Found a Reason for me? Weakly-supervised Grounded Visual Question Answering using Capsules

Aisha Urooj Khan[1,2], Hilde Kuehne[2], Kevin Duarte[1], Chuang Gan[2], Niels Lobo[1], Mubarak Shah[1]
[1] CRCV, University of Central Florida , [2] MIT-IBM Watson AI Lab

Abstract

The problem of grounding VQA tasks has seen an increased attention in the research community recently, with most attempts usually focusing on solving this task by using pretrained object detectors. However, pre-trained object detectors require bounding box annotations for detecting relevant objects in the vocabulary, which may not always be feasible for real-life large-scale applications. In this paper, we focus on a more relaxed setting: the grounding of relevant visual entities in a weakly supervised manner by training on the VQA task alone. To address this problem, we propose a visual capsule module with a query-based selection mechanism of capsule features, that allows the model to focus on relevant regions based on the textual cues about visual information in the question. We show that integrating the proposed capsule module in existing VQA systems significantly improves their performance on the weakly supervised grounding task. Overall, we demonstrate the effectiveness of our approach on two state-of-the-art VQA systems, stacked NMN and MAC, on the CLEVR-Answers benchmark, our new evaluation set based on CLEVR scenes with groundtruth bounding boxes for objects that are relevant for the correct answer, as well as on GQA, a real world VQA dataset with compositional questions. We show that the systems with the proposed capsule module consistently outperform the respective baseline systems in terms of answer grounding, while achieving comparable performance on VQA task.[1]

1. Introduction

VQA systems have now matured to the point where their usage is increasing in real life applications such as answering questions based on radiology images [1], helping visually impaired people [10], and human-robot interactions [38]. However, with the increasing maturity of such systems, it also becomes important to know how the answer is actually generated in order to assess if it is based on the right cues

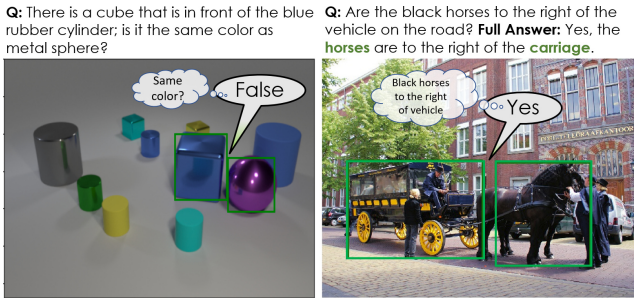

Figure 1. **Problem definition:** Given an input image and a question, we want to answer the question as well as localize the evidence (shown in green boxes) with VQA supervision alone. Best viewed in color.

or not. If the question is "Are there black horses to the right of the vehicle?" (see figure 1), it may be important to know if the answer is generated because the network found black horses at the right place in the image or not. This allows to judge the overall correctness beyond simply evaluating the textual answer. Recent works [17, 36, 4, 20] try to address this problem by starting to evaluate not only the VQA accuracy, but also the accuracy of grounding that the answer is based on. The grounding of an answer is usually assessed by considering the respective attention map of the image for the given answer, and by evaluating if the objects that are relevant for the right answer are attended to or not.

To achieve good grounding accuracy, most approaches in this field rely on input feature maps from object detection models that are pretrained with the relevant object classes. This restricts the scope to known object classes such as MS COCO [31], or require to annotate the regions of relevant objects, and to pretrain an object detector for them[17]. Only few attempts have been made so far to address this problem to train both, the VQA as well as the grounding, without pretrained object detection based on the information of the VQA task alone as e.g. in context of the GQA dataset by only using spatial (appearance) features [17]. This paper focuses on exactly this scenario: weakly supervised visual grounding based on VQA supervision. The idea here is that both tasks, the visual question-answering as well as the correct visual grounding, should be learned

[1]Code will be available at https://github.com/aurooj/WeakGroundedVQA_Capsules.git

from the VQA task alone. Hence, we do not use any object-level information as an input or in supervision.

The correct grounding in this case is usually based on two major tasks, finding the relevant visual instances and, usually, modeling the relation between those instances as seen in figure 1. To address this problem, we propose extending current VQA frameworks with capsules. Capsule networks were introduced by Sabour *et al*. [39], and have shown promising results for image interpretability [25] and segmentation in various fields such as 3D point clouds [48], videos [7] and medical images [28]. This is the result of capsule layers' ability to learn part-to-whole relationships for object entities through routing-by-agreement. We believe this capability to model objects and their relations qualifies capsules as a good choice for addressing the problem of weakly-supervised grounding in VQA.

Current capsule-based methods follow the practice of adding capsule layers on top of convolutional features, and training them with object class supervision. A discrete and supervised masking operation, *i.e*. masking all capsules except the ground-truth class capsule, is often applied to reconstruct or segment the object corresponding to the given class. In case of weak VQA grounding, no class or object based supervision is available; only an embedding of a natural language question is given. Therefore, we propose a "soft-masking" procedure which selects the capsule(s) based on the input question. For example, if the reasoning operation is $Find$("blue spheres"), the soft-masking operation will mask all capsules not representing the *"blue spheres"*. Once the irrelevant capsules are masked, the capsule representations are passed to future reasoning operations to complete the VQA task.

To evaluate VQA systems for their answer grounding ability, we consider two datasets, the recently proposed GQA dataset [17] as well as the CLEVR dataset [23]. To allow the evaluation of grounding accuracy on CLEVR, we propose a new CLEVR validation set, named CLEVR-Answers. CLEVR-Answers provides VQA pairs with the respective ground truth bounding boxes for all objects that the answer is based on. Note that, as we are not interested in using any object annotations during training, *we only need ground truth bounding boxes during evaluation, but not during training*. The idea is, thus, to train on the standard CLEVR training set and to learn visual representations of objects during this training without further annotation. We use this new evaluation set to test current state-of-the-art frameworks, MAC [16] and Stacked NMN [13] with respect to their grounding abilities. We show that, although all frameworks perform at the same level with respect to VQA accuracy, there are major differences with respect to their grounding abilities. We show that using capsules with soft query-based masking significantly improves existing methods' grounding abilities.

