

SwapMix: Diagnosing and Regularizing the Over-Reliance on Visual Context in Visual Question Answering

Vipul Gupta¹ Zhuowan Li² Adam Kortylewski² Chenyu Zhang² Yingwei Li² Alan Yuille²
¹ Pennsylvania State University ² Johns Hopkins University
 vkg5164@psu.edu {zli1110, akorty11, czhan129, yingwei.li, ayuille1}@jhu.edu

Abstract

While Visual Question Answering (VQA) has progressed rapidly, previous works raise concerns about robustness of current VQA models. In this work, we study the robustness of VQA models from a novel perspective: visual context. We suggest that the models over-rely on the visual context, i.e., irrelevant objects in the image, to make predictions. To diagnose the models' reliance on visual context and measure their robustness, we propose a simple yet effective perturbation technique, SwapMix. SwapMix perturbs the visual context by swapping features of irrelevant context objects with features from other objects in the dataset. Using SwapMix we are able to change answers to more than 45% of the questions for a representative VQA model. Additionally, we train the models with perfect sight and find that the context over-reliance highly depends on the quality of visual representations. In addition to diagnosing, SwapMix can also be applied as a data augmentation strategy during training in order to regularize the context over-reliance. By swapping the context object features, the model reliance on context can be suppressed effectively. Two representative VQA models are studied using SwapMix: a co-attention model MCAN and a large-scale pretrained model LXMERT. Our experiments on the popular GQA dataset show the effectiveness of SwapMix for both diagnosing model robustness, and regularizing the over-reliance on visual context. The code for our method is available at <https://github.com/vipulgupta1011/swapmix>

1. Introduction

Visual Question Answering (VQA) is a challenging task that requires a model to answer open-ended questions based on images. In recent years, VQA performance is greatly boosted by different techniques including intra- and inter-modality attentions [7, 49], large scale multi-modal pretraining [26, 27, 42], etc. However, previous works study the robustness of VQA models and show that the models may

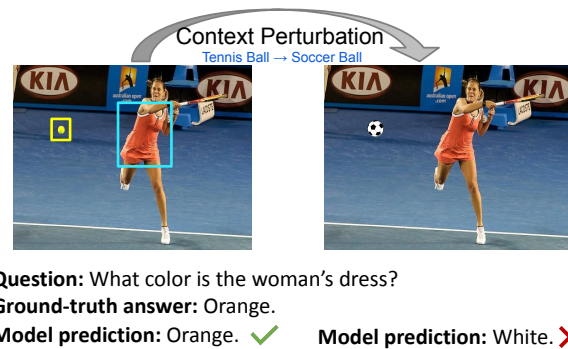


Figure 1. VQA models over-rely on visual context. By swapping features of irrelevant context objects, we can perturb the model prediction. Here the tennis ball (in yellow box) is an irrelevant context object for the question. Changing feature of the tennis ball to feature of soccer ball results in change in model prediction.

exploit language prior [2, 33, 34], statistical bias [1, 16] or dataset shortcuts [20, 21] to answer questions.

While previous works studied VQA robustness from the perspective of language context, in this work, we study the robustness of VQA models from a different view: visual context. The visual context refers to the background in the image or the irrelevant objects that are not needed during the reasoning process to answer the question. For example, in Figure 1, the tennis ball is irrelevant for the question "What color is the women's dress", so we say it is a context object. Ideally, a model with real perception and reasoning ability should be robust to the irrelevant context. However, in our work, we find that VQA models are vulnerable to context changes, which suggests the models' over-reliance on the irrelevant context in the image.

To study the role of visual context, we propose a simple perturbation strategy named *SwapMix*, which perturbs the visual context by swapping features of context object with features from another object in the dataset. We first identify the visual features corresponding to irrelevant objects in the image, then randomly swap them with feature vectors of another similar object from the dataset. For example, in Figure 1, the tennis ball is a context object for the given question,

so we swap tennis ball feature vector with a feature vector of soccer ball. The swapping confuses the model to misrecognize the color of the dress. In the swapping process, we carefully control the swapped objects to ensure that the new object is compatible to the scene (*e.g.*, we don't want to change the ball into a car).

Surprisingly, by perturbing the irrelevant context, more than 45% of the correct answers get changed. This reveals that VQA models highly rely on the context in the image, thus are vulnerable to context perturbations. The model may utilize shortcut correlations in the visual context to make predictions. We diagnose two representative VQA models: MCAN [49] as representative for attention-based models, and LXMERT [42] as representative for large-scale pretrained models. Our experiments show that LXMERT is much more robust to context perturbations, which indicates that large-scale pretraining may increase model robustness.

We further find that the context over-reliance highly depends on the quality of visual representations: a perfect sighted model relies much less on context. We achieve this by replacing the visual representations¹ with the ground-truth object and attribute encoding, which can be viewed as gold visual representation that provides the model the perfect sight. By studying this perfectly sighted model, we can exclude the influence of imperfect visual perception, thus purely focus on the reliance on relevant objects in the reasoning process. Our results shows that by providing VQA models with the perfect visual encoding, the answer changes are greatly reduced from 45.0% to 16.4% (for MCAN model). This suggests that models trained with perfect visual representations are more robust and that the context over-reliance largely comes from the imperfection of visual perception features.

In addition to diagnosing context over-reliance, SwapMix can also be used as a data augmentation technique during training. In training, we randomly swap a part of the context features with other object features from the dataset. This forces the model to focus more on relevant objects in the image and less on irrelevant context. Our empirical results show that by applying *SwapMix* in training, the model robustness improves by more than 40% and effective accuracy improves by more than 5% on GQA dataset [19].

