3D Question Answering

Shuquan Ye, Dongdong Chen, Songfang Han, and Jing Liao

Abstract—Visual question answering (VQA) has experienced tremendous progress in recent years. However, most efforts have only focused on 2D image question-answering tasks. In this article, we extend VQA to its 3D counterpart, 3D question answering (3DQA), which can facilitate a machine’s perception of 3D real-world scenarios. Unlike 2D image VQA, 3DQA takes the color point cloud as input and requires both appearance and 3D geometrical comprehension to answer the 3D-related questions. To this end, we propose a novel transformer-based 3DQA framework “3DQA-TR”, which consists of two encoders to exploit the appearance and geometry information, respectively. Finally, the multi-modal information about the appearance, geometry, and linguistic question can attend to each other via a 3D-linguistic Bert to predict the target answers. To verify the effectiveness of our proposed 3DQA framework, we further develop the first 3DQA dataset “ScanQA”, which builds on the ScanNet dataset and contains over 10 K question-answer pairs for 806 scenes. To the best of our knowledge, ScanQA is the first large-scale dataset with natural-language questions and free-form answers in 3D environments that is fully human-annotated. We also use several visualizations and experiments to investigate the astonishing diversity of the collected questions and the significant differences between this task from 2D VQA and 3D captioning. Extensive experiments on this dataset demonstrate the obvious superiority of our proposed 3DQA framework over state-of-the-art VQA frameworks and the effectiveness of our major designs. Our code and dataset will be made publicly available to facilitate research in this direction. The code and data are available at http://shuquanye.com/3DQA_website/.

Index Terms—Point cloud, scene understanding

1 INTRODUCTION

In recent years, we have witnessed tremendous artificial intelligence (AI) progress in vision and language understanding. Among them, visual question answering (VQA) [1], [3], [4], [5], [6], [7], which finds the correct answers to questions based on understanding images, has attracted a significant amount of research effort. In this area, a number of datasets with well-defined tasks and evaluation protocols have been introduced and various methods [1], [3], [4], [5], [6], [7] have been proposed. While existing works in VQA are restricted primarily to images, we take the first step toward extending it to the 3D question answering (3DQA) task, i.e., answering questions given a color point cloud. A well-defined 3DQA task will broaden AI’s perception to 3D spatial understanding that mimics real-world scenarios and will benefit a wide range of applications, such as robot interactions in real-world environments, information queries in augmented and virtual reality, and linguistic-based navigation of autonomous vehicles. However, extending existing VQA methods to solve 3DQA is non-trivial. Unlike VQA, which relies on 2D appearance information to answer questions, 3DQA has a significantly greater need to understand the 3D geometry. For example, answering the first question shown in Fig. 1 “Are the sofas next to the door of this room placed parallel or perpendicular to the door, or neither?” requires understanding not only the appearance, but also the geometry structure of the individual objects, and even the spatial relationships among different objects.

To address these challenges, we propose the first transformer-based 3DQA framework “3DQA-TR”. It uses two encoders to extract geometry and appearance information from the point cloud and color point cloud, respectively. Given these appearance and geometry encodings along with question embedding, a 3D-Linguistic Bert (3D-L BERT) performs both intra-modal and inter-modal fusion to predict the target answer. Specifically, in the geometry encoder, we not only consider the geometry features of individual objects, but also explicitly incorporate the coordinates and scales to the spatial embedding in order to model the spatial relationship between objects. Moreover, to provide rich appearance information for the appearance encoder, we pretrain it on a synthetic dataset tailored for appearance information extraction.

In addition to the framework, we also collect the first 3DQA dataset “ScanQA”. It builds upon the real-world indoor scene dataset ScanNet [43], which contains 1613 scans from 806 scenes. Annotators were free to change the viewpoint in the 3DQA dataset collection and ask different types of questions, such as object appearance, object geometry, spatial relationship, and their comparison. After carefully filtering and cleaning, we finally get 10,062 question-answer pairs. Along with the answers, the annotators’ confidence is also provided. Some sample questions are shown in Fig. 1.

Manuscript received 18 April 2022; revised 7 November 2022; accepted 22 November 2022. Date of publication 29 November 2022; date of current version 30 January 2024.
This work was supported by Hong Kong Research Grants Council (RGC) GRF Scheme under Grant CityU 11216122.
(Corresponding author: Jing Liao.)
Recommended for acceptance by Y. He.
This article has supplementary downloadable material available at https://doi.org/10.1109/TVCG.2022.3225327, provided by the authors.
Digital Object Identifier no. 10.1109/TVCG.2022.3225327

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
To demonstrate the superiority of our framework, we compared it with representative VQA methods answering 3DQA questions given images from ScanNet videos. The comparatively excellent performance of our framework demonstrates the necessity of including spatial information in the 3DQA task and the effectiveness of our method in exploiting both geometry and appearance information. An extensive ablation study also demonstrate the effectiveness of our designs.

In summary, our contributions are threefold:

- **3DQA task**: We introduce the novel 3DQA task, which involves both language processing and 3D scene understanding.
- **3DQA-TR framework**: We design a new transformer-based framework 3DQA-TR to solve 3DQA task. It utilizes one language tokenizer for question embedding and two encoders for extracting the appearance and geometry information, respectively, and then uses a 3D-L BERT to perform multi-modal fusion for question answering.
- **ScanQA dataset**: We took the lead in building the fully human-annotated 3D question answering dataset, ScanQA, which provides natural, free-form, and open-ended questions and answers in free-perspective 3D scans.

We will make our data collection tools, ScanQA dataset and code public to facilitate future research.

## 2 RELATED WORK

### 2.1 VQA

The VQA task, which combines the challenges of both visual and linguistic processing to answer questions about a given images, has attracted intense research efforts. Many different VQA datasets and methods have been proposed in the last few years. The most famous VQA datasets include VQAv1 [1] and the VQAv2 [3] annotated by humans, GQA [4] and CLEVR [5] with synthetic questions and answers from real-world or generated images. Based on them, several splits have been further established to study bias and language prior, such as the GQA-OOD [7] dataset of infrequent concepts, and the bias-sensitive VQA-CP [6] dataset. As for VQA methods, advances in deep learning have brought tremendous success in solving VQA tasks by utilizing multi-view [83], panoramic [86], [87], RGB-D [84], and video [87] to capture the 3D information. However, none of the methods directly captures 3D information like ours, by utilizing 3D point clouds. Generally, the VQA model [35], [40], [44], [45], [70], [89] consists of three components - an image encoder to extract visual information, a language encoder to encode questions, and a fusion module to aggregate information and classify answers. Recently, transformers [24], the de-facto standard model for language tasks, have been successfully applied to VQA as well. For example, LXMERT [54], ViLBERT [38] and VL-BERT [46], which extend the popular BERT architecture to accept joint representations of image content and natural language, have demonstrated the superiority of transformers in solving VQA tasks.