2. Related Work

VQA and visual grounding Recent approaches for VQA task rely on object level features as input to improve the VQA accuracy [2, 14, 26, 45, 21, 40, 8, 15, 44]. Those features are extracted from pretrained object detectors. This makes the VQA task easier and usually performs better than spatial or appearance features, but it also adds an additional preprocessing step (detecting objects) to the pipeline. Additionally, since the pretraining relies on the object classes in the training set, it limits the extension of such methods to datasets with object-level annotation. Basic appearance or grid-based features, *e.g*., based on a backbone pretrained on ImageNet, are easier to generate and have recently been shown to work as well as object level features [19] for the VQA task. All these approaches usually only focus on the accuracy of the VQA task, and *do not evaluate the respective grounding of their answers*.

Focusing on this capability, several VQA datasets now provide grounding labels such as GQA [17], VCR [46], VQS [8], CLEVRER [42, 5] and TVQA+ [29]. Here, object annotations are either provided for all objects in the visual input, or only for the objects relevant to both question and answer. Out of those, GQA specifically focuses on the evaluation of grounding accuracy with and without object detection supervision and attempts to evaluate MAC [16] and BottomUp [2] for their grounding ability in natural images. We, therefore, choose GQA to evaluate capsule-augmented systems in real world for weakly supervised grounding. Additionally, we compute the answer grounding in terms of overlap and IOU to measure how precise this grounding is in correlation to the answer.

VQA and visual reasoning on CLEVR CLEVR [23] is a diagnostic visual reasoning dataset with compositional questions to test performance of VQA systems on a variety of reasoning skills. Since the introduction of CLEVR, a large number of VQA systems [13, 24, 43, 16, 33, 34, 41] have surfaced to solve VQA task on this benchmark achieving near perfect VQA accuracy [22]. One line of works [24, 13, 34] uses reasoning layouts supervision provided with the image-QA pairs. Additionally, neuro-symbolic approaches over object level features are proposed, *e.g*., in [43, 33]. Many of those ideas implicitly or explicitly include the concept of grounding on this dataset, but usually rely on a pretrained object detector to generate initial object attention maps. Several other variants of CLEVR have also emerged trying to solve various problems such as CLEVR-CoGent [23], CLEVR-Dialog [27], CLOSURE [3], simply-CLEVR [36], and CLEVR-Ref+ [32], with CLEVR-Ref+ focusing on grounding based on referential expressions, but not VQA, and simply-CLEVR providing only grounding labels for one or all objects in the image. We, on the other hand, provide bounding box labels for all question types

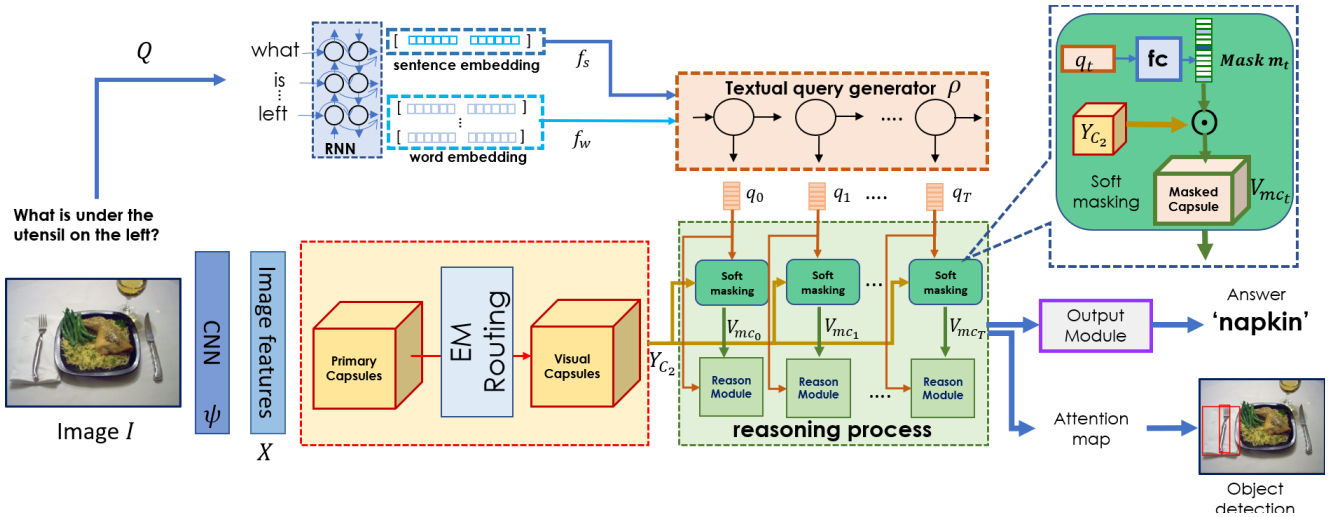

Figure 2. Overview of our pipeline: given the question-image pair, we obtain image features X using an image encoder ψ and question features (both sentence f_s and word embeddings f_w) using an RNN ρ. These question-based features are then input to a multi-hop reasoning module which generates T textual queries $q_0, q_1, ..., q_T$. The Primary Capsules layer then transforms the convolutional image features into capsules (each capsule has a $k \times k$ pose matrix and an activation weight). The primary capsules use a routing-by-agreement algorithm to vote for higher level capsules at each spatial location. These capsules are used as visual representation inside the reasoning process. At each timestep $t \in \{1, 2, ..., T\}$, a soft masking module first masks irrelevant capsules using textual query q_t to select a subset of capsules at each spatial location denoted as V_{mc_t}. Then, the reasoning operation is performed over selected capsules (using attention combined with other reasoning). Output of the reasoning operation is a vector. Output Module then aggregates these outputs and predicts the answer. We train the system with VQA supervision only. At inference time, we post-process the attention maps produced by the reasoning modules to obtain grounding predictions.

without imposing any constraint on the number of objects relevant to answer grounding. Thus, CLEVR-Answers enables us to evaluate grounding abilities of current state-of-the-art methods without any constraints.