Our main contributions in this paper are three-fold. First, we are the first to study VQA robustness from the perspective of visual context. With our simple context perturbation strategy named *SwapMix*, we benchmark robustness of two representative VQA models and find their over-reliance on visual context. Second, we find that a perfect sighted model relies much less on visual context. We provide models with perfect visual encodings and observe the improvement in model robustness. Third, we define 2 metrics, context re-

¹Majority of VQA models use object features extracted by pretrained object detectors as visual representation.

liance and effective accuracy and shows improvement by using *SwapMix* as data augmentation technique.

2. Related Works

Visual Question Answering. The most common approach for VQA is to first extract visual features using convolution neural networks and question features using LSTM [4], then fuse them together to make answer predictions [53]. Multiple works have shown the effectiveness of attention in VQA [6, 10, 14, 23, 48, 50, 51]. BAN [22] proposes bilinear attention that utilizes vision and language information. MCAN [49] is a co-attention model which uses self-attention and guided-attention units to model the intra-modal and inter-modal interactions between visual and question input. OSCAR [27] uses object tags in images as anchors to improve alignment between modalities. LXMERT [42] is a large-scale Transformer [44] model that consists of three encoders: an object relationship encoder, a language encoder, and a cross-modality encoder. In concurrent work, [9] proposes feature swapping for domain adaptation from synthetic to real data.

Biases and Robustness in VQA. Despite the prosperity in the development of VQA, multiple previous works show bias in VQA models. [1] points out the generalization incapacity of VQA models. [18] shows bias reliance of VQA models. [31] discover and enumerate explicitly biases learned by the model. Many work show that the models exploit language prior [2, 33, 34], statistical bias [1, 16] or dataset shortcuts [20, 21] to answer questions. There are many approaches to mitigate the bias in models. [2] introduces a method that reorganizes the VQA v2 dataset. Some works use question-only model: [37] introduces training as an adversarial game between the base model and a question-only adversary, while [8] adds a question-only branch to do joint training with the base model, and omits it at test time. CSS [11] generates counterfactual samples during training, which improves the visual-explainable ability. [40, 47] leverage the important visual information by humans to focus on selected regions during training. [12] designed a two-stage model, the first stage trains only on biases, and the second stage focuses on the other patterns of the dataset. In addition to decreasing modal biases, there are lot of work on measuring biases more accurately and efficiently. MUTANT [15] and GQA-OOD [20] use out-of-distribution (OOD) generalization. Early work like [30] provided a soft measure score based on a lexical set. [5] measures the performance of the models based on both the baseline questions and the CLOSURE test, indicating that the gap between these two measurements is the behavior of generalization. [13] measures bias in VQA by finding counter-examples from validation set with their proposed rules and use the mined counter-examples to evaluate model. Differ-

ent from the above previous works, our work is the first to study the reliance on visual context of VQA models by generating new examples.

Context in computer vision. Contextual information is important for computer vision. For object recognition, early work by [43] introduced a context-based model using place categorization to simplify object recognition, [35] studied how context influences object recognition, and recent work by [52] modified a global context model to enhance performance. Moreover, [45] demonstrate that object detection models rely too much on contextual information when objects are occluded, and resolve this using a compositional generative model [24, 25] that separates the object and context in the representation. For scene graphs, [38, 46] introduce a hierarchical context model to generate a scene graph, and [29] augment the node features of scene graphs with contextual information. For segmentation, [17] presents multi-scale contextual representation with context modules, which leverage the global image representations to estimate local affinity of sub-regions, and [28] introduces a switchable context network to improve the performance of semantic segmentation of RGB-D images. In the field of VQA, [41] add a visual context based attention that takes into account the previously attended visual content.

3. Method

VQA models are not robust to minor perturbations. In this section, we provide a simple perturbation technique that measures the reliance of VQA models on visual context, i.e. irrelevant objects image. We swap features corresponding to irrelevant objects in the image with other objects from the dataset. In an ideal scenario, changing the context objects in the image should not affect the model’s prediction, while in our experiments, we found that VQA models rely heavily on the context and are not robust to small perturbations.

We name our method, SwapMix, which performs perturbations on visual context to diagnose the robustness of the model. SwapMix can also be used as a data augmentation technique during training to improve the robustness and effective accuracy of the model. We first define what visual context is, then introduce VQA models with perfect sight which leads to interesting diagnosing findings, next describe how we perform SwapMix, and finally talk about how to apply SwapMix as a training strategy.

3.1. Definition of Visual Context

Here we clarify the definition of visual context and provide formulation for the problem.

$$f = Model(V, Q)$$

Here, V represents the visual representation and Q represents the question input. A widely-used visual representation is the object-based features [3] extracted by pretrained

object detector Faster RCNN [39]. In this case, $V \in \mathbb{R}^{n \times d}$ is a set of object features, where n is the number of objects in the image and d is the dimension of the feature vector for each object.

Among the n objects in the image, there are some irrelevant objects that are not needed in the reasoning process of question answering. For a fully robust model, changing the context, C should not change model’s prediction as shown in Figure 1. We refer to those irrelevant objects as visual context and denote visual context by C . $C \in \mathbb{R}^{m \times d}$ is a subset of V . It contains feature vectors corresponding to m irrelevant objects. Each row of the context C , denoted as c_i , is a feature vector corresponding to an irrelevant object.

The context objects are identified using the question reasoning steps. For example, in order to answer the question “What color is the statue in front of the trees”, we need to first find the tree, then find the statue in front of the tree and finally query its color. The GQA dataset [19] provides the ground-truth reasoning steps for each question, as well as the selected objects after each step. We use those reasoning steps to filter out all the relevant and irrelevant objects for the question. Then Intersection-over-Union (IoU) ratio is used to match the predicted objects with the ground-truth ones.