In addition to standard VQA, there are some extensions of VQA to other sub-areas, such as diagrams and document analysis [8], [9], [12], [15], video understanding [10], [11], [13], [14], multiview and viewpoint selection [23], knowledge-based question answering [14], [16], [17], visual commonsense reasoning (VCR) [18], visual dialog [19] and embodied question answering (EQA) with navigation (VLN) [21], [22]. However, these extensions are still limited to 2D images and there is no extension of VQA from 2D to 3D, for answering questions about a given 3D scene. Our work takes the first step towards it and proposes a new transformer-based framework for 3D understanding and linguistic processing.

### 2.2 3D Vision and Language

3D scene understanding with point clouds, such as segmentation [20], [65], [66], registration [67], upsampling [69], denoising [68], and generation [88] has made great progress in recent years. There are also some works that further exploring 3D scene understanding through language, such as 3D object localization (e.g., [2], [27], [28], [29], [30], [31]), 3D object captioning (e.g., [26], [27]) and relationship grounding (e.g., [71], [72]). Among them, ScanRefer [27] is a representative work for both 3D localization and
depending, but it requires the input of multi-view photos provided by the ScanNet dataset. However, this requirement greatly limits its application because 3D datasets do not always have multi-view photos along with them. In contrast, our method takes the point cloud as input. Moreover, these methods and datasets for 3D localization and captioning are restricted to objects in limited categories, which is different from our free-form and open-ended question and answering task in real scenes.

In particular, MP3D-EQA [25] proposes a novel navigation task using 2.5D RGB-D frames and language to navigate and answer templated questions in photorealistic environments. First, their task is different from our 3DQA. MP3D-EQA focuses on navigation, so it only supports a few object classes and only three types of template questions (location, color, color_room) which are all related to navigation. In contrast, our 3DQA questions contain a wider variety of objects, spatial and visual concepts. Moreover, their RGB-D inputs limit viewpoints to observing the 3D scene, while our point cloud input enables more viewpoints to be observed in the 3D scene, and with our point cloud input, it is possible to observe the scene from different angles.

1774 IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS, VOL. 30, NO. 3, MARCH 2024
Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
existence, counts, or colors of known objects. These pre-defined known objects (e.g., wall, floor, and cabinet) and scene classes (e.g., bedroom, hotel, and living room) are given on the instruction page. For example, questions such as ‘What color is the table there?’, ‘Is this room a kitchen?’ will be rejected, and questions such as ‘Is there a computer in front of the swivel chair closest to the door?’ will not. Specifically, we define several patterns of easy-to-answer questions and reject questions that fit into these patterns:

- **Existence Pattern:** Is | Are there a | any | some <KnownObjectClass> ... ? (with < 8 words)
- **Count Pattern:** How many <KnownObjectClass> is | are | in | there | this | the | ... ? (with < 9 words)
- **Color Pattern:** What color | colour is | are the <KnownObjectClass> is | are | in | there | this | the | ... ? (with < 6 words)
- **Scene Type Pattern:** Is it | this [scene | scan | room] a | an <KnownSceneClass> ... ?

To promote interesting and diverse questions, we present annotators with previously asked questions and reject duplicated ones, as illustrated in Fig. 2 (right). In addition, we encourage annotators to ask questions specific to one scene rather than generic questions. To do this, we present annotators with scene scans in the form of colored meshes, allowing annotators to freely modify the viewpoint and camera settings via rotation, moving, and zooming, as well as modifying the transparency of the mesh surfaces.

### 3.2 ScanQA Dataset

In total, the ScanQA dataset provides 10,062 question-answer pairs (an average of 12.5 question-answer pairs per scene). Further details can be found in the supplementary materials.

#### 3.2.1 Question and Answer Statistics

Fig 3 shows the length distribution of the natural-language, free-form, open-ended questions in our dataset. It can be seen that the majority of the length of the questions have between 7 and 18 words, and the average length of the questions is 12.51 which is more than twice as long as the average question length of 5.1 of the VQA dataset, one of the most representative datasets in the image VQA tasks, reflecting the high quality of our ScanQA dataset and the complexity of the 3DQA task.

In Fig. 3, we show the length distribution of the answers of our ScanQA dataset. While most of our answers are one word long, there are still 25.51% of the answers consist of two or more words. It’s much more than the VQA dataset which has only (9.65%) answers containing two or more words.

We also note that simple answers do not imply that the questions are straightforward. On the contrary, the questions usually require complex, multi-step reasoning. For example, the Yes/No answer to question 2 in Fig. 7 requires recognition of fine-grained class objects (massage chairs), and aggregation the placement information from multiple objects.

#### 3.2.2 Question Distribution Based on the First Three Words

In Fig. 4, we show the distribution of questions, grouped roughly by their first three words. In addition to questions
that require appearance information (e.g., questions that begin with ‘Is there a... colored,’ ‘Are there any... colored,’ ‘How many different colors,’ ‘What color is/are,’ ‘Is there natural light’), there are several types of questions that capture spatial concepts (e.g., ‘Is this room spacious,’ ‘Is this a narrow/large/small,’ ‘Is this a large/’), ‘What is on/hanging/next,’ ‘What shape,’ ‘Can you reach’) and placement (e.g., ‘Is the arrangement’). In addition a certain percentage of the questions shown here necessitate navigation ability (e.g., ‘Is/are there any,’ ‘How many different,’ ‘Are all,’ ‘Which is closer to’).

3.2.3 Answer Distributions Over Different Type of Questions

Fig. 5 shows the distribution of answers across different types of questions, grouped by their first words. The heights of the bars show the number of occurrences for each answer, and the colors indicate different answers. We observe that only a few types of questions can typically be answered by ‘yes’ or ‘no’. Thus the majority of question types have rich diverse answers.

3.2.4 Comparison With VQA Dataset

Compared to the previous VQA task with 2D images, our 3DQA takes a 3D scene as input and is more concerned with object geometry and spatial relationships between objects. We demonstrate the different characteristics between the VQA dataset and our ScanQA dataset based on the question types. We first categorize the questions into various types and then analyze their statistics for each dataset.