Capsule networks Hinton *et al.* [11] first proposed capsule networks to learn vectors of view equivariant features from images. More recently, Sabour *et al.* [39] extended capsule networks with an iterative routing-by-agreement algorithm to classify and segment multiple digits within an image. Several works have proposed improved methods for routing [12, 47, 30, 18] as well as have applied capsule networks to different tasks and domains [6, 28, 48, 35]. While most previous works tend to be supervised by calculating a loss over a set of "class capsules", our proposed approach does not have this capsule-to-object supervision; rather, capsules are incorporated in our system as intermediate layers and are learned by using weak supervision from question answers. Several capsule networks which perform classification tasks [39, 6, 37] tend to use a masking operation to ensure capsules learn class-specific representations. This operation masks all capsule pose values to 0 except for the selected (i.e. ground-truth or predicted) class capsule, and uses this masked representation to reconstruct or segment the input image or video. Since there are no ground-truth class annotations, we propose a novel soft-masking operation which effectively selects the capsule(s) relevant to the input query and masks irrelevant capsules.

3. Proposed Approach

3.1. Problem Formulation

Given an input image I and a question Q, our goal is to output the correct answer $a \in A$, where A denotes the answer vocabulary, and B bounding box predictions for the objects which led to the answer a. Figure 1 illustrates the problem. Our pipeline is explained in the following sections. An overview of the framework is given in figure 2.

3.2. Input Embeddings

Question embedding We are given a question Q of words $w_1, w_2, ..., w_l$, where l is the length of Q in words. Let V be the vocabulary for question words in the training set with a lookup embedding $E \in \mathbb{R}^{|V| \times d_e}$. Each word in Q is represented by a d_e-dimensional initial embedding vector. Let $\phi(Q, [w_1, w_2, ..., w_l])$ be a sentence encoder which outputs both sentence level embedding f_s for Q as well as word-level features f_w. These sentence-level and word-level embeddings are then input to our system. Following previous works [13, 16], we choose ϕ to be a BiLSTM. Output dimensions for sentence embedding f_s and word embeddings f_w are, therefore, $f_s \in \mathbb{R}^{d_q}$, $f_w \in \mathbb{R}^{d_q}$, where $d_q = 2 \times d$, and d is the dimension of sentence encoder.

Image embedding Given an input image I, we compute

a feature map $X = \psi(I)$, where ψ is a pretrained image encoder and $X \in \mathbb{R}^{H \times W \times d_f}$ denotes the features extracted for I (d_f is the feature dimension).

3.3. Textual query generator

To answer a question based on an image, a VQA system performs attentional parsing of the question, i.e. attends to selected words from the question iteratively depending on the reasoning required to answer the question. This approach of splitting the question into subqueries is often termed as multi-hop or recurrent reasoning, where a query is generated at each reasoning step to attend to the image to collect answer-relevant knowledge. Let ρ be our query generator which takes sentence embedding, f_s, and word embeddings, f_w, as an input at each time step t ($t = 1, 2, .., T$), and outputs query q_t as an output.

$$q_t = \rho(f_s, f_w), \forall t \in \{1, 2, ..., T\}. \tag{1}$$

More details are discussed in the supplementary.

3.4. Capsules with soft masking

A capsule is a group of neurons representing an entity or a part of an entity. In this work, we use matrix capsules [12], which are composed of a logistic unit (called the activation) and a 4x4 pose matrix (called the pose). The activation indicates the presence of a specific entity, whereas the pose represents the entity's properties. A capsule layer consists of many capsules, which use a routing-by-agreement algorithm to vote for capsules in the following layer in order to model part-to-whole relationships. Matrix capsules use EM-Routing algorithm for capsule routing. We integrate them into the process as follows.

Visual capsules: From the image embedding, X, the primary capsules are obtained by using a learned convolution operation resulting in C_1 capsule types each with a 4x4 pose matrix and an activation for each spatial position. The output dimension of the primary capsule layer is $\mathbb{R}^{H \times W \times C_1 \times 4 \times 4}$ and $\mathbb{R}^{H \times W \times C_1 \times 1}$ for poses and activations respectively. To obtain a higher-level capsule representation, we perform EM-routing over primary capsules to obtain a set of C_2 capsules at each spatial position. These capsules model different objects within the scene (including the background). Output dimensions for poses and activations are $\mathbb{R}^{H \times W \times C_2 \times 4 \times 4}$ and $\mathbb{R}^{H \times W \times C_2 \times 1}$ respectively. They are used as the visual representation of the input image in future steps.

Soft masking: The trivial approach of leveraging this capsule representation for VQA would be to group the poses and activations to form a tensor of shape $\mathbb{R}^{H \times W \times C_2 \times (4 \times 4 + 1)}$, and use them as a single feature map like standard convolution-based methods. Although, this performs decently well, it is not an ideal solution since it treats each dimension of the capsule poses as independent features and disregards the fact that all dimensions in the capsule pose represent a single object or entity.

Instead of this independent feature selection, we propose the selection of individual capsules based on the question. This is achieved by masking capsules which are irrelevant to the reasoning operation. Previous capsule methods use masking for image reconstruction [12] or segmentation [7], however they require ground-truth class labels to select the single capsule type which is not masked. Since no object/class-level supervision is present for this task, we propose learning which capsules should be masked in an end-to-end manner. For each reasoning step, a fully connected layer generates a set of C_2 logits denoting each capsule types' relevance for the given query. Mathematically, this can be defined as:

$$m_{t_{\text{logits}}} = \eta(q_t), \tag{2}$$

where q_t is the textual query at reasoning step t, and η is the fully connected layer. Then, a one-hot mask, $m_t \in \mathbb{R}^{C_2}$ is generated where $m_i = 1$ for $i = argmax(m_{t_{\text{logits}}})$. This mask is then applied to the visual capsule layer:

$$V_{mc_t} = m_t \odot Y_{c_2}, \tag{3}$$

where, Y_{c_2} is the output of visual capsules layer, and V_{mc_t} are the masked visual capsules corresponding to textual query q_t. We call this operation hard masking.

We find that hard masking operation leads to sub-optimal performance, since the lack of supervision leads to some capsule never being selected, resulting in poor representations. To remedy this, we present a novel soft masking method as visualized in the green box in figure 2, which allows gradients to flow through all capsules. Instead of creating a one-hot mask, a softmax operation is used on the logits to create a set of soft weights, which then mask the visual capsules, as follows:

$$V_{mc_t} = softmax(\eta(q_t)) \odot Y_{c_2}. \tag{4}$$

These masked visual capsules are then used for reasoning operations as defined by their respective modules. We show that incorporating capsules and soft masking within an attentional VQA system can boost its grounding ability significantly without compromising VQA accuracy, therefore, reducing the performance-explainability trade-off.