3.2. VQA Model with Perfect Sight

We conjecture that the model robustness is related to the quality of visual perception. The majority of the current VQA models use the object features described above as visual input to the model. The features are extracted by a pretrained off-the-shelf object detector which is not updated in VQA training. These pre-extracted features may contain a large amount of noise and miss out on important information that is required to answer the question. In this case, the model may be forced to learn unreasonable data correlations from irrelevant context to predict the answers correctly, which reduces the robustness of the model.

Therefore, to study the influence of visual perception imperfection, we train a model with perfect sight and compare its behavior with model trained with commonly used detected features. The models with perfect sight are trained using the scene graph annotations in GQA dataset. We replace the object features with the encoding of ground-truth object annotations. More specifically, for each object, we encode its annotated class label and attributes into one-hot encodings, which are then encoded using GloVe [36] embeddings and finally converted to inner dimension d with a FC layer. The object bounding box coordinates are also converted to same dimension using a FC layer. The final representation of an object i is the average of three parts: $c_i = Avg(o_i, a_i, b_i)$. $o_i, a_i, b_i \in \mathbb{R}^{1 \times d}$ are encodings for object class label, attributes and bounding box coordinates respectively.

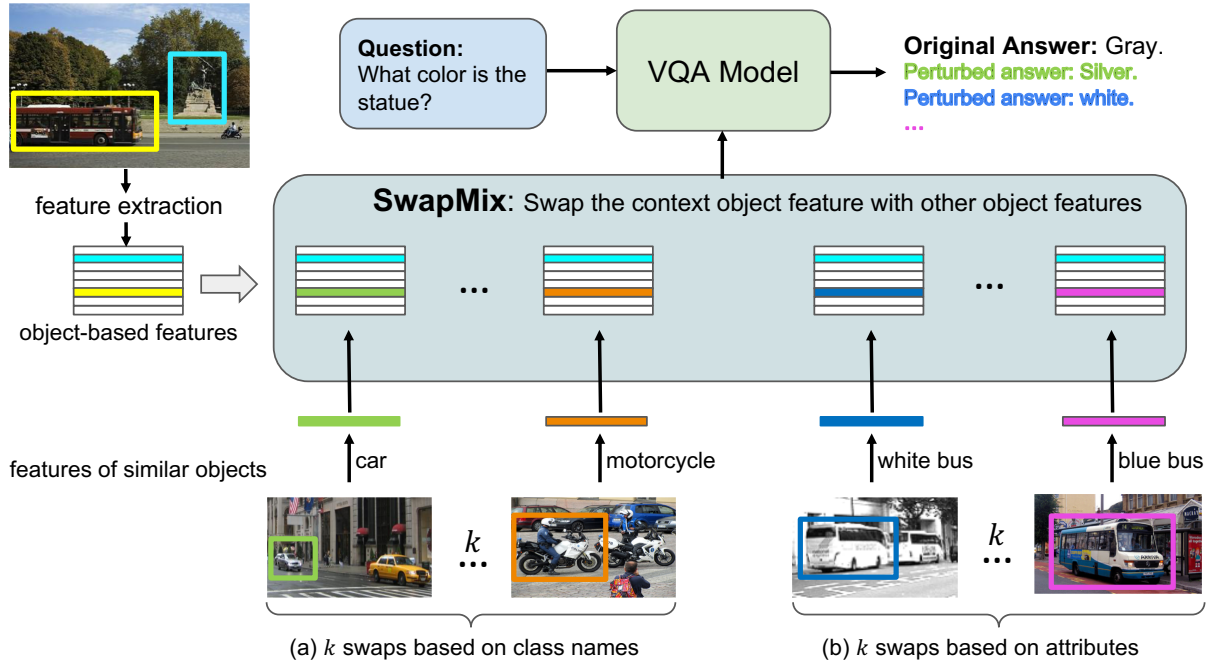


Figure 2. Overview of our method. Given an image and a question, we first find context object (e.g. red bus in the yellow box) using the reasoning steps of the question. Then we swap the context object feature with other similar object features in the dataset. We perform k swaps based on (a) object class names and (b) object attributes each. The model’s reliance on context can be evaluated with the percentage of answer changes when context gets perturbed.

3.3. SwapMix

Now we introduce our proposed context perturbation strategy: SwapMix. The overall idea is shown in Figure 2. First, we describe the broad idea about the method and then we go into details on how we select candidates for context swapping, how we perform context swapping in terms of object class labels and attributes, and finally, how we apply SwapMix as a training strategy to improve robustness of the models.

After discovering the context objects as described in Section 3.1, we swap their features with other object features from the dataset. For each context object, we perform two types of context swapping based on (a) class label and (b) attributes. In (a) we swap the object feature with the feature of an object from a different class. For example, we change a *bus* into a *car*. In (b) we swap the object feature with feature of an object from same class but with different attributes. For example, we change a *red bus* with *yellow bus*. For both (a) and (b), we perform k feature swaps per context object, therefore altogether we have $2k$ swaps for each context object. We perform context swapping iteratively for each irrelevant object in the image and measure the percentage of answer changes.

We control the swapping process to make sure that the new object is compatible with the image. For example, we may want to change a *bus* into a *car*, but we don’t want to change it into a *computer* because a *computer* parking

on the roadside is unnatural. swapped feature always corresponding to an object from the dataset. The swapped feature resembles a real object and thus this perturbation is equivalent to replacing the irrelevant object with another object in the image. Next, we will provide more details on how we choose candidate features to swap with.

To better describe our feature swapping strategy, we denote each context object feature (each row of context C) as $c_i(o; a_1, a_2, \dots)$. Here o is the class label for the object i , and a_1, a_2, \dots are its attributes. Each object belongs to a unique object class while it can have an arbitrary number of attributes. For example in Figure 2, feature corresponding to the large red bus can be written as $c(\text{bus}; \text{red}, \text{large})$.