Fig. 6 illustrates the statistics by question types for our ScanQA dataset and the VQA dataset. First, the spatial concept is much more important for 3D scenes, compared to perspective projected and scale agnostic images. Thus, questions about “spatial” concepts (e.g., scale, angle, position and their comparison), “placement” (e.g., symmetrical), and “spatial comparison to average” (e.g., to estimate the size and compare it to an average one) are common in 3DQA. Second, compared to the 3D point cloud, 2D images are viewpoint specific, arising from different conventional spatial representations. For example, we commonly use words like “front,” or references like “the closer desk (to me)” to describe an object in 2D images. However, as these descriptions are viewpoint-dependent, they will cause ambiguity in a 3D scene. Instead, people tend to specify one object in 3D scenes by describing a viewpoint or doing navigation.

Third, compared to a single-view 2D image, a 3D scan reconstructed from thousands of views is less restricted by occlusion. As a result, a 3D scan generally contains much more information than one single image. In order to answer these questions specific to the 3DQA task, the network should be designed with long-term memory and the ability...
Fig. 7. Illustration of various question types in our ScanQA dataset, which includes a comprehensive variety of appearance and geometry concepts rare in 2D VQA. Question 1 requires the appearance and geometry information of a single object, and the spatial relationships between objects. Question 2 captures the placement information from the aggregation of more than two objects. Question 3 requires navigation capability and spatial comparison with an average value. Question 4 captures complex spatial placement relationships that cannot be drawn from 2D clues even with the left 2D images which include all the 2D clues in this scene. In question 5, the clues cannot be found within a single view because of occlusions, and the correct answers can be drawn only by observing the 3D scan with two or even more perspectives.

In this section, we explain the rationality of our dataset being fully human-annotated rather than template-based generated from 3D captioning in several ways. We show that our 3DQA task requires richer perception than captioning tasks on 3D scenes, as well as having significant differences in the target setup and dataset distribution.

Another way to demonstrate the difference between 3DQA and VQA tasks is to show the top unique nouns that are rare in VQA and VQA-scene datasets. In Table 1, we present the analysis group by different scene types.

### 3.2.5 Comparison With 3D Captioning Dataset

In this section, we explain the rationality of our dataset being fully human-annotated rather than template-based generated from 3D captioning in several ways. We show that our 3DQA task requires richer perception than captioning tasks on 3D scenes, as well as having significant differences in the target setup and dataset distribution.

By conducting a human study, we prove the richer perception required by 3DQA. Table 2 compares the performance of human participants who are given both questions and 3D scenes, to the performance of other human participants who are given both questions and all the captions of the corresponding scene. Notably, the human performances with captions are significantly lower than when the people are given 3D scenes, indicating that referring expressions are insufficient to solve 3DQA problems. This demonstrates that the 3D scene perception required in 3DQA is beyond what can be captured by captions and emphasizes the importance of 3D scene comprehension in 3DQA.

The target of our 3DQA also has significant differences from 3D captioning. While 3D captions are restricted to objects of limited classes as we mentioned in Section 2.2, we assume a challenging problem setup where the questions are free-form, allowing it to capture everything from small and detailed items to fine-grained class objects, and even the entire scene.

We statistically measure the dataset distribution differences in the word distributions between our dataset and the 3D captioning dataset. Here we take one of the representative 3D captioning works ScanRefer [27]. Specifically, we apply the Kolmogorov-Smirnov test on the normalized frequencies of nouns, verbs, and adjective tokens mentioned in the two datasets, following VQA [1]. The results prove that the underlying distributions of the two datasets differ significantly (p < 1e-05). In addition, we perform the Paired T-test [81] and Anderson-
Darling test [82], which also show a significant difference \( p = 1.4 \times 10^{-5} \) and \( p < 1 \times 10^{-5} \), respectively).

All of these demonstrate that 3DQA captures information that goes beyond 3D captions by a wide margin and encourages us to collect our fully human-annotated dataset rather than relying on template-based generation from 3D captioning datasets.

### 4 Method

#### 4.1 3DQA-TR Framework Overview

The overall framework of 3DQA-TR is summarized in Fig. 8. Given an input 3D scene and a corresponding question, our 3DQA-TR regresses the answer as a classification problem over all candidate answers. The candidate answers are collected from training split, considering both subject agreement and answer confidence.

In the geometry encoder, given a scene point cloud \( S \in \mathbb{R}^{N \times 3} \) (xyz without RGB), a geometry network detects \( K \) 3D object proposals and extracts the geometry feature of each object and spatial embedding for relationship modeling. In the appearance encoder, given the point cloud \( S' \in \mathbb{R}^{N \times 3} \) (xyz with RGB), an appearance network extracts color features for each detected object to support appearance-related questions. The extracted geometry and appearance information, as well as question embedding are further embedded as appearance, geometry and linguistic elements. Taking these elements as input, 3D-L BERT conducts both intra- and inter-modal interactions and predicts the answer.

#### 4.2 Geometry Encoder

We first introduce the extraction of geometry features and spatial embedding in the geometry network, as shown in Fig. 9. Here, geometry features are defined as features of each object extracted from one 3D object detection network. The pipeline of the detection network is shown in the blue region. In detail, following the common practice [32], [33], [34], we take point clouds of the entire scene \( S \in \mathbb{R}^{N \times 3} \) (xyz without RGB) as input and feed it into a point cloud geometry feature backbone network (PointNet++ [41] in our implementation), which extracts the feature representations \( F_M \) of a subset of \( M \) points with features of \( 3 + C \)-dim. Then
P object candidates are generated by initial object proposal subsampling, whose sampling index is denoted by $I_1$. The refined representations $F_P$ of $P$ points will be extracted by the stacked attentions if the detector backbone is GroupFree [34]. The final 3D bounding boxes of a maximum of $K$ detected objects are generated by 3D NMS, whose sampling index is denoted by $I_2$. In most seniors, the object number is less than $K$, and we will apply zero padding. To extract the geometry features of each object, the object indices of the above two sampling steps are used to trace back to the corresponding feature representations in the backbone model and the refined representation in the detector for each object. In detail, we obtain the object feature representations by $Idx(Idx(F_M, I_1), I_2)$ and refined object representations by $Idx(F_P, I_2)$, where $Idx(F, I)$ denotes index select operation on feature $F$ with index $I$. These are served as geometry features.