3.5. Output module

The reasoning modules output features which are aggregated over reasoning steps and sent to an output module i.e., a classifier which outputs answer scores. For grounding predictions, we consider the spatial attention maps produced by reasoning modules and post-process them to obtain the object detections. The post-processing is described in the following section.

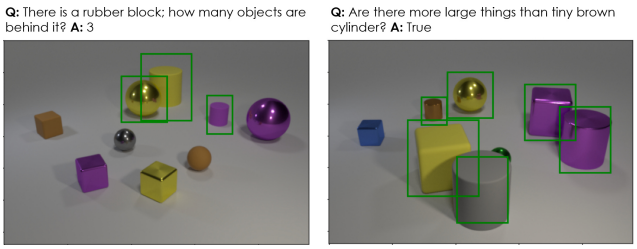

Figure 3. Sample images with QA pair and generated answer grounding labels for CLEVR-Answers dataset.

4. Implementation details

We integrate capsules into two baseline VQA systems: Stacked Neural Module Networks [13] and MAC [16]. We make the following architectural changes to these systems.

Capsules with MAC. MAC [16] is a recurrent reasoning architecture which performs T reasoning steps to answer the question. Each reasoning step involves generating a question-based control signal (textual query), using this control signal to read from image features (using attention), and writing memory. The final output after T reasoning steps is then combined with the question and goes into the answer classifier. MAC also produces interpretable attention maps for explaining the reasoning process behind VQA. For capsule integration into the MAC cell, we make the following changes: First, capsule layers are added on top of the convolutional layer to obtain visual capsules from image features. The read module is responsible for attending to spatial image features and retrieving query-relevant image features, based on the previous output and the current control signal (question based feature at timestep t). Inside the read module, we first map the control signal to a feature vector of dimensions $C_2 \times (4 \times 4 + 1)$ using a trainable linear layer. This feature vector is then used to generate a soft mask to obtain only query-related capsules for further reasoning. Weights for the masking layer are shared among MAC cells. These masked capsules are then used for further reasoning inside the *read* module.

Capsules with Stacked Neural Module Network (SNMN). Stacked neural module network is an attentional VQA method following the same reasoning pipeline as explained above. SNMN produces human interpretable attention maps. SNMN trains convolutional layers on pretrained image features. The output of these convolutional layers then goes into the reasoning modules and uses textual query to perform the reasoning operation producing an attention map as output. To integrate capsules into SNMN, we append our capsule module on top of image features to obtain C_2 visual capsules. Instead of convolutional image features, reasoning modules now perform their reasoning operation on capsules. For query-based soft masking, each neural module has a fully connected layer which takes textual query $q_t \in \mathbb{R}^d$ as input and outputs a feature vector of dimension $C_2 \times (4 \times 4 + 1)$. This feature vector is then used to generate capsule mask of size C_2, and for further interaction between query and masked capsules. Each reasoning module in SNMN has its own masking layer except $Scene$, And, and Or, since these modules do not use the textual parameter in their computations.

Generation of attention maps. During training, capsule layers learn to attend to different visual cues in the image, including background regions when no grounding evidence is available for the answer. In order to give more weight to high attention regions and suppress attention on the background, we introduce an opacity parameter α. For uniform attention regions, opacity is scaled up by α. After post-processing spatial attention using α, an attention threshold of 0.5 is applied to get a binary mask with high attention regions. Each connected component in this binary mask is considered an object detection. See supplementary for results w.r.t. variations in α.

5. Datasets

We perform our evaluation on two datasets: GQA and CLEVR. GQA, as real world dataset for visual reasoning and compositional question answering, combines the two aspects of the proposed idea by providing an evaluation of grounding VQA tasks. It also provides a baseline for the weakly supervised grounding task by using only spatial features that are not pretrained on object annotations. CLEVR, as opposed to GQA, provides a sandbox for visual reasoning VQA tasks with synthetic images only, no visual overlap to any ImageNet categories, and a challenging grounding setting with objects in various combinations of color, shape, size, and material. Recent work on attentional VQA systems (SNMN [13] and MAC [16]) show high VQA accuracy, making this dataset a good candidate to explore the relationship of grounding and VQA accuracy.

GQA. GQA is a real world visual reasoning dataset with multi-hop reasoning questions. GQA provides compositional questions for challenging real world images. Questions in GQA are more diverse than VQA 2.0 [9] in several ways with more coverage to relation, spatial, and compositional questions [17]. This dataset consists of 22M QA pairs for more than 113K images. GQA provides grounding labels for objects referenced in the question and answer which makes it a suitable test bed for our task. We use the balanced version of this dataset with the standard split provided by the authors for our experiments.

CLEVR-Answers for Visual Grounding. In this paper, we extend CLEVR dataset to CLEVR-Answers for visual grounding of answers. The CLEVR dataset is a synthetically rendered dataset for the evaluation of visual reasoning

Method	T	#param	Acc.	Overlap			IOU		
				P	R	F1	P	R	F1
MAC [16]	4	12.20M	**97.70**	24.92	56.27	34.55	13.99	33.50	19.73
MAC-Caps		12.92M	96.79	**47.04**	**73.06**	**57.23**	**23.97**	**39.06**	**29.71**
MAC [16]	6	12.72M	98.00	30.10	52.41	38.24	12.59	23.62	16.42
MAC-Caps		12.76M	**98.02**	**48.49**	**79.75**	**60.31**	**29.03**	**47.63**	**36.07**
MAC [16]	12	14.30M	**98.54**	28.66	53.27	37.27	8.50	18.11	11.57
MAC-Caps		15.02M	97.88	**50.90**	**94.61**	**66.19**	**27.72**	**49.84**	**35.62**
SNMN [13]	9	7.32M	96.18	52.87	67.03	59.12	37.81	47.50	42.11
SNMN-Caps		6.94M	**96.66**	**73.81**	**78.13**	**75.91**	**50.58**	**51.80**	**51.18**