Swapping the context class. We swap the object feature with the feature of an object from a different class. For example in Figure 2 we swap the *bus* to *car*, *motorcycle*, etc. This type of feature swapping is similar to putting an object of a different class in place of the irrelevant context object in the image. The context swapping helps us understand the dependency of the VQA model on irrelevant objects in the visual input.

To ensure that the swapped class is in the similar domain as object of interest, we only swap the object into similar classes. To achieve this, for each context object, we find the k nearest classes to its class name. More specifically, we compute cosine similarities between GloVe embeddings of class names and pick the top k class labels. Additionally,

we set a threshold (0.5) to filter out the classes with small similarities². In this way, we get the candidates for swapping each context object and ensure the selected matching classes are in the similar domain. For example, for the class *car*, its top-5 similar classes are: *truck*, *motorcycle*, *vehicle*, *taxi*, *bus*.

For each context object $c_i(o; a_1, a_2, \dots)$ with class o , we select the k nearest class labels to o for swapping as described above. For each of the k classes, we randomly pick one object from the dataset that belongs to that class. This results in k candidate features for swapping: $\hat{c}_i^j(o_j; a_1^j, a_2^j, \dots)$ where $j = \{1, 2, \dots, k\}$. Then by swapping c_i to each of the \hat{c}_i^j , we get k perturbations for each irrelevant context object.

To perturb the perfectly sighted models trained with the encoding of ground truth object class and attributes, a straight-forward way is to simply modify the one-hot encoding for the class label. However, this might cause the object names to be incompatible with attributes, such as *pink elephant* or *green basketball*. Therefore, to ensure that the swapped context corresponds to a real object, we pick a random object of the swapped class from the dataset and use its attributes to generate one-hot encodings for the swapped object attributes.

Swapping the context attributes. To further study the context reliance, we change the object attributes while keeping the object class unchanged. The object feature is swapped with the feature of an object from the same class but with different attributes. For example in Figure 2, the *red bus* can be changed to *orange bus*, *yellow bus*, etc. Compared with object class swapping, attribute swapping can be viewed as a more controlled perturbation that helps reveal the models' reliance on the context in more detail.

For each context object $c_i(o; a_1, a_2, \dots)$, we swap it with k objects of same class label but different attributes. To get the k swapping candidates, we randomly select k objects from the list of objects belonging to same class with different attributes from the dataset³. This results in $\hat{c}_i^j(o; a_1^j, a_2^j, \dots)$ where $j = \{k + 1, k + 2, \dots, 2k\}$. The object o remains the same across all context swaps.

To perturb the model with perfect sight, we just change the one-hot encodings for the attributes. We pick top k attributes which are similar to attribute of interest using GloVe similarity. For example, the attribute *black* can be swapped with: *blue*, *green*, *red*, *purple*, *yellow*.

Algorithm. Let $c_j \in \mathbb{R}^{1 \times d}$ be the context corresponding to j^{th} irrelevant object. The aim is to swap c_j with context

²For cases where the number of matching classes with threshold 0.5 are less than k , we select random classes from the datasets to make number of swaps exactly k . We also tried not padding with random classes. The experimental results are similar for the two settings.

³If the list of objects belonging to same class is less than k , let's say k' , we perform only k' feature swapping for that object.

$c_p(o^p; a_1^p, a_2^p, \dots)$ belonging to an object p from the dataset. We define a matrix for swapping, $S \in \mathbb{R}^{m \times d}$, where each row of S is equal to c_p . We perform feature swapping to convert C to C^p with the following operation :

$$C^p = C \odot P + S \odot P^c$$

$P \in \{0, 1\}^{m \times d}$ is the perturbation matrix and \odot is Hadamard product [32], also known as element wise matrix multiplication. All the rows of P are 1's except j^{th} row corresponding to c_j . P^c is complementary matrix of P , $P^c = J - P$, where J is the matrix with each entry being 1. Thus, all entries of P^c are 0's except for j^{th} row. Effectively, we modify the context C to C^p by changing context of j^{th} row, from c_j to c_p .

Summary. For each context object, we get k swaps for its class labels and another k swaps for attributes. Thus $2k$ context swaps are performed for each irrelevant object. To generate one perturbation for an image, we only perturb one context object at a time. Given m context objects in the image, we perform $m * 2k$ perturbations. This is detailed testing on the model to check if it depends on context for predictions. The results of these $m * 2k$ perturbations are used to measure the robustness of the model.

3.4. SwapMix as a training strategy.

We can further use SwapMix to improve the robustness of the model. We use SwapMix during training to augment the training images. The model sees a new version of the image at every epoch based on context swapping. Using SwapMix with training, we force the model to pay less attention to context, C , and focus on relevant objects in the image to answer the questions.

During training, we swap the feature vectors belonging to context with other feature vectors from the dataset. We identify the context and perform context swapping based on (a) class label and (b) attributes in the same way as explained in the above sections. We perform context swapping on some irrelevant objects. For every irrelevant object, we decide to swap a feature with a probability of $p = 0.5$. If selected for context swapping, we decide if we have to perform context swapping based on the class label with a probability of $p = 0.5$, otherwise, we perform context swapping based on attributes.

SwapMix training can be performed on both models trained with FasterRCNN features and model with perfect sight. We add a new function in the data loading part of training, which changes the context in the image. As we do context swapping for every image during data loading, the training time increases by a factor of 1.4 times. In our analysis, we show that both the robustness of the model and the effective accuracy increases using SwapMix on both FasterRCNN and Perfect Sight embeddings.