Spatial embedding is to explicitly incorporate inter-object relationship to handle a number of questions about spatial relationship in 3DQA. To this end, we utilize point locations, scale, and bound information of each detected object. In detail, we characterize each predicted bounding boxes into a 12-dimensional vector. For box $k \in [1, K]$, the vector is formed as

$$v^k = \left( \frac{x^k}{X}, \frac{y^k}{Y}, \frac{z^k}{Z}, \frac{dx^k}{X}, \frac{dy^k}{Y}, \frac{dz^k}{Z}, \frac{x_{\text{min}}^k}{X}, \frac{y_{\text{min}}^k}{Y}, \frac{z_{\text{min}}^k}{Z}, \frac{x_{\text{max}}^k}{X}, \frac{y_{\text{max}}^k}{Y}, \frac{z_{\text{max}}^k}{Z} \right) \in \mathbb{R}^{12},$$

where $x^k, y^k, z^k$ are the coordinates of the center of the box, and $dx^k, dy^k, dz^k$ are the scales of this box, and $x_{\text{min}}^k, x_{\text{max}}^k$ are the minimum and maximum $x$ values of this box. To ensure the scale invariance, they are divided by the scales of this scan in $X, Y, Z$ directions. Then, positional encoding [24] is applied to the vector.

$$\{ \text{PE}(v^k, 2i) = \sin(v^k/1000^{2i/d_{\text{model}}}) \},$$
$$\{ \text{PE}(v^k, 2i + 1) = \cos(v^k/1000^{2i/d_{\text{model}}}) \},$$

where $i \in [0, \ldots, d_{\text{model}}/2]$ and $d_{\text{model}}$ is the target embedding dimension. Thus, for each object, we obtain a high-dimensional representation $e^k = \text{PE}(v^k) \in \mathbb{R}^{12 \times d_{\text{model}}}$ for the location, scale and bounds, which is considered as the spatial embedding. The spatial embeddings of $K$ objects is formed by the set of the embeddings of all objects, $E = \{e^k\}_{k=1}^K$.

Finally, we obtain geometry features and spatial embeddings, both of which represent the geometry information of $K$ objects. Before feeding them into 3D-L BERT, they will form geometry feature embeddings of $K$ objects, by concatenation and feeding into a feed-forward network.

To extract a global geometry feature, we first apply positional encoding to the entire scene’s box vector to obtain a global spatial embedding $v = (x_c, y_c, z_c, dx, dy, dz, x_{\text{min}}, x_{\text{max}}, y_{\text{min}}, y_{\text{max}}, z_{\text{min}}, z_{\text{max}}) \in \mathbb{R}^{12}$. This is then concatenated with the global feature, which is extracted by average-pooling on the feature representations $F_M$. After applying a feed-forward network, the global geometry feature is obtained.

### 4.3 Appearance Encoder

To answer appearance-related questions, we must extract color features for each object proposal. However, 3D object detectors tend to ignore the RGB information and focus primarily on the geometry information. To mitigate this issue, we design a separate appearance encoder to capture the color information of the object proposals and pre-train this appearance encoder on color-related questions.

Given a color point cloud $S \in \mathbb{R}^{N \times 3}$, the appearance network outputs the appearance features of each object proposal. In detail, we first use a PointNet++ network [41] to extract features $F_S$ of all points. Because objects are detected in the geometry encoder via the initial object proposal subsampling of index $I_1$ and 3D NMS of index $I_2$ as described in the previous section, here we can use these indices to select the appearance features of each detected object. It is denoted by $F_{\text{app}} = \text{Idx}(\text{Idx}(F_S, I_1), I_2)$ and will form appearance features $F_{\text{app}}$ of $K$ objects. To extract global appearance features, we apply average-pooling on all point features.

To enforce the appearance extraction, we pre-train the appearance network on a synthetic question answering dataset tailored for appearance information extraction. This dataset is oriented for appearance information extraction – the question and answer pairs only concern the colors of the objects. The question generation templates are of only two
types ‘What color is the <ObjName>?’ (single object) or ‘What color are the <ObjName>?’ (multiple objects), to avoid overfitting on the language prior. In detail, to generate the questions, we use the annotated objects in ScanNet [43], except the wall, floor, and ceiling classes. To generate the answers, we first collect all points of the corresponding objects with the instance masks provided by ScanNet. Then, we find the color name of each point by selecting the closest color from the 17 named CSS 2.1 colors. Then, using all the color names for each point in this object, we vote for one to two colors. For a single object, this color is the final answer. For multiple objects, we combine the set of all the color names of the individual objects to form the final answer. The ablation experiments in the following section will demonstrate the effectiveness of the appearance encoder and the pretraining. We note that there is no overlap between the synthetic dataset and the ScanQA. The number of questions in the synthetic dataset is 3710, 982, 0 for train, validation and test splits, respectively.

4.4 3D-Linguistic BERT

The BERT [42] model has proven its ability to aggregate and align multi-modal information in visual-language tasks. Inspired by this, we propose 3D-L BERT to aggregate the multi-modal information from the appearance encoder, geometry encoder, and linguistic tokenizer, to make the final answer prediction. The 3D-L BERT input is mainly composed of three types of elements: appearance, geometry, and linguistics. Some auxiliary elements for modality separation or information summarization (classification token) are also added. As shown in Fig. 8, each element is the sum of Point Cloud Feature Embedding, Token Embedding, and Sequence Position Embedding.

Point cloud feature embedding is the geometry embedding from a point cloud or appearance embedding from a color point cloud. In detail, in geometry elements, they are the geometry feature embedding of K objects. To gain the geometry feature embedding, we concatenate the geometry features with spatial embedding and feed them into a feed-forward network. In appearance elements, they are the appearance embeddings of K objects, which are formed by feeding the appearance features into a fully connected layer. In linguistic and other auxiliary elements, they are the same embedding derived from a combination of both the appearance and geometry features of the scene. They are obtained by applying a linear layer after concatenating the global appearance and geometry features.

Token embedding in the linguistic element is the WordPiece embedding of the question following the practice in BERT. For appearance and geometry elements, special <APP> and <GEO> tokens are defined. Sequence Position embedding is to indicate the position of an element among all the elements. Specifically, the positions among the appearance elements are identical because they are not sequential, and so are geometry elements.