Table 1. Comparison with baseline systems on CLEVR-Answers validation set. MAC-Caps and SNMN-Caps are the variants with the proposed soft masked capsules. For MAC, results are shown with varying reasoning steps, T (column 2). SNMN uses T=9. See section 6.1 for details. Numbers are reported in percentages.

and complex VQA tasks. It consists of a train set with 70K images and approximately 700K question-answer pairs and a validation set of 15K images with about 150K question-answer pairs. To allow for an evaluation of visual grounding on this task, we use the framework provided by [23], and generate new question-answer pairs with the bounding box labels for the answers as shown in figure 3. We use the same training and validation scenes (images) and generate 10 new QA pairs for each image. To get localization labels for each answer, we follow a two step process: First, we obtain the set of object ids which leads to the answer. Each question in CLEVR dataset is accompanied with a question graph, a stepwise reasoning layout with the information required to solve the question [23]. We traverse question graph in a backward direction starting from the last node and do breadth-first-search (BFS) till we traverse all nodes which are at breadth level=1. This gives us the list of objects which were used in the final reasoning step and generated an answer. Please note that not every answer will have grounding labels. For instance, if the question is *"how many blue rubber blocks are behind red cylinder?"* and the answer is 0, then there will be no bounding box labels. Second, to get bounding boxes for this set of objects, we need scene information. For each question and its corresponding answer grounding objects, we use the center pixel coordinate information (available with each scene object) to locate each object in the scene. Then, based on the object size and shape, we use a few heuristics to get a rough estimate of the bounding box around each object of interest.

This two-step process results in 901K bounding boxes (for about 700K QA pairs) for training set and 193K boxes (for about 150K QA pairs) for validation set i.e. more than 1M bounding boxes labels. Note that we do not use those bounding boxes for training, but we will provide them as well to spur further research. To have a standard train-val-test setup for our experiments, we separate 1K training images with 10K QA pairs for validation of hyper parameters. The original CLEVR validation set is used as test set and is never seen during training or validation.

Method	Grd. GT	Acc.	Overlap			IOU		
			P	R	F1	P	R	F1
MAC	Q	**57.09**	19.75	30.69	24.04	2.88	4.36	3.46
MAC-Caps		55.13	**37.77**	**63.65**	**47.41**	**5.39**	**8.65**	**6.64**
MAC	FA	**57.09**	22.43	31.35	26.15	3.30	4.48	3.80
MAC-Caps		55.13	**41.53**	**63.00**	**50.06**	**6.14**	**8.85**	**7.25**
MAC	A	**57.09**	5.61	27.36	9.31	0.92	4.46	1.52
MAC-Caps		55.13	**11.95**	**62.56**	**20.07**	**2.32**	**11.91**	**3.88**
MAC	All	**57.09**	25.01	30.48	27.47	3.66	4.28	3.95
MAC-Caps		55.13	**46.06**	**62.30**	**52.96**	**7.03**	**8.72**	**7.79**

Table 2. Results on GQA validation set for MAC with T=4. Results are based on grounding of objects referenced in the question (Q), full answer (FA), short answer (A), as well as combined grounding of question and answer (All). We consistently outperform MAC in all metrics. When evaluating for a certain grounding label type, other detected objects are treated as false positives. Numbers are reported in percentages.

6. Experiments and Results

Evaluation Metrics. To evaluate the correct answer localization (grounding), we report precision, recall, and F1-score based on two criteria: intersection over detection (Overlap), and intersection over union (IOU). Bounding boxes for the object detections are compared with the ground truth bounding boxes to evaluate how close they are to the ground truth labels in terms of overlap and IOU. Predicted regions are considered true positives if the spatial overlap of predicted bounding box and ground truth bounding box is greater than a certain threshold. The detection threshold is 0.5. The baseline systems use a multi-hop reasoning process producing attention maps for each reasoning step. Since the reasoning process is divided into sub-operations resulting in each operation producing a separate attention map, it is possible that evidence for the correct answer was attended at some intermediate step and not necessarily at the last step. To give advantage to the baseline methods, we consider the best attention map in the reasoning process with respect to F1 score.

6.1. Comparison to baseline method

We first compare the impact of the proposed capsule module on the two baseline systems, MAC and SNMN, on the CLEVR-Answers dataset as well as MAC on GQA. See tables 1, 2 and 3. We use SNMN and MAC as our baselines. These VQA systems take a question with image-based holistic features as input and generate answers with interpretable attention maps. Visual capsules module has the same number of capsules in both layers i.e., we set $C_1 = C_2 = C$ in all our experiments (C = no. of capsules).

CLEVR-Answers: We first evaluate the performance of both systems on the CLEVR-Answer benchmark. We extract 14x14x1024 dimensional features from the conv4 layer of a ResNet-101 backbone pretrained on ImageNet,

