Auto QA : The Question Is Not Only What, but Also Where

Sumit Kumar
IIT Kanpur
ksumit@iitk.ac.in

Badri N. Patro *
IIT Kanpur
badri@iitk.ac.in

Vinay P. Namboodiri
University of Bath
vpn22@bath.ac.uk

Abstract

Visual Question Answering can be a functionally relevant task if purposed as such. In this paper, we aim to investigate and evaluate its efficacy in terms of localization-based question answering. We do this specifically in the context of autonomous driving where this functionality is important. To achieve our aim, we provide a new dataset, Auto-QA. Our new dataset is built over the Argoverse dataset and provides a truly multi-modal setting with seven views per frame and point-cloud LIDAR data being available for answering a localization-based question. We contribute localized attention adaptations of most popular VQA baselines and evaluate them on this task. We also provide joint point-cloud and image-based baselines that perform well on this task. An additional evaluation that we perform is to analyse whether the attention module is accurate or not for the image-based VQA baselines. To summarize, through this work we thoroughly analyze the localization abilities through visual question answering for autonomous driving and provide a new benchmark task for the same. Our best joint baseline model achieves a useful 74.8% accuracy on this task. We release our dataset and source code for our baseline modules in the following webpage: `https://delta-lab-iitk.github.io/AUTO-QA/`

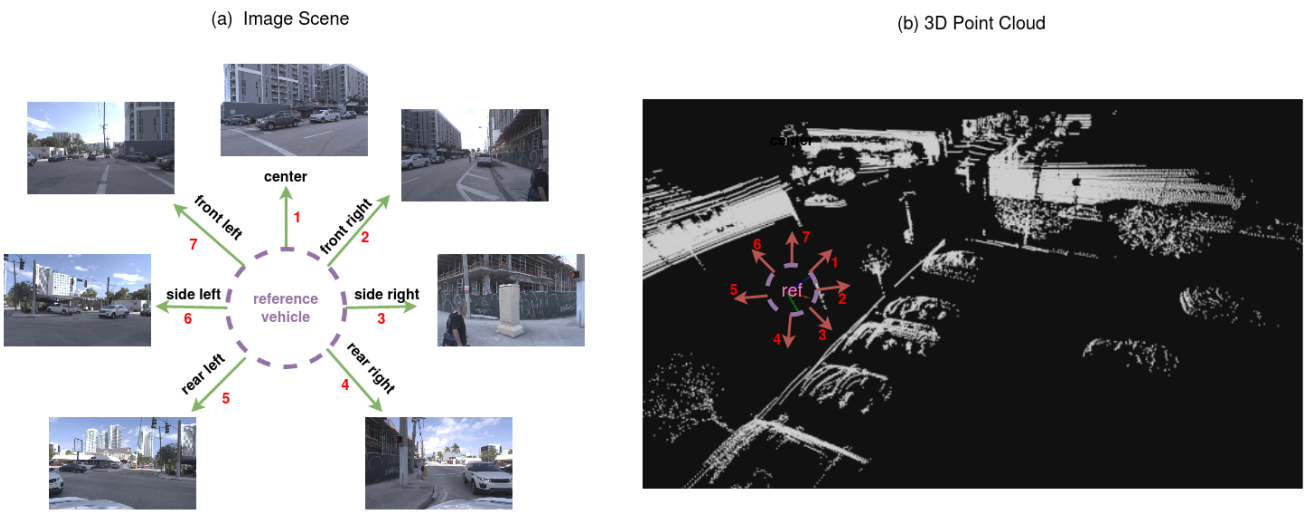
1. Introduction

The task of answering questions given an image, i.e. 'Visual Question Answering' [2] is an interesting task as it analyses the ability of an AI agent to answer a wide variety of questions in diverse scenarios. This task was meant to analyse the generalization ability of an AI agent and we observe that there has been substantial progress made in solving this task. In this work, there are two specific aspects in which we aim to extend this ability: a) Can we have a multi-modal input set comprising of several image views and LIDAR point-cloud data as input and use all the information to answer a question? b) Can we address specific functionally important task of being able to answer questions that relate to localization ability that is particularly important in tasks such as autonomous driving? Both these aspects are crucial if we are interested in broadening the applicability of visual question answering task to practically relevant scenarios. These would also serve as a semantic benchmark that would be required by AI agents in order to convince us about their ability to autonomously drive cars. If an AI agent fails to answer whether a pedestrian is present to its left, we could hardly expect to trust its ability to drive safely. Motivated by this aim, we embark on a novel visual question answering challenge that we term 'Auto-QA'.