By adaptively integrating intra-modal information inside each element type, 3D-L BERT enables the aggregation within geometry features or appearance features, and spatial relationship modeling among the objects represented by spatial embedding. By aligning inter-modal information among the three different element types, the linguistic information can selectively attend to the geometry or appearance clues of any object without including the redundant information. In addition, a scaling parameter for each embedding and each element is jointly learned to achieve a good balance. The following ablation experiments concerning individual embedding will demonstrate the superiority of the design.

5 EXPERIMENTS

We first describe the data preparation, evaluation metric and training. Then, we show both quantitative and qualitative results of our framework compared with state-of-the-art 2D VQA approaches, and the performance of human participants under different input settings. We also conduct ablation experiments to demonstrate the effectiveness of our major components.

5.1 Data Preparation and Evaluation Metric

We split our data into training/validation/test sets following the standard split [32], [34] of ScanNet [43], and ensure disjoint scenes in each split. The number of questions is 8551, 609, 902 for train, validation and test splits, respectively. According to the type of the answers for each question, we split the test set into four sub-classes: Yes/No (Y/N), Color, Number, and Others. We use the top-1 accuracy, called the exact token match (EM), following the common practice in VQA [8], [53], [56], [73], [74], [75], [76], [77], as the evaluation metric for the following experiments, as EM was introduced as a good evaluation metric in QA tasks, especially for single-word answers. We also include the METEOR [78], [79], [80] metric, which was designed to evaluate longer phrases, as more than \( \frac{1}{5} \) of the answers in our ScanQA dataset exceed one word.

5.2 Training

The 3D-L BERT model is initialized by the official PyTorch pretrained BERT-base model as described in [42]. Before jointly fine-tuning the whole framework, the detector backbone (Group-Free [34] by default) is pre-trained on the 3D object detection task, and the appearance network is pretrained by question answering on the synthetic color-related object detection task, and the appearance network is pre-trained by question answering on the synthetic color-related questions. We train our framework on the training and validation splits and report both quantitative and qualitative results on the test split. We train the model on four NVIDIA GeForce 2080Ti GPUs with a total batch size of 64. For training, the AdamW optimizer is applied, with a base learning rate of 1e-8 for the feature extractor backbone PointNet++ and 5e-6 for the remainder, weight decay of 1e-4, learning rate warming up over the first 500 steps, and then following a cyclical learning rate policy.

5.3 Main Results

Comparison With State-of-the-Art VQA Methods. Table 3 compares our method with the state-of-the-art 2D VQA methods LXMERT [54], VILBERT [38] and 12-in-1 [39], which take images and questions as input. These Image+Q baselines are trained with questions from ScanQA and the corresponding video frames from ScanNet as input. To favor their performances, we chose the image whose viewpoint...
To show that 3D scene comprehension is required for 3DQA, we also conduct a human-study experiment that examines human performance under three different input settings: 1) only the question, 2) the question and a single image that is closest to the viewpoint and view direction of the ground-truth human answer, and 3) the question and the corresponding 3D scene. For fairness, the questions are randomly chosen, and different settings of the same scene are not assigned to the same subject. In addition, participants were not given any prior knowledge of the dataset. Table 5 shows that human participants who are given both questions and 3D scenes perform much better than the others, demonstrating the importance of 3D scene comprehension in 3DQA. We note that the human EM given both questions and 3D scenes is greater than 83%, indicating the high quality of our dataset. Notably, the performance gap (＞20.0%) between Question+Image and Question+Scene is larger for spatial-related questions (the rightmost five columns). On the other hand, the performance gap between the humans and our 3DQA-TR also shows that there is still a significant amount of room to improve.

Qualitative Comparison With VQA Methods. Fig. 10 shows some qualitative examples of 3DQA-TR, the 2D VQA baseline, and the ground-truth human answers in our test set, for questions of different types – questions about “spatial” concepts (1), “placements” (6, 8), “viewpoint and navigation” (8), and “aggregation” (2, 5). Commonsense reasoning ability is also required to answer Questions (2, 4, 6, 7) in our dataset, which further verifies the variance of our dataset and the robustness of our method to different question types. In all the questions of all types, our predictions are scored 100% correct, demonstrating the capability of the proposed framework for both appearance and geometry comprehension. Specifically, Question 8 shows popular scenarios where VQA methods fail for specifying viewpoint or doing navigation. In addition, Question 1, which requires a comparison of spatial information, namely, the sizes of two beds, is a common scenario in ScanQA. However, we can see all the state-of-the-art 2D VQA baselines end up with wrong predictions. This result is expectable because of the perspective projecting nature of 2D images, which renders the size of the white one larger than the blue one. In contrast, the prediction of our model is correct. We attribute it to the superiority of our geometry encoder over the 2D-based baselines, as our encoder explicitly extracts and incorporates geometry embeddings of the objects’ spatial information in 3D scans.

Comparison With Related Question Answering Methods. To further illustrate the advantage of question answering with the point cloud and the effectiveness of our proposed method, in the first eight rows of Table 4, we compare the results of four groups of related question answering methods: the state-of-the-art frameworks with inputs of multi-view images, panoramic images, panoramic video, and spatial reasoning VQA. In each group, the second lines are the performance of the state-of-the-art frameworks with each input setting, respectively, and the first lines are their backbone VQA models. The metric is EM. Our 3DQA-TR outperforms related question answering works by a large margin, especially in spatial questions (an improvement of more than 12.28%) and aggregation (an improvement of more than 12.5%). Although the other methods had performance gains compared with their backbones, they still fail to solve this 3DQA task as effectively as our framework. These results confirmed the effectiveness of the proposed framework, which we attribute to its 3D perception ability and the advantage of the point cloud, which directly captures more direct access to rich, accurate 3D spatial information.

Rows 9 to 11 of Table 4 show the results of adapting the Image+Q baselines for 3D+Q, by substituting the object detection backbone with a 3D counterpart. Although the state-of-the-art methods’ performance with the 3D detector is better than their performance without the 3D detector in the paper, they still fail to outperform our method, demonstrating that these methods are incapable of solving 3DQA. These results further demonstrate the effectiveness of the proposed framework, and we owe it to the superiority of the design of the framework with three different element types: appearance, geometry and language elements. These elements are specifically created for solving 3DQA task, especially with the extraction of appearance features from...
Fig. 10. Example predictions from 2D VQA baselines and our 3DQA-TR, compared to human answers in ScanQA. The answers with 100% EM scores are in green and the others are in red. ‘<unk>’ is the unknown tokens that represent the prediction is not in the candidate answers, following the official implementations [38], [39], [54].