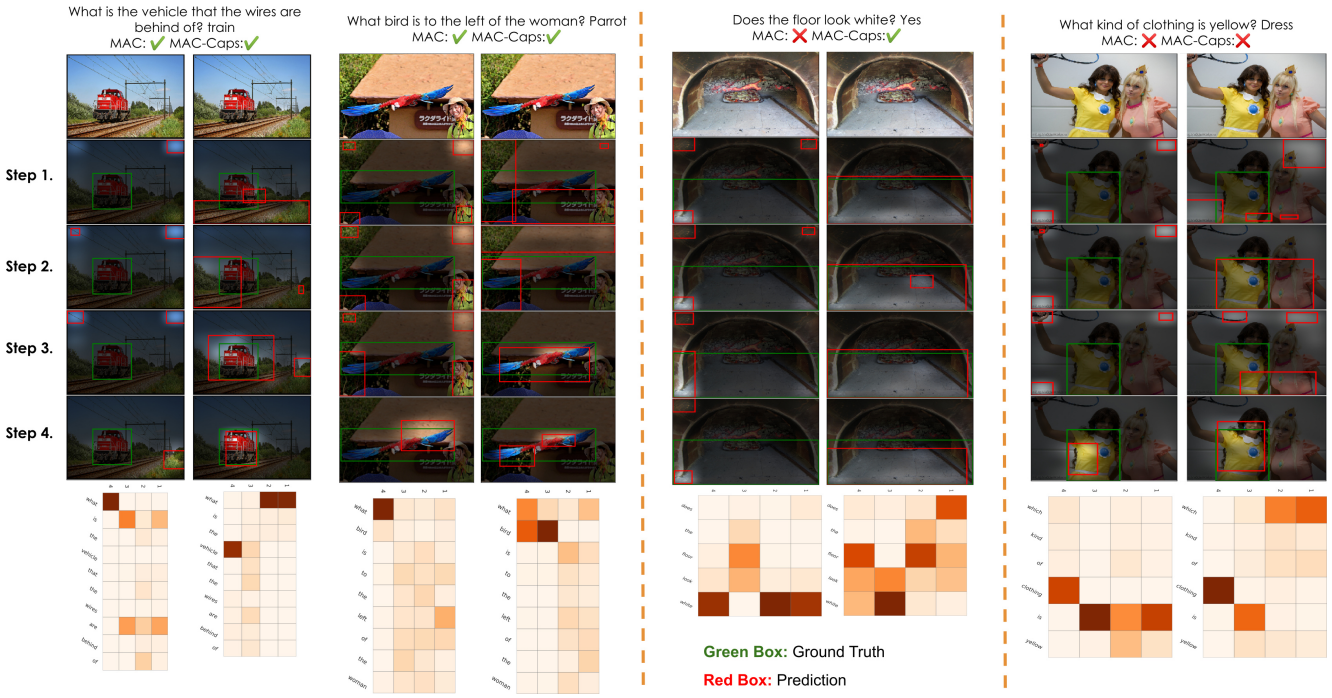

Figure 4. Attention visualizations for MAC on GQA dataset. Column 1 shows results for MAC, column 2 shows results for MAC-Caps and the same order is followed onwards. Row 1 shows input image, rows 2-5 are attention visualizations for each reasoning step (T=4) with ground truth (green boxes) and detected grounding objects (red boxes), followed by attention on question words for each step. See how MAC-Caps attends to the correct boxes. The attention on question words is also improved. Refer to section 6.3 for further details and discussion. Best viewed in color.

which are referred to as spatial features in [17]. We pass them through conv layers by MAC and SNMN to generate 14x14x512 dimensional features. We train the models for 25 epochs, using the model with best VQA accuracy for grounding evaluation. The original **MAC** baseline reports their best VQA accuracy with T=12 system [16]. However, it is recommended to use four to six reasoning steps to get interpretable attention maps. Thus, we train both, MAC and MAC-Caps for T=4, 6, and 12 (α is set to 1 for MAC). Table 1 shows the difference of both systems, the MAC and the MAC-Caps with respect to visual grounding. The MAC baseline achieves best IOU F1-score of 19.73 with T=4 whereas MAC-Caps achieves the best IOU F1-score of 36.07 (19.41% ↑) using T=6 without hurting VQA accuracy. Note that for MAC-Caps, the best Overlap F1-score is reached at T=12, which is an indicator that larger attention maps are produced, which are not rated by the Overlap measurement. Overall, we see a significant and constant increase across all evaluated scores of the proposed MAC-Caps compared to the MAC baseline. For the evaluation of **SNMN** and SNMN-Caps, we train both systems with input features as described above and choose the hyperparameters as mentioned in [13]. Note that SNMN, opposed to MAC, uses an expert layout setting, i.e, question graph layouts are used and learned during training. We get our best results with 24 capsules for SNMN-Caps as further evaluated in section 6.2. Using $\alpha = 7$ gives us best grounding results for SNMN. Overall, we see a similar increase in performance as for MAC and MAC-Caps, with an IOU F1-score of 42.11 for SNMN and an IOU F1-score of 51.18 for SNMN-Caps.

Method	Validity	Plausibility	Consistency	Distribution	Grounding
MAC	95.14	91.34	**84.90**	6.44	41.68 (30.34)
MAC-Caps	**95.17**	**91.48**	80.90	**5.67**	**45.54 (38.82)**

Table 3. Results on GQA validation set for other evaluation metrics. Grounding results are shown for attention maps from the last (mean) reasoning step(s). Numbers are reported in percentages.

Method	Acc.	Overlap			IOU		
		P	R	F1	P	R	F1
(1) masked conv.	95.69	59.71	71.13	64.92	42.28	49.24	45.49
(2) hard masking	88.48	63.24	72.76	67.67	43.35	48.29	45.69
(3) shared mask layer	95.76	70.40	76.13	73.15	48.01	49.77	48.87
(4) w/mask (C=8)	95.34	63.12	74.38	68.29	42.08	47.61	44.67
(5) w/mask (C=16)	95.79	73.72	76.27	74.97	**50.82**	50.02	50.42
(6) w/mask (C=24)	**96.66**	**73.81**	**78.13**	**75.91**	50.58	**51.80**	**51.18**

Table 4. Ablations over the design choices for the proposed architecture on CLEVR-Answers val set with SNMN as base architecture. Rows 1-3 show the influence of masking (with 16 capsules), where, masked conv.= masking of convolutional layer, hard masking=one hot masking, shared mask layer=weights for masking layer are shared among reasoning modules; w/mask=soft weights are used to mask the capsules (rows 4-6). The lower part shows the impact of number of capsules: C = no. of capsules.

GQA: To assess the performance on real world data, we evaluate our system in context of MAC on the GQA dataset.

GQA provides grounding labels for question, single word answer and sentence-based answer. We compare both setups, MAC and MAC-Caps, using the proposed grounding score based on overlap and IOU (see table 2), as well as the metrics proposed by Hudson et al. [17] where, for each question-image pair, the grounding score is the sum of attention over ground truth region(s) *r*, averaged over all data samples (see table 3). We use T=4 for both the MAC baseline and MAC-Caps, showing the best performance on this dataset. We report results on the GQA validation set.