Each question involves answering a particular location-based question such as 'How many vehicles are on my front left?'. This question should be answered by referring to the input data available to the agent. We provide as input, the seven views around the vehicle, the LIDAR point-cloud data along with the question as input to the agent. This data can be provided as it is built over the 'Argoverse' dataset [5] and for each frame we have access to all this information. The setting is thus an interesting multi-modal challenge. The automated system needs to understand the question provided as text input, analyse multiple images and identify the images most related to answering the question, comprehend the LIDAR data that is available as a point-cloud and then correctly answer the question. This task thus comprehensively extends previous visual question answering tasks that were mainly based on answering questions given a single image or in some cases a video. The dataset is sufficiently large scale with a total of 31259 scenes and close to 300K questions. There are around 15 different kinds of objects present and there are around 4000 unique questions possible. Thus, though the setting is specific, there is enough diversity in terms of the questions. We also ensure that the question distribution is not biased and provide analysis about the same.

In order to solve this challenging task, the existing baseline image-based VQA models need to be adapted and be able to attend to the appropriate image set relevant for answering the question. We do so and provide these as baselines for this task. We observe that a baseline MUTAN [4] and MLB [30] method obtain accuracy of 68.2% and 62.8% accuracy respectively. These are the current state-of-the art

*Currently working at KU Leuven



Question	Question_type	Image_to_attend	Answer
Is there a *pedestrian* to my *side right*	Exist	3	True
What is the *count* of *vehicle* on my *rear right*?	Count	4	1
What is the nearby object to my *side right*?	Closest	3	Pedestrian
Is there any *on road obstacle* to my *front*?	Exist	1	False

Figure 1. An instance of Autonomous Question Answering **(a) Image Scene** which consists of seven images corresponding to seven different directions , **(b) 3D-Point Cloud** from lidar sensor for corresponding image-scene and different variety of generated questions

models for VQA task that do not require object-detection inputs and can be efficiently implemented. This is relevant as we would require any practical VQA approach for this task to also be efficient. A joint multi-modal model comprising of adapted attention and with point-cloud data achieves a much higher accuracy of 74.8% accuracy. We note that these are just the baseline models that we provide through this work. Further evaluation by the community can provide improved models that obtain higher accuracy and are efficient in inference.

To summarize, through this paper, we provide a novel semantic task that evaluates the functional capabilities of a VQA system to truly understand localization aspects accurately. This serves as an important semantic analysis task for autonomous driving setting. Moreover, we provide an appropriate 'Auto-QA' dataset for solving this task and suitable multi-modal baselines and the analysis for the same.

2. Related work

VQA Task: Answering Question based on the and image is a challenging task in vision and language domain. Image based question answering task is first started by [16] based on very small dataset (DAQUAR) and mainly focused on indoor scene.[22] have proposed a large dataset COCO-QA. This dataset uses MS-COCO images and its caption to answer the visual questions. The main disadvantage of this dataset is that the Question and answer are synthetic data, means it is not annotated by human annotator.[2] has proposed a larger version of this dataset known as Visual Question answering (VQA-v1 and v2) dataset. They used images from MS-CCOCO and per each Image there are three question and answer are annotated by the human annotator.

Methods: To solve VQA task, basic method is to combining both visual feature with question feature and predict the answer[22, 2, 17]. It has shown that attention plays a major role for solving visual question answering task. *Grid Based Attention:* [28] proposed a Stack Attention Network (SAN) for searching various regions on the image by stacking attention modules. [15] proposed a hierarchical attention method based on question words, phrases and sentences. [26] explained about question guided attention method for visual question answering. [8] has proposed a multi-model compact bi-linear pooling (MCB) based attention method for combining image and question for answer prediction. [27] has proposed a method Dual Attention Network (DAN) to attend both image features, and also to question features. [12] has proposed a, Multimodal Low-Rank Bilinear Attention Networks (MLB), low rank bilinear polling method using Hadamard product for a efficient attention mechanism for multimodel learning task. [30] has proposed Multi-modal Factorized Bilinear Pooling (MFB) method to combine multi-modal features with more efficiently and effectively. [4] has proposed a, Multimodal Tucker Fusion (MUTAN), tensor-based Tucker decomposition method to capture efficiently bilinear interactions between image and question representation. [18] has proposed an exemplar based method to improve attention

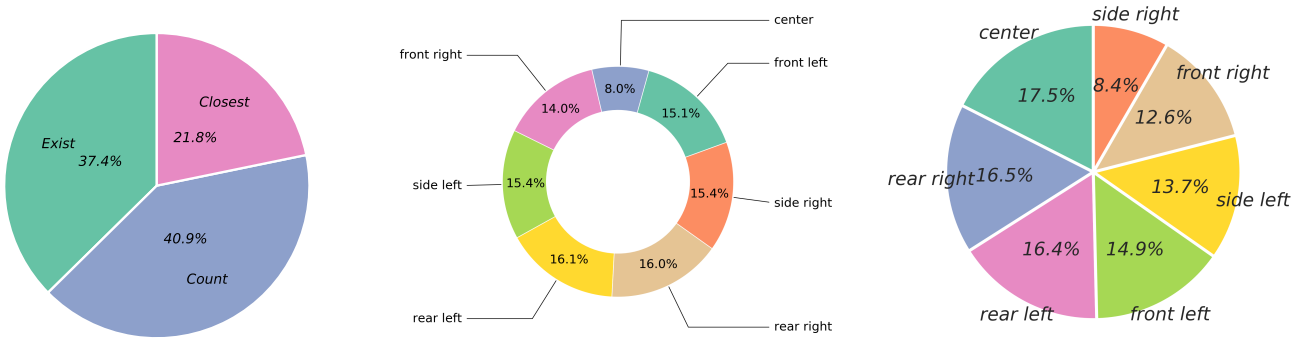


Figure 2. Question Distribution a) **over different question family**, b) **over seven directions around reference** and c) Distribution of different object (see table 1) present in dataset over different direction