	MCAN			LXMERT		
	Acc.	Context Reliance	Effective Acc.	Acc.	Context Reliance	Effective Acc.
Faster RCNN	70.55	45.05	38.77	83.78 ⁴	10.10	75.32
Perfect Sight	90.34	16.40	75.53	91.58	18.85	74.31
Faster RCNN + SwapMix	61.04	26.94	44.61	83.72	7.31	77.60
Perfect Sight + SwapMix	88.10	11.65	77.83	91.45	17.34	75.59

Table 1. Results for diagnosing the context reliance for MCAN [49] and LXMERT [42] models. We study models trained with both FasterRCNN features and perfect sight embeddings. Here *Context Reliance* is the percentage of correctly-answered questions that are successfully perturbed by SwapMix; *Effective Acc.* is the context-robust accuracy.

4. Experiment

4.1. Dataset and Experiment Setup

Dataset. Our experiments are based on the GQA dataset [19]. GQA train split contains 72140 images with 943k questions and val split contains 10243 images with 132k questions. The dataset provides annotated scene graphs for each image and ground-truth reasoning steps for each question. We leverage the reasoning steps to identify visual context and leverage the scene graph annotation to train models with perfect sight. We train the models on GQA train set and test them on GQA val test. GQA also has a test-dev split and a test split, which are not used in our work because they do not have scene graph and reasoning step annotation.

Models. Among the many different VQA models, in this work, we focus on two representative models: MCAN [49] and LXMERT [42]. MCAN is a representative of attention-based models, which contains self-attention and guided-attention units to model the intra-modal and inter-modal interactions between visual and question input. LXMERT is a representative of large-scale pretrained models which can be then finetuned to solve a set of downstream tasks.

Implementation Details. We use the official released code for both MCAN and LXMERT models. We finetune both MCAN and LXMERT pre-trained models using FasterRCNN features on the GQA train set using the default hyperparameters as described by respective authors. For training models with perfect sight, we get ground-truth object names and attributes from scene graph annotation in GQA dataset. MCAN with perfect sight takes a total of 50 epochs to converge and LXMERT model takes 6 epochs. For LXMERT, we use the object features provided by its authors in the official codebase. For MCAN, we used the object features released with GQA dataset.

⁴We note that LXMERT finetuned with Faster RCNN features has higher accuracy and shows high robustness towards SwapMix perturbation. This is because we test both the models on GQA val split and LXMERT was pretrained with five large vision-language datasets where it has seen images in GQA val set during pretraining. This is reported in the codebase. Therefore the LXMERT results with Faster RCNN features needs to be viewed with cautious.

Evaluation metrics. We introduce 2 new metrics to evaluate the model robustness, *context reliance* and *effective accuracy*. As explained in Section 3.3, we apply $2mk$ perturbations for each question where m is the number of irrelevant objects in the image and $2k$ is the number of feature swaps per irrelevant object. We consider a question relying on context if its answer changes for any for the $2mk$ perturbations. Based on this definition, *context reliance* is the percentage of context-relying questions that are originally answered correctly. *Effective Accuracy* is the percentage of questions that are consistently predicted correctly and survive all $2mk$ SwapMix perturbations. Mathematically, it can be written as $effective\ Acc = \sum_i^N q_i / N$, where N is the total number of questions in the dataset and q_i is defined as:

$$q_i = \begin{cases} 0, & \text{if } gt \neq Model(V^j, Q) \text{ for any perturbation } j \\ 1, & \text{otherwise.} \end{cases}$$

4.2. SwapMix Perturbation Results

We finetune the MCAN model and LXMERT model on GQA training split with object features extracted by pretrained Faster RCNN. After finetuning, MCAN reaches 70.55% accuracy and LXMERT reaches 83.78% accuracy on GQA validation split. These results are comparable with ones reported by original authors.

Then we perform SwapMix perturbation to extensively test models' reliance on context. The evaluation results for both the MCAN model and LXMERT model are shown in Table 1. For measuring robustness, context reliance and the effective accuracy are reported. Surprisingly, 45% of MCAN answers get changed after perturbation and the effective accuracy drops significantly from 70.55% to 38.77%. The significant drop suggests that the MCAN model relies heavily on the context and is not robust to context swapping. On the contrary, LXMERT is more robust. We conjecture this is because LXMERT is pretrained on a large amount of image-text pairs from five vision-and-language datasets [42] and the large-scale pretraining equipped the model with better robustness.

Next, we study VQA models with perfect sight. We train both models with perfect sight using encodings of ground

		MCAN		LXMERT	
		Class Reliance	Attr Reliance	Class Reliance	Attr Reliance
k=5	Faster RCNN	32.52	28.41	7.29	7.61
	Faster RCNN + SwapMix	18.19	17.10	5.78	5.94
	Perfect Sight	11.11	3.16	14.57	1.34
	Perfect Sight + SwapMix	7.64	2.36	12.89	1.32
k=10	Faster RCNN	39.40	34.47	8.18	8.81
	Faster RCNN + SwapMix	22.18	20.91	6.27	6.58
	Perfect Sight	15.52	3.71	18.82	1.35
	Perfect Sight + SwapMix	10.89	2.72	17.28	1.36

Table 2. Detailed analysis of reliance on context. Here we measure the reliance on (a) class names and (b) attributes of irrelevant objects on model prediction. We provide analysis on k perturbations on context for each irrelevant object.

truth object names and attributes. As shown in table 1, both models achieve more than 90% accuracy with perfect sight. We observe a significant improvement in robustness of MCAN with perfect sight: its context reliance drops by 28.7% compared with training on Faster RCNN features (from 45.1% to 16.4%) and the effective accuracy improves from 38.77% to 75.53%. This suggests that models trained with perfect sight are more robust than its FasterRCNN counterparts when trained with the same amount of data. It is also noticeable that the LXMERT performance is in a similar range with MCAN, which suggests that LXMERT is no more robust than MCAN without seeing more pretraining data in the same domain.