We again observe that MAC-Caps consistently outperforms the MAC baseline on all metrics in table 2. We notice significant improvement (23.91% $\uparrow$ on F1-score) in terms of Overlap with 3.45% improvement in F1-score in terms of IOU for full answer grounding. Note that the scores, especially in context of IOU, are much lower on this dataset compared to the CLEVR benchmark, which can be attributed to the complexity of the natural images in this context. Regarding the comparison to the metrics proposed by [17] shown in table 2, we see the increase with respect to the grounding abilities of the MAC-Caps compared to the MAC baseline as well as compared to the reported spatial feature baseline of 43% in [17]. Overall, both evaluations show that the proposed capsule module allows for a better learning of visual grounding from weak VQA supervision even in a challenging real world setting given with GQA.

6.2. Ablations and Analysis

Convolutional layers vs. Capsules. To investigate how much capsules contribute compared to convolutional layers, we mask convolutional features instead of using capsules. We add a convolutional layer on top of image features resulting in $C \times (4 \times 4 + 1)$ features to keep same number of channels as in capsules (here, C=16). Similar to soft masking in capsules, these convolutional features are also masked before performing the reasoning operation. We find that masked convolutional features perform 3.38% better than the SNMN baseline in terms of IOU, but capsules still outperform them with a large margin (45.49% vs. 50.42%) for convolutional masking (see table 4 (1)=masked convolutions and (5)=baseline). This shows that query-based masking of capsules performs superior when compared to masked convolutional features .

Hard Masking vs. Soft Masking. There are two possible ways to mask capsules based on the query input. The first is masking them using softmax scores which we call soft masking; the second is keeping the capsule with highest probability and mask out the rest of the capsules (using one hot vector as mask), which we call hard masking. We find that using soft masking gives best results. When using hard masking of capsules (C=16), it hurts VQA accuracy (88.07%), although giving comparable results on grounding metrics (see table 4 (2)=hard masking and (5)=baseline). Therefore, we use soft masking for all our experiments.

Shared masking vs. separate masking. For SNMN, our final architecture uses a separate masking layer for each reasoning module. We also experiment with using a single masking layer with shared weights for all reasoning modules. While shared masking layer yields good results, we get the best grounding scores using separate masking layer (see table 4 (3)=shared mask layer and (5)=baseline).

Performance analysis w.r.t. no. of capsules. We finally analyze the system with varying number of capsules. We train the SNMN-Caps model with C=8, 16, and 24. All of them perform superior to the original SNMN in terms of grounding while achieving comparable VQA accuracy. With 24 capsules, SNMN-Caps outperforms the baseline SNMN on both VQA and grounding task (table 4 (4-6)).

6.3. Qualitative Results

Figure 4 shows qualitative analysis on GQA dataset with MAC-Caps. For samples, where both systems give the correct answer (columns 1-4), we observe that MAC often attends to corners in the image during the intermediate reasoning steps and attends to the region(s) of interest only at the final stage. For instance, on first sample (columns 1-2), MAC never attended to the correct object yet somehow produces the correct answer. MAC-Caps, on the other hand, pays attention correctly to relevant regions on earlier stages even for the case where the final answer is incorrect (columns 7-8). Additionally, MAC-Caps produces more precise attention than the baseline system. Attention on question words also seems to be improved for MAC-Caps (last row). Columns 5-6 show the case where, better grounding leads the model to predict the answer correctly.

7. Conclusion

This work proposes a novel approach for the weakly supervised grounding of VQA tasks. The proposed capsule-based module can be integrated into current VQA systems. To allow a combination of capsules with VQA based text processing, we proposed a soft masking function that further improves weakly supervised answer grounding. We show by evaluating the system on two challenging datasets, GQA and CLEVR-Answers, the impact of the proposed idea to learn a weakly supervised grounding in VQA tasks.

Acknowledgements. We thank reviewers for their helpful feedback. We also thank Hui Wu for the helpful discussions in the initial phase of this project. Aisha Urooj is supported by the ARO grant W911NF-19-1-0356 and Hilde Kuehne is supported by IARPA via DOI/IBC contract number D17PC00341 for this work. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. **Disclaimer:** The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of ARO, IARPA, DOI/IBC, or the U.S. Government.

References

[1] Asma Ben Abacha, Sadid A Hasan, Vivek V Datla, Joey Liu, Dina Demner-Fushman, and Henning Müller. Vqa-med: Overview of the medical visual question answering task at imageclef 2019. 2019. 1

[2] 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 *CVPR*, 2018. 2

[3] 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

[4] Shi Chen, Ming Jiang, Jinhui Yang, and Qi Zhao. Air: Attention with reasoning capability. *arXiv preprint arXiv:2007.14419*, 2020. 1

[5] Zhenfang Chen, Jiayuan Mao, Jiajun Wu, Kwan-Yee Kenneth Wong, Joshua B Tenenbaum, and Chuang Gan. Grounding physical concepts of objects and events through dynamic visual reasoning. *ICLR*, 2021. 2

[6] Kevin Duarte, Yogesh Rawat, and Mubarak Shah. Videocapsulenet: A simplified network for action detection. In *Advances in Neural Information Processing Systems*, pages 7610–7619, 2018. 3

[7] Kevin Duarte, Yogesh S Rawat, and Mubarak Shah. Capsulevos: Semi-supervised video object segmentation using capsule routing. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 8480–8489, 2019. 2, 4

[8] Chuang Gan, Yandong Li, Haoxiang Li, Chen Sun, and Boqing Gong. Vqs: Linking segmentations to questions and answers for supervised attention in vqa and question-focused semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1811–1820, 2017. 2

[9] 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. 5

[10] Danna Gurari, Qing Li, Abigale J Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P Bigham. Vizwiz grand challenge: Answering visual questions from blind people. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3608–3617, 2018. 1

[11] Geoffrey E Hinton, Alex Krizhevsky, and Sida D Wang. Transforming auto-encoders. In *International conference on artificial neural networks*, pages 44–51. Springer, 2011. 3

[12] Geoffrey E Hinton, Sara Sabour, and Nicholas Frosst. Matrix capsules with em routing. In *International conference on learning representations*, 2018. 3, 4

[13] Ronghang Hu, Jacob Andreas, Trevor Darrell, and Kate Saenko. Explainable neural computation via stack neural module networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018. 2, 3, 5, 6, 7

[14] Ronghang Hu, Anna Rohrbach, Trevor Darrell, and Kate Saenko. Language-conditioned graph networks for relational reasoning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 10294–10303, 2019. 2