TABLE 4
Additional Comparison With Various Possible Question Answering Methods

| Question | BUTD [35] | multi-view images [85] | MLB [89] | panoramic image [86] | LXMER [54] | spatial reasoning [84] | BERT [42] | panoramic video [87] |
|----------|------------|------------------------|---------|---------------------|-------------|------------------------|---------|---------------------|
|          | All        | Number                 | Color   | Y/N                 | Other       | aggregation            | placement | spatial             | viewpoint |
| 1. Which of the two beds is larger, the blue one or the white one? | 27.27 | 11.83 | 11.32 | 52.39 | 8.42 | 20.50 | 22.22 | 21.05 | 18.58 |
| 2. What is the name of the instrument in this scene? | 28.82 | 15.05 | 13.21 | 53.45 | 10.0 | 25.50 | 25.00 | 25.00 | 19.47 |
| 3. Is there a poster that is not hung on the same wall with the others? | 28.93 | 15.05 | 13.21 | 52.93 | 10.26 | 26.50 | 23.61 | 23.39 | 18.58 |
| 4. Is this room the attic or the basement of the house? | 29.81 | 16.13 | 18.87 | 52.93 | 11.84 | 27.50 | 25.00 | 24.56 | 18.58 |
| 5. Is the dining table the tallest table in the room? | 31.15 | 17.20 | 20.75 | 56.12 | 11.32 | 27.00 | 29.17 | 28.07 | 22.12 |
| 6. What is the material of the vintage cabinet’s transparent upper part, which is next to the dining table? | 31.82 | 19.35 | 20.75 | 56.65 | 11.84 | 27.00 | 30.56 | 23.93 | 18.58 |
| 7. What kind of pets do they have? | 30.26 | 13.98 | 20.75 | 55.86 | 10.26 | 22.50 | 27.78 | 23.98 | 19.47 |
| 8. Is the cabinet with a basket on it to your left or right as you sit on the sofa with a lamp next to it and face the TV? | 30.26 | 13.98 | 20.75 | 55.86 | 10.26 | 22.50 | 27.78 | 23.98 | 19.47 |

The first 8 rows show related question answering methods divided into 4 groups based on their input settings: multi-view image, panoramic image, panoramic video, and spatial reasoning VQA. In each group, the second line is the performance of the state-of-the-art framework respectively, and the first line is of their backbone VQA model. The 9-11 rows provide the performances of adapting the state-of-the-art image+q baselines for 3D+Q, by substituting their object detection backbone with 3D counterpart.
the pre-trained appearance encoder, the extraction of geometry features and spatial embeddings, and the intra- and inter-fusion among the three modalities.

5.4 Ablation Experiments

5.4.1 Component Validation

In the first three rows of Table 6, we show the ablation with components removed from our framework. In the “Qonly” row, both the appearance and geometry encoders, as well as the appearance and geometry elements of 3D-L BERT are removed. Thus the model works like the original BERT model and provides a lower bound for solving 3DQA with only questions as input. It has the worst overall performance of all the baselines, especially compared with the full framework (from 42.35% to 27.94%). This indicates that information beyond the question is required, and it demonstrates the reasoning ability of our framework rather than relying on language bias. We also noticed an interesting phenomenon: the overall performance is slightly better than that of the humans in the 5. It is unsurprising because the network can learn the prior knowledge of the data distribution while the tested people are not the annotators of our dataset and have no such prior.

On the “Geo+Q” row, the appearance element is removed from the input. We can see an overall performance decline compared to the full framework, especially in the “color” split (from 35.85% to 13.20%), which further demonstrates the need for appearance information for 3D question answering. However, the performance is clearly better than with the “Qonly” setting, especially in all the spatial-related questions, such as “spatial” (23.39% to 38.01%), “placement” (25.00% to 40.28%), “viewpoint and navigation” (18.58% to 33.63%), and “aggregation” (26.00% to 37.50%) subsets, which we owe to the superiority of the geometry encoder.

On the “App+Q” row, we removed the geometry element from the input. The performance, especially for the spatial-related questions, such as the “spatial” and “placement” subsets, is worse than the full model, indicating that the geometry information really matters. However, the performance gain in the “color” split (11.32% to 30.19%) is noticeable, demonstrating the capability of the appearance encoder.

5.4.2 Appearance Questions Pre-Training

In this ablation study, we demonstrate the design of the proposed pre-training of the appearance encoder in Table 6. The performance with the appearance encoder trained from scratch is shown in the “AppFromScratch” row. In contrast, for our default design, the appearance encoder is pretrained on a generated question-answering dataset. On this generated

---

**TABLE 5**

| Input             | All | Number | Color | Y/N | Other | aggregation | placement | spatial | viewpoint |
|-------------------|-----|--------|-------|-----|-------|-------------|-----------|---------|-----------|
| Question          | 27.28 | 13.98 | 15.09 | 51.86 | 7.89 | 16.00 | 23.61 | 26.32 | 23.01 |
| Question+Image    | 53.54 | 47.31 | 49.06 | 62.50 | 46.84 | 46.00 | 48.61 | 49.12 | 50.44 |
| Question+Scene    | 83.26 | 79.57 | 79.25 | 94.15 | 73.95 | 76.50 | 81.94 | 82.46 | 82.30 |

**TABLE 6**

| EM                | All | Number | Color | Y/N | Other | aggregation | placement | spatial | viewpoint |
|-------------------|-----|--------|-------|-----|-------|-------------|-----------|---------|-----------|
| Qonly             | 27.94 | 16.13 | 11.32 | 52.13 | 9.21 | 26.00 | 25.00 | 23.39 | 18.58 |
| Geo+Q             | 40.58 | 37.63 | 13.20 | 64.89 | 21.05 | 37.50 | 40.28 | 38.01 | 33.63 |
| App+Q             | 33.59 | 29.03 | 30.19 | 54.79 | 14.21 | 33.50 | 25.00 | 26.32 | 21.24 |
| NoSpaEmbedding    | 38.02 | 33.33 | 32.08 | 60.64 | 17.63 | 36.00 | 31.94 | 28.65 | 24.78 |
| OneElementForAll  | 38.91 | 35.48 | 33.96 | 60.10 | 19.47 | 35.00 | 41.67 | 35.67 | 30.09 |
| AppFromScratch    | 40.80 | 37.63 | 15.09 | 64.63 | 21.58 | 38.50 | 38.89 | 39.77 | 30.97 |
| 3DQA-TR (backbone B) | 41.79 | 41.94 | 39.62 | 63.30 | 20.79 | 39.00 | 40.28 | 39.18 | 30.97 |
| 3DQA-TR           | 42.35 | 40.86 | 35.85 | 64.63 | 21.58 | 40.00 | 41.67 | 40.35 | 31.86 |