In Table 2, we provide more detailed results of perturbations on object class and attribute separately. Interestingly, we observe that models with perfect sight are highly robust to *attribute* perturbations: only 3.7% and 1.4% of the correct answers get changed by attribute perturbation for MCAN and LXMERT respectively. This suggests that given the ground-truth class name, the model can distinguish the relevant and irrelevant objects well, thus are robust to perturbation on the attributes of context objects.

To further support our claim of generalisation of SwapMix, we tested our approach on OSCAR [27]. We see 26.3% of OSCAR answers relies on visual context. The results are consistent with our initial results that transformer models are more robust than MCAN.

4.3. SwapMix Training Results

Using SwapMix as a training data augmentation strategy consistently improves the robustness of both models in all settings. For both MCAN and LXMERT, trained with both FasterRCNN features and perfect encodings, SwapMix training reduces the models’ reliance on context and boosts the effective accuracy.

As shown in Table 1 (marked as +SwapMix), SwapMix training significantly decreases the context reliance of MCAN by 40% (from 45% to 27%) and increases its effective accuracy by 5.8% (row 3). The results are also con-

sistent for MCAN with perfect sight and LXMERT. Table 2 further shows that SwapMix training improves robustness in both context class reliance and attribute reliance. The results consistently show that SwapMix as a training strategy decreases model reliance on context, encourages model robustness and improves effective accuracy.

Interestingly, we notice that there is a trade-off between model robustness and overall accuracy. While we see significant improvement in model robustness, it is noticeable that the overall model accuracy drops to some extent. For example, when applying SwapMix training to MCAN model with perfect sight, its context reliance reduces by 4.7% and effective accuracy improves by 2.3%, while the overall model accuracy drops by 2.2%. The model utilizes biases and correlations in context to achieve high performance, thus when the context reliance is reduced by SwapMix training, the effective accuracy improves while the overall accuracy drops. Hereby we suggest that the effective accuracy is a better description of the models’ true ability to understand the task without relying on context bias.

4.4. Ablations and Analysis

Ablating the swapping number k . In Table 2, we additionally provide ablation study results for the hyperparameter k , which is the perturbation number. The results for $k = 5$ or $k = 10$ are shown in the table. We do k perturbations on class names and attributes of context objects and report the percentage of questions affected. The result shows that when we increase the perturbations number of k from 5 to 10, the reported answer changes increase accordingly for both models, which is expected. Whereas it is also notable that the reliance increase is not significant, showing that most reliance on context can be revealed with a relatively small number of perturbations. By default, we use $k=10$ to benchmark reliance on the context of VQA models.

Random padding to k swaps. When doing object name perturbation, for cases where number of compatible classes is less than k , we select random classes from the dataset to pad the perturbation number to exactly k . To verify that

this random padding does not bring extra noise in the result, we compare results with and without random padding. As shown in table 3, the effect of random padding is negligible.

MCAN			
		Random	w/o random
k=10	FRCNN	45.1	40.3
	FRCNN + SwapMix	26.9	24.3
k=5	FRCNN	38.1	35.0
	FRCNN + SwapMix	22.4	20.8

Table 3. Context reliance measured by SwapMix with and without random padding to generate k perturbations. This table verifies that random padding does not lead to significant difference.

Examples for SwapMix Perturbation. In Figure 3, we show examples for our proposed SwapMix perturbation. Example (a) is based on class name swapping and example (b) is based on attribute swapping, both of which resulted in the change of model prediction. In example (a), the boot is irrelevant to the question about sweater color, while changing boots into snow boots results in a change in model prediction. In example (b), when we swap the blue signboard in the background with a green signboard, the model prediction on the short’s color changed to green as well. The examples are based on the results of MCAN model with perfect sight. The examples intuitively show that VQA models rely heavily on context and by perturbing irrelevant context in the image, we can change model prediction.

Attention visualization for SwapMix training. Training using SwapMix as data augmentation reduces the models’ reliance on context. In Figure 4, we show the visualization of attention weights for models trained without SwapMix and models trained using SwapMix as data augmentation. The visualization is based on the LXMERT model with perfect sight. For the given question, “Is the camera silver or tan”, a model with vanilla training pays more attention to ir-

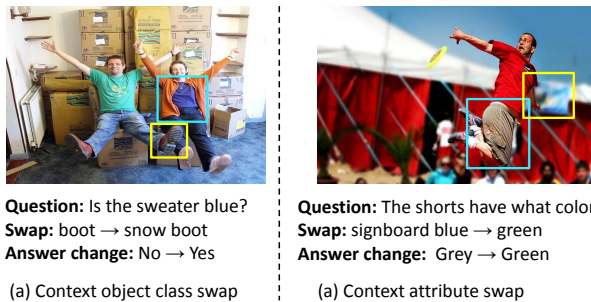


Figure 3. Examples for SwapMix perturbations on GQA val split. Blue boxes show relevant objects and yellow boxes show context objects. In (a) we perform class name perturbation and change boots to snow boots. In (b) we perform attribute perturbation and change color of signboard from blue to green. Both these SwapMix perturbations change model prediction.

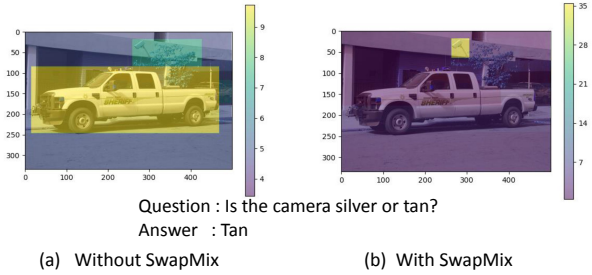


Figure 4. Visualization of attention weights for models (a) without SwapMix training (b) with SwapMix training. SwapMix training effectively suppresses models’ reliance on visual context.

relevant context objects such as the car, the tree, etc., while model trained with SwapMix augmentation focuses highly on the relevant object, camera, and pays very little attention to other objects. The visualization qualitatively shows that when applied as data augmentation strategy, SwapMix effectively suppresses the model’s dependency on visual context and forces the model to focus more on relevant objects.

