fu, and W. Wang, "Check it again: Progressive visual question answering via visual entailment," in *Proc. 59th Annu. Meeting Assoc. Comput. Linguistics 11th Int. Joint Conf. Natural Lang. Process.*, 2021, pp. 4101–4110.

[34] W. Su, X. Zhu, Y. Cao, B. Li, L. Lu, F. Wei, and J. Dai, "VL-BERT: Pre-training of generic visual-linguistic representations," 2019, *arXiv:1908.08530*.

[35] J. Lu, D. Batra, D. Parikh, and S. Lee, "ViLBERT: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks," 2019, *arXiv:1908.02265*.

[36] Y.-C. Chen, L. Li, L. Yu, A. E. Kholy, F. Ahmed, Z. Gan, Y. Cheng, and J. Liu, "Uniter: Universal image-text representation learning," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 1–13.

[37] M. Gutmann and A. Hyvärinen, "Noise-contrastive estimation: A new estimation principle for unnormalized statistical models," in *Proc. Int. Conf. Artif. Intell. Statist. (AISTATS)*, 2010, pp. 297–304.

[38] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," 2018, *arXiv:1807.03748*.

[39] R. D. Hjelm, A. Fedorov, S. Lavoie-Marchildon, K. Grewal, P. Bachman, A. Trischler, and Y. Bengio, "Learning deep representations by mutual information estimation and maximization," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2019.

[40] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2020, pp. 1597–1607.

[41] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9726–9735.

[42] T. Gupta, A. Vahdat, G. Chechik, X. Yang, J. Kautz, and D. Hoiem, "Contrastive learning for weakly supervised phrase grounding," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 752–768.

[43] T. Park, A. A. Efros, R. Zhang, and J.-Y. Zhu, "Contrastive learning for unpaired image-to-image translation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 752–768.

[44] Z. Liang, W. Jiang, H. Hu, and J. Zhu, "Learning to contrast the counterfactual samples for robust visual question answering," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2020, pp. 3285–3292.

[45] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2015, pp. 91–99.

[46] J. Pennington, R. Socher, and C. Manning, "GloVe: Global vectors for word representation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1532–1543.

[47] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *J. Big Data*, vol. 6, no. 1, pp. 1–48, Dec. 2019.

[48] S. Yun, D. Han, S. Chun, S. J. Oh, Y. Yoo, and J. Choe, "CutMix: Regularization strategy to train strong classifiers with localizable features," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6022–6031.

[49] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin, "Unsupervised feature learning via non-parametric instance discrimination," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3733–3742.

[50] J. Ba and D. Kingma, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2015.

[51] Z. Yang, X. He, J. Gao, L. Deng, and A. Smola, "Stacked attention networks for image question answering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 21–29.

JAE WON CHO (Student Member, IEEE) received the B.S. degree in electrical engineering from the Georgia Institute of Technology, Atlanta, GA, USA, in 2018. He is currently pursuing the Ph.D. degree in electrical engineering with the Korea Advanced Institute of Science and Technology (KAIST) under the supervision of Professor In So Kweon. He was awarded a bronze prize from Samsung Humantech paper awards. His current research interests include deep learning topics, such as active learning and high-level computer vision application, such as vision and language.

DONG-JIN KIM (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in electrical engineering from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 2015, 2017, and 2021, respectively. He was a Postdoctoral Researcher in EECS with UC Berkeley, in 2022. He is currently an Assistant Professor with Hanyang University. He was a Research Intern with the Visual Computing Group, Microsoft Research Asia (MSRA). He was awarded a silver prize from Samsung Humantech paper awards and Qualcomm Innovation awards. His research interest includes data issues in computer vision especially in high-level computer vision problems.

YUNJAE JUNG (Student Member, IEEE) received the B.S. degree in electrical engineering from Sogang University, Seoul, South Korea, in 2018, and the M.S. degree in electrical engineering from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 2019, where he is currently pursuing the Ph.D. degree in electrical engineering. He was awarded a honorable mention from Samsung Humantech paper awards. His research interests include high-level computer vision, such as video scene understanding and vision and language.

IN SO KWEON (Member, IEEE) received the B.S. and M.S. degrees in mechanical design and production engineering from Seoul National University, South Korea, in 1981 and 1983, respectively, and the Ph.D. degree in robotics from the Robotics Institute, Carnegie Mellon University, USA, in 1990. He was with the Toshiba Research and Development Center, Japan, and he is currently a KEPCO Chair Professor with the Department of Electrical Engineering, since 1992. He served as the Program Co-Chair for ACCV 07' and ICCV 19', and the General Chair for ACCV 12'. He is on the honorary board of *IJCV*. He was a member of "Team KAIST," which won the first place in DARPA Robotics Challenge Finals 2015. He is a member of the KROS.