**METEOR**

| EM                | All | Number | Color | Y/N | Other | aggregation | placement | spatial | viewpoint |
|-------------------|-----|--------|-------|-----|-------|-------------|-----------|---------|-----------|
| Qonly             | 17.54 | 16.13 | 11.32 | 63.01 | 8.83 | 18.43 | 16.16 | 15.19 | 11.44 |
| Geo+Q             | 24.45 | 37.63 | 13.21 | 72.56 | 15.60 | 25.16 | 36.44 | 22.02 | 20.94 |
| App+Q             | 20.60 | 29.03 | 30.02 | 61.99 | 11.60 | 22.71 | 22.08 | 15.56 | 13.97 |
| NoSpaEmbedding    | 23.35 | 33.33 | 31.89 | 66.81 | 13.74 | 24.14 | 27.78 | 17.53 | 15.79 |
| OneElementForAll  | 23.72 | 35.48 | 33.77 | 68.12 | 14.66 | 23.55 | 36.27 | 20.80 | 19.57 |
| AppFromScratch    | 24.52 | 37.63 | 15.09 | 71.93 | 15.60 | 25.64 | 35.19 | 23.68 | 19.37 |
| 3DQA-TR (backbone B) | 25.35 | 41.94 | 39.40 | 71.65 | 15.03 | 25.83 | 35.86 | 22.84 | 19.70 |
| 3DQA-TR           | 25.71 | 40.86 | 35.65 | 73.49 | 15.87 | 26.50 | 36.80 | 23.49 | 20.44 |

The evaluation metric for the first eight rows is EM, and the metric for the last eight rows is METEOR.
dataset, our model’s accuracy is 45.03%, which is much higher than 25.74%, the lower bound of predicting the most frequent answer to the question (e.g., “What color is the wall?”-“White”). After applying the pretraining weight and then jointly-training on the ScanQA dataset (“3DQA-TR” row), the performances on all splits are better than on the counterpart trained from scratch (EM improved from 40.80% to 42.35%), especially in the “color” split (EM improved from 15.09% to 35.85%).

5.4.3 Element Design
We show the experimental results to demonstrate that embedding the geometry and appearance into one element will cause a redundancy problem. In the “OneElementForAll” row of Table 6, we embedded the geometry feature, spatial embedding, and appearance feature into one element by concatenation, rather than embedding them into two separate elements. We can see a clear performance decline in the “OneElementForAll” setting for all splits, for both “Color” questions (EM from 35.85% to 33.96%) and spatial-related questions (e.g., EM of “spatial” from 40.35% to 35.67%). This demonstrates the need to separate geometry and appearance elements in our design.

5.4.4 Spatial Embedding
Here we conduct one ablation on the geometry encoder to remove the spatial embedding, leaving only the geometry features. The results are shown in the “NoSpaEmbedding” row of Table 6. The overall performance of both metrics are worse than that of the full framework (EM from 42.35% to 38.02%, METEOR from 25.71% to 23.35%). Specifically, there is a significant decline in the EM for spatial-related questions (“spatial” (40.35% to 28.65%), “placement” (41.67% to 31.94%), and “viewpoint” (31.86% to 24.78%). In addition, the drop in the METEOR performance in these questions can clearly be seen. The performance gaps in the spatial-related questions indicate the importance of modeling the spatial information in bounding boxes and the relationships among them via spatial embeddings.

5.4.5 Different Detector
To demonstrate the generalizability of our 3DQA-TR framework, we replace the default detector Group-Free [34] with another backbone network Vote [32], denoted as 3DQA-TR (backbone B). As shown in Table 6, the overall performances and the performance in each subclass of 3DQA-TR (backbone B) are comparable with 3DQA-TR with the default backbone Group-Free using both metrics, demonstrating the generalizability of the proposed framework to different detectors.

6 Conclusion
This paper extends the 2D VQA task into its 3D counterpart, the 3DQA task. To answer questions about a real-world 3D scene, 3DQA must understand both appearance and 3D geometry. To this end, we propose a novel end-to-end 3DQA framework 3DQA-TR by designing a multi-modal 3D-Linguistic BERT. The appearance and geometry information are encoded by two separate encoders, and then fed into the BERT together with the question embedding to predict the final answers. To support the 3DQA task, we also develop a new dataset, “ScanQA” for this task, which contains 10,062 questions and answers from 806 scenes. Extensive experiments and analyses have demonstrated the superiority of our 3DQA-TR over existing VQA baselines.

7 Limitations
Though we have tried our best to increase the diversity of the newly built ScanQA dataset, it is still not as diverse as existing large-scale image VQA datasets. This is mainly because ScanQA is built upon the ScanNet dataset, which only contains 806 indoor scenes. The limited diversity of the dataset will impair the generalizability and robustness of the model. For example, our model tends to give incorrect answers to questions about humans, as humans are scarce in this dataset. To address this problem, we are currently investigating large-scale 3DQA data generation and developing a multi-modality data augmentation technique for 3DQA. Another possible solution is to exploit extra large-scale 2D VQA data by applying knowledge distillation techniques. In addition, as discussed in Section 5.3, in our ScanQA dataset, common sense regarding entity names and reasoning ability is necessary, but it is still unclear how far the model can go beyond perception and recognition.

Acknowledgments
We would like to thank the reviewers for their constructive feedback.