5. Conclusion

In this work, we study the reliance of VQA models on context, i.e. irrelevant objects in the image for prediction. We propose a simple yet effective perturbation technique: SwapMix. SwapMix is effective in both diagnosing model robustness on context reliance, and regularizing the context reliance of VQA models thus making them more robust. Our experiments of two representative models on GQA show the effectiveness of SwapMix. Interestingly, we find that the robustness of VQA models highly depends on the quality of visual perception and models with perfect sight are more robust to context perturbation. Large-scale pretraining also helps improve model robustness. We hope that our initial analysis on reliance on visual context can serve as a starting point for future researchers to study VQA robustness and reliability.

Negative impact and limitations. Our work study the robustness of VQA models and find that the models are vulnerable to context perturbations. The proposed SwapMix perturbation strategy may be used maliciously to attack VQA models. To overcome this potential negative impact, we suggest that training with SwapMix can effectively regularize reliance on context and that better visual representation may improve model robustness. The limitation of our work is that we only study two representative models, using two types of visual features on the GQA dataset.

Acknowledgements

This work is supported by NSF 1763705.

References

- [1] Aishwarya Agrawal, Dhruv Batra, and Devi Parikh. Analyzing the behavior of visual question answering models. *arXiv preprint arXiv:1606.07356*, 2016. 1, 2
- [2] Aishwarya Agrawal, Dhruv Batra, Devi Parikh, and Anirudha Kembhavi. Don't just assume; look and answer: Overcoming priors for visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4971–4980, 2018. 1, 2
- [3] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6077–6086, 2018. 3
- [4] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433, 2015. 2
- [5] Dzmitry Bahdanau, Harm de Vries, Timothy J O'Donnell, Shikhar Murty, Philippe Beaudoin, Yoshua Bengio, and Aaron Courville. Closure: Assessing systematic generalization of clevr models. *arXiv preprint arXiv:1912.05783*, 2019. 2
- [6] Hedi Ben-Younes, Rémi Cadene, Matthieu Cord, and Nicolas Thome. Mutan: Multimodal tucker fusion for visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2612–2620, 2017. 2
- [7] Rémi Cadène, Hedi Ben-younes, Matthieu Cord, and Nicolas Thome. Murel: Multimodal relational reasoning for visual question answering. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1989–1998, 2019. 1
- [8] Remi Cadene, Corentin Dancette, Matthieu Cord, Devi Parikh, et al. Rubi: Reducing unimodal biases for visual question answering. *Advances in neural information processing systems*, 32:841–852, 2019. 2
- [9] Paola Cascante-Bonilla, Hui Wu, Letao Wang, Rogerio Feris, and Vicente Ordonez. Simvqa: Exploring simulated environments for visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2
- [10] Kan Chen, Jiang Wang, Liang-Chieh Chen, Haoyuan Gao, Wei Xu, and Ram Nevatia. Abc-cnn: An attention based convolutional neural network for visual question answering. *arXiv preprint arXiv:1511.05960*, 2015. 2
- [11] Long Chen, Xin Yan, Jun Xiao, Hanwang Zhang, Shiliang Pu, and Yueting Zhuang. Counterfactual samples synthesizing for robust visual question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10800–10809, 2020. 2
- [12] Christopher Clark, Mark Yatskar, and Luke Zettlemoyer. Don't take the easy way out: Ensemble based methods for avoiding known dataset biases. *arXiv preprint arXiv:1909.03683*, 2019. 2
- [13] Corentin Dancette, Rémi Cadène, Damien Teney, and Matthieu Cord. Beyond question-based biases: Assessing multimodal shortcut learning in visual question answering. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1554–1563, 2021. 2
- [14] Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach. Multimodal compact bilinear pooling for visual question answering and visual grounding. *arXiv preprint arXiv:1606.01847*, 2016. 2
- [15] Tejas Gokhale, Pratyay Banerjee, Chitta Baral, and Yezhou Yang. Mutant: A training paradigm for out-of-distribution generalization in visual question answering. *arXiv preprint arXiv:2009.08566*, 2020. 2
- [16] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6904–6913, 2017. 1, 2
- [17] Junjun He, Zhongying Deng, Lei Zhou, Yali Wang, and Yu Qiao. Adaptive pyramid context network for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7519–7528, 2019. 3
- [18] Lisa Anne Hendricks, Kaylee Burns, Kate Saenko, Trevor Darrell, and Anna Rohrbach. Women also snowboard: Overcoming bias in captioning models. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 771–787, 2018. 2
- [19] Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6700–6709, 2019. 2, 3, 6
- [20] Corentin Kervadec, Grigory Antipov, Moez Baccouche, and Christian Wolf. Roses are red, violets are blue... but should vqa expect them to? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2776–2785, 2021. 1, 2
- [21] Corentin Kervadec, Theo Jaunet, Grigory Antipov, Moez Baccouche, Romain Vuillemot, and Christian Wolf. How transferable are reasoning patterns in vqa? *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4205–4214, 2021. 1, 2
- [22] Jin-Hwa Kim, Jaehyun Jun, and Byoung-Tak Zhang. Bilinear attention networks. *arXiv preprint arXiv:1805.07932*, 2018. 2
- [23] Jin-Hwa Kim, Kyoung-Woon On, Woosang Lim, Jeonghee Kim, Jung-Woo Ha, and Byoung-Tak Zhang. Hadamard product for low-rank bilinear pooling. *arXiv preprint arXiv:1610.04325*, 2016. 2
- [24] Adam Kortylewski, Ju He, Qing Liu, and Alan L Yuille. Compositional convolutional neural networks: A deep architecture with innate robustness to partial occlusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8940–8949, 2020. 3
- [25] Adam Kortylewski, Qing Liu, Angtian Wang, Yihong Sun, and Alan Yuille. Compositional convolutional neural net-