[15] Ronghang Hu, Amanpreet Singh, Trevor Darrell, and Marcus Rohrbach. Iterative answer prediction with pointer-augmented multimodal transformers for textvqa. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[16] Drew A Hudson and Christopher D Manning. Compositional attention networks for machine reasoning. *International Conference on Learning Representations (ICLR)*, 2018. 2, 3, 5, 6, 7

[17] Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1, 2, 5, 7, 8

[18] Taewon Jeong, Youngmin Lee, and Heeyoung Kim. Ladder capsule network. In *International Conference on Machine Learning*, pages 3071–3079, 2019. 3

[19] Huaizu Jiang, Ishan Misra, Marcus Rohrbach, Erik Learned-Miller, and Xinlei Chen. In defense of grid features for visual question answering. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[20] Ming Jiang, Shi Chen, Jinhui Yang, and Qi Zhao. Fantastic answers and where to find them: Immersive question-directed visual attention. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 1

[21] Yu Jiang, Vivek Natarajan, Xinlei Chen, Marcus Rohrbach, Dhruv Batra, and Devi Parikh. Pythia v0. 1: the winning entry to the vqa challenge 2018. *arXiv preprint arXiv:1807.09956*, 2018. 2

[22] Juanzi Li Jiaxin Shi, Hanwang Zhang. Explainable and explicit visual reasoning over scene graphs. In *CVPR*, 2019. 2

[23] Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *CVPR*, 2017. 2, 6

[24] Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Judy Hoffman, Li Fei-Fei, C. Lawrence Zitnick, and Ross Girshick. Inferring and executing programs for visual reasoning. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 2

[25] Dahuin Jung, Jonghyun Lee, Jihun Yi, and Sungroh Yoon. icaps: An interpretable classifier via disentangled capsule networks. 2020. 2

[26] Aisha Urooj Khan, Amir Mazaheri, Niels da Vitoria Lobo, and Mubarak Shah. Mmft-bert: Multimodal fusion transformer with bert encodings for visual question answering, 2020. 2

[27] Satwik Kottur, José MF Moura, Devi Parikh, Dhruv Batra, and Marcus Rohrbach. Clevr-dialog: A diagnostic dataset for multi-round reasoning in visual dialog. *arXiv preprint arXiv:1903.03166*, 2019. 2

[28] Rodney LaLonde and Ulas Bagci. Capsules for object segmentation. *arXiv preprint arXiv:1804.04241*, 2018. 2, 3

[29] Jie Lei, Licheng Yu, Tamara L Berg, and Mohit Bansal. Tvqa+: Spatio-temporal grounding for video question answering. In *Tech Report, arXiv*, 2019. 2

[30] Hongyang Li, Xiaoyang Guo, Bo DaiWanli Ouyang, and Xiaogang Wang. Neural network encapsulation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 252–267, 2018. 3

[31] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 1

[32] Runtao Liu, Chenxi Liu, Yutong Bai, and Alan L Yuille. Clevr-ref+: Diagnosing visual reasoning with referring expressions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4185–4194, 2019. 2

[33] Jiayuan Mao, Chuang Gan, Pushmeet Kohli, Joshua B Tenenbaum, and Jiajun Wu. The neuro-symbolic concept learner: Interpreting scenes, words, and sentences from natural supervision. In *International Conference on Learning Representations*, 2018. 2

[34] David Mascharka, Philip Tran, Ryan Soklaski, and Arjun Majumdar. Transparency by design: Closing the gap between performance and interpretability in visual reasoning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 2

[35] Bruce McIntosh, Kevin Duarte, Yogesh S Rawat, and Mubarak Shah. Visual-textual capsule routing for text-based video segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 3

[36] Ahmed Osman, Leila Arras, and Wojciech Samek. Towards ground truth evaluation of visual explanations. *arXiv preprint arXiv:2003.07258*, 2020. 1, 2

[37] Yao Qin, Nicholas Frosst, Sara Sabour, Colin Raffel, Garrison Cottrell, and Geoffrey Hinton. Detecting and diagnosing adversarial images with class-conditional capsule reconstructions. *arXiv preprint arXiv:1907.02957*, 2019. 3

[38] Yue Qiu, Yutaka Satoh, Ryota Suzuki, Kenji Iwata, and Hirokatsu Kataoka. Multi-view visual question answering with active viewpoint selection. *Sensors*, 20(8):2281, 2020. 1

[39] Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton. Dynamic routing between capsules. In *NIPS*. 2017. 2, 3

[40] Damien Teney, Peter Anderson, Xiaodong He, and Anton Van Den Hengel. Tips and tricks for visual question answering: Learnings from the 2017 challenge. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4223–4232, 2018. 2

[41] Ramakrishna Vedantam, Karan Desai, Stefan Lee, Marcus Rohrbach, Dhruv Batra, and Devi Parikh. Probabilistic neural-symbolic models for interpretable visual question answering. *arXiv preprint arXiv:1902.07864*, 2019. 2

[42] Kexin Yi, Chuang Gan, Yunzhu Li, Pushmeet Kohli, Jiajun Wu, Antonio Torralba, and Joshua B Tenenbaum. Clevrer: Collision events for video representation and reasoning. *ICLR*, 2020. 2

[43] Kexin Yi, Jiajun Wu, Chuang Gan, Antonio Torralba, Pushmeet Kohli, and Joshua B Tenenbaum. Neural-Symbolic VQA: Disentangling Reasoning from Vision and Language Understanding. In *Advances in Neural Information Processing Systems (NIPS)*, 2018. 2

[44] D. Yu, J. Fu, T. Mei, and Y. Rui. Multi-level attention networks for visual question answering. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4187–4195, 2017. 2

[45] 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 (CVPR)*, June 2019. 2

[46] Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. From recognition to cognition: Visual commonsense reasoning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2

[47] Suofei Zhang, Quan Zhou, and Xiaofu Wu. Fast dynamic routing based on weighted kernel density estimation. In *International Symposium on Artificial Intelligence and Robotics*, pages 301–309. Springer, 2018. 3

[48] Yongheng Zhao, Tolga Birdal, Haowen Deng, and Federico Tombari. 3d point capsule networks. In *CVPR*, 2019. 2, 3