References
[1] S. Antol et al., “VQA: Visual question answering,” in Proc. IEEE Int. Conf. Comput. Vis., 2015, pp. 2425–2433.
[2] P. Achlioptas, A. Abdelreheem, F. Xia, M. Elhoseiny, and L. Guibas, “Referit3D: Neural listeners for fine-grained 3D object identification in real-world scenes,” in Proc. Eur. Conf. Comput. Vis., 2020, pp. 422–440.
[3] Y. Goyal, T. Khot, D. Summers, D. StayBatra, and D. Parikh, “Making the V in VQA matter: Elevating the role of image understanding in visual question answering,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2017, pp. 6325–6334.
[4] D. Hudson and C. Manning, “CQA: A new dataset for real-world visual reasoning and compositional question answering,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2019, pp. 6693–6702.
[5] J. Johnson, B. Hariharan, L. van der Maaten, L. Fei-Fei, C. Zitnick, and R. Girshick, “CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2017, pp. 1988–1997.
[6] A. Agrawal, D. Batra, D. Parikh, and A. Kembhavi, “Don’t just assume; look and answer: Overcoming priors for visual question answering,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2020, pp. 4971–4980.
[7] C. Kervadec, G. Antipov, M. Baccouche, and C. Wolf, “Roses are red, violets are blue... but should VQA expect them to?,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2021, pp. 2775–2784.
[8] K. Kalbe, B. Price, S. Cohen, and C. Kanan, “DQVA: Understanding data visualizations via question answering,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 5648–5656.
[9] S. Ebrahimi Kahou, V. Michalski, A. Akbarpour, A. Kadar, A. Trischler, and Y. Bengio, “FigureQA: An annotated figure dataset for visual reasoning,” 2017, arXiv:1710.07300.
[10] J. Lei, L. Yu, M. Bansal, and T. L. Berg, “TVQA: Localized, Compositional Video Question Answering,” in Proc. Conf. Empirical Methods Natural Lang. Process., 2018, pp. 1369–1379.
[11] J. Lei, L. Yu, T. L. Berg, and M. Bansal, “TVQA: Spatio-temporal grounding for video question answering,” 2019, arXiv:1904.11574.
[12] R. Chaudhry, S. Shekhar, U. Gupta, P. Maneriker, P. Bansal, and A. Joshi, “LEAF-QA: Locate, encode & attend for figure question answering,” in Proc. IEEE Winter Conf. Appl. Comput. Vis., 2020, pp. 3501–3510.

[13] A. Zadeh, M. Chan, P. P. Liang, E. Tong, and L.-P. Morency, “Social-QA: A question answering benchmark for artificial social intelligence,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2019, pp. 8807–8817.

[14] N. Garcia, M. Otani, C. Chu, and Y. Nakashima, “Knowit VQA: Answering knowledge-based questions about videos,” in Proc. AAAI Conf. Artif. Intell., 2020, pp. 10 826–10 834.

[15] M. Mathew, D. Karatzas, and C. Jawahar, “DocVQA: A dataset for VQA on document images,” in Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis., 2020, pp. 2190–2199.

[16] P. Wang, Q. Wu, C. Shen, A. Dick, and A. Van Den Hengel, “FVQA: Fact-based visual question answering,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 40, no. 10, pp. 2413–2427, Oct. 2018.

[17] K. Marino, M. Rastegari, A. Farhadi, and R. Mottaghi, “OK-VQA: A visual question answering benchmark requiring external knowledge,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2019, pp. 326–335.

[18] S. Ye, D. Chen, S. Han, and J. Liao, “Learning with noisy labels for robust point cloud segmentation,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2019, pp. 6443–6452.

[19] P. Anderson et al., “Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 3674–3683.

[20] A. Das, S. Datta, G. Gkioxari, S. Lee, D. Parikh, and D. Batra, “Embodied question answering,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 1–10.

[21] Y. Frigla, K. Iwata, and H. Katoaka, “Multi-view visual question answering with active viewpoint selection,” Sensors, vol. 20, no. 8, 2020. [Online]. Available: https://www.mdpi.com/1424-8220/20/8/2281.

[22] A. Vaswani et al., “Attention is all you need,” in Proc. Adv. Neural Inf. Process. Syst., 2017, pp. 5998–6008.

[23] W. Su et al., “VL-BERT: Pre-training of generic visual-linguistic representations,” in Proc. Int. Conf. Learn. Representations, 2020. [Online]. Available: https://openreview.net/forum?id=SkgXpAeyVh.

[24] Z. Lan, M. Chen, S. Goodman, K. Gimel, P. Sharma, and R. Sorciut, “ALBERT: A lite BERT for self-supervised learning of language representations,” in Proc. Int. Conf. Learn. Representations, 2020.

[25] A. Agrawal, D. Batra, and D. Parikh, “Analyzing the behavior of visual question answering models,” 2016, arXiv:1606.07356.

[26] Y. Goyal, T. Khot, D. Summers-Stay, D. Batra, and D. Parikh, “Making the V in VQA matter: Elevating the role of image understanding in visual question answering,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2017, pp. 6904–6913.

[27] H. Gao, J. Mao, J. Zhou, Z. Huang, L. Wang, and W. Xu, “Are you talking to a machine? Concept representations for multi-modal question answering,” in Proc. 28th Int. Conf. Neural Inf. Process. Syst., 2015, pp. 2296–2304.

[28] R. Krishna et al., “Visual genome: Connecting language and vision using crowdsourced dense image annotations,” Int. J. Comput. Vis., vol. 123, no. 1, pp. 32–73, 2017.

[29] M. Tapaswi, Y. Zhu, R. Stiefelhagen, H. Torralba, R. Urtasun, and S. Fidler, “MovieQA: Grounded question answering in movies through question-answering,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2016, pp. 4631–4640.

[30] M. Ren, R. Kiros, and R. Zemel, “Exploring models and data for image question answering,” in Proc. Adv. Neural Inf. Process. Syst., 2015, pp. 2953–2961.

[31] H. Tan and M. Bansal, “LXMERT: Learning cross-modality encoder representations from transformers,” in Proc. Empir. Methods Natural Lang. Proc., 2019, pp. 1501–1514.

[32] L. Yu, E. Park, A. C. Berg, and T. L. Berg, “Visual madlibs: Fill in the blank image generation and question answering,” 2015, arXiv:1506.00278.

[33] Y. Zhu, O. Groth, M. Bernstein, and L. Fei-Fei, “Visual7W: Grounded question answering in images,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2016, pp. 4945–5004.

[34] J. Johnson, B. Hariharan, L. Van Der Maaten, L. Fei-Fei, C. Lawrence Zitnick, and R. Girshick, “CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2017, pp. 2901–2910.

[35] A. Agrawal, D. Batra, D. Parikh, and A. Kembhavi, “Don’t just assume; look and answer: Overcoming priors for visual question answering,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 4971–4980.

[36] P. Zhang, Y. Goyal, D. Summers-Stay, D. Batra, and D. Parikh, “Yin and yang: Balancing and answering binary visual questions,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2016, pp. 5014–5022.