- works: A robust and interpretable model for object recognition under occlusion. *International Journal of Computer Vision*, 129(3):736–760, 2021. 3
- [26] Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language. *ArXiv*, abs/1908.03557, 2019. 1
- [27] Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, et al. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *European Conference on Computer Vision*, pages 121–137. Springer, 2020. 1, 2, 7
- [28] Di Lin, Ruimao Zhang, Yuanfeng Ji, Ping Li, and Hui Huang. Scn: switchable context network for semantic segmentation of rgb-d images. *IEEE transactions on cybernetics*, 50(3):1120–1131, 2018. 3
- [29] Xin Lin, Changxing Ding, Jinquan Zeng, and Dacheng Tao. Gps-net: Graph property sensing network for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3746–3753, 2020. 3
- [30] Mateusz Malinowski and Mario Fritz. A multi-world approach to question answering about real-world scenes based on uncertain input. *Advances in neural information processing systems*, 27:1682–1690, 2014. 2
- [31] Varun Manjunatha, Nirat Saini, and Larry S Davis. Explicit bias discovery in visual question answering models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9562–9571, 2019. 2
- [32] Elizabeth Million. The hadamard product. 2007. 5
- [33] Yulei Niu, Kaihua Tang, Hanwang Zhang, Zhiwu Lu, Xian-sheng Hua, and Ji-Rong Wen. Counterfactual vqa: A cause-effect look at language bias. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12695–12705, 2021. 1, 2
- [34] Yulei Niu, Kaihua Tang, Hanwang Zhang, Zhiwu Lu, Xian-Sheng Hua, and Ji-Rong Wen. Counterfactual vqa: A cause-effect look at language bias. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12700–12710, June 2021. 1, 2
- [35] Aude Oliva and Antonio Torralba. The role of context in object recognition. *Trends in cognitive sciences*, 11(12):520–527, 2007. 3
- [36] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014. 3
- [37] Sainandan Ramakrishnan, Aishwarya Agrawal, and Stefan Lee. Overcoming language priors in visual question answering with adversarial regularization. *arXiv preprint arXiv:1810.03649*, 2018. 2
- [38] Guanghui Ren, Lejian Ren, Yue Liao, Si Liu, Bo Li, Jizhong Han, and Shuicheng Yan. Scene graph generation with hierarchical context. *IEEE Transactions on Neural Networks and Learning Systems*, 32(2):909–915, 2020. 3
- [39] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:1137–1149, 2015. 3
- [40] Ramprasaath R Selvaraju, Stefan Lee, Yilin Shen, Hongxia Jin, Shalini Ghosh, Larry Heck, Dhruv Batra, and Devi Parikh. Taking a hint: Leveraging explanations to make vision and language models more grounded. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2591–2600, 2019. 2
- [41] Himanshu Sharma and Anand Singh Jalal. Visual question answering model based on graph neural network and contextual attention. *Image and Vision Computing*, 110:104165, 2021. 3
- [42] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490*, 2019. 1, 2, 6
- [43] Antonio Torralba, Kevin P Murphy, William T Freeman, and Mark A Rubin. Context-based vision system for place and object recognition. In *Computer Vision, IEEE International Conference on*, volume 2, pages 273–273. IEEE Computer Society, 2003. 3
- [44] 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*, pages 5998–6008, 2017. 2
- [45] Angtian Wang, Yihong Sun, Adam Kortylewski, and Alan L Yuille. Robust object detection under occlusion with context-aware compositionalnets. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12645–12654, 2020. 3
- [46] Wenbin Wang, Ruiping Wang, Shiguang Shan, and Xilin Chen. Sketching image gist: Human-mimetic hierarchical scene graph generation. In *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIII 16*, pages 222–239. Springer, 2020. 3
- [47] Jialin Wu and Raymond J Mooney. Self-critical reasoning for robust visual question answering. *arXiv preprint arXiv:1905.09998*, 2019. 2
- [48] Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng, and Alex Smola. Stacked attention networks for image question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 21–29, 2016. 2
- [49] Zhou Yu, Jun Yu, Yuhao Cui, Dacheng Tao, and Qi Tian. Deep modular co-attention networks for visual question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6281–6290, 2019. 1, 2, 6
- [50] Zhou Yu, Jun Yu, Jianping Fan, and Dacheng Tao. Multimodal factorized bilinear pooling with co-attention learning for visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 1821–1830, 2017. 2
- [51] Zhou Yu, Jun Yu, Chenchao Xiang, Jianping Fan, and Dacheng Tao. Beyond bilinear: Generalized multimodal factorized high-order pooling for visual question answering. *IEEE transactions on neural networks and learning systems*, 29(12):5947–5959, 2018. 2

- [52] Qiaoyong Zhong, Chao Li, Yingying Zhang, Di Xie, Shikai Yang, and Shiliang Pu. Cascade region proposal and global context for deep object detection. *Neurocomputing*, 395:170–177, 2020. [3](#)
- [53] Bolei Zhou, Yuandong Tian, Sainbayar Sukhbaatar, Arthur Szlam, and Rob Fergus. Simple baseline for visual question answering. *arXiv preprint arXiv:1512.02167*, 2015. [2](#)