on VQA task. [25, 3] have proposed a pixel based attention methods to solve visual questtion answering task. [1, 11] has proposed attention method based on Top-down and bottom-up object feature based attention for VQA task. Considering seven images per question in our dataset, size of pretrained bottom up feature was too large for training (around 750 GB) even for fixed no of features per image. Thus we limit our approach to grid features only and subsequently current state of art model using these bottom up features are not adapted. [29] has proposed a self attention based modular co-attention network for solving VQA task.

Point cloud Use of Lidar Point cloud in autonomous driving has brought rapid development in this field. PointNet [20] PointNet++ [21] used 3D CNN to extract features from 3D point cloud data for classification and segmentation. Then there has been for object detection methods in 3D point-cloud. These methods either used 3D CNN or project lidar point cloud in image [14]. [31] proposes partitioning of point into voxels followed by pointnet and 3D CNN for object detections. There has been multistage network involving proposal generation [19, 24, 6, 23]. [13] proposed detection method without using expensive 3D CNN. As per our knowledge, none of these datasets provide information on Autonomous based question answering. We proposed a multi-model reasoning dataset based on tracking information. We have multiple image which describe environment around you at a particular position also it provide point cloud information around that position. We ask a question by considering all surrounding images and point cloud position and the model will do reasoning on them and provide answer of the visual question.

3. AutoQA Dataset

The main contribution of this work is the new task of answering questions that are relevant in the autonomous driving setting. We thus propose an Autonomous Question Answering (AutoQA) dataset for analysing the ability of an autonomous agent to obtain correct prediction based on the query question in the visual scene. This task helps us in analysing the capabilities of such autonomous agents in the driving setting they are meant to work in. Primarily, AutoQA dataset consists of image scenes, question, question type and and the ground truth answer. This dataset also contains an important requirement that the agent needs to decide which image it needs to attend to while answering the question with respect to the different image scenes. The main components of the dataset are as follows: {*Image scenes (7 scenes), 3D Cloud points, Question, Question Type, Image_to_Attend, Answer*}. The image scenes consist of 7 scenes such as {*center, front right, side right, rear right, front left, side left and rear left*} as shown in figure-1. We build over the Argoverse 3D tracking dataset [5] for 3D cloud points and scenes. Argoverse 3D tracking dataset consists of 113 log segment(videos). Length of each log segment varies between 15 to 30 sec. Log segments are collected in two different cities (Miami and Pittsburgh). These segments are collected in different seasons, weather conditions and also at different times in a day. Each log segment in the dataset contains data from seven ring cameras, two lidar sensors and two front-facing stereo cameras. In our Auto QA data-set, we have used data from ring cameras and lidar sensors.Images from seven high definition ring cameras are of dimension (1921 x 1220) at 30 Hz. lidar sensors having a range of 200m produce approx 107000 points at 10 hz around reference vehicle. Seven images from ring cameras correspond to seven different directions around the reference vehicle and provide 360 degree field of view around the reference car. Argoverse 3D tracking dataset also contains 3D bounding box annotation for lidar data. These annotations provide labels for 15 different object categories. Questions for the dataset are generated using this 3D bounding box annotation. Process of question-answer pair generation is explained in 3.2.2

For images of our Auto QA dataset, we sampled 2D images from logs. We sampled images at 10hz, so that images are in sync with lidar data. Thus each instance of AutoQA dataset, consists of seven 2D images, corresponding point

cloud data and QA(question-answer) pairs. We will be referring to the collection of these seven images as a "scene" in the AutoQA dataset. The statistics for the dataset are provided in table 2.

3.1. Dataset Analysis

We separate 24 logs from Argoverse Dataset for test data generation while rest are used for training dataset generation. For both test and training data, we sampled image scene and point cloud data at 10hz from each log segment and randomly shuffle all scenes. Thus training data consists of approximately 13000 scenes, while test data consists of approximately 5000 scenes that are independent of the training scenes. For both training and test data we synthesize QA pairs (see section 3.2.2). Out of all generated question on training data scenes, we use an 80-20 split for obtaining a validation QA set.

Object	count
vehicle	166349
pedestrian	52368
obstacle	22844
other_mover	2413
bicycle	6637
large_vehicle	5895
motorcycle	1108
bicyclist	3928
motorcyclist	925
emergency_vehicle	414
moped	849
animal	83
bus	4359
trailer	1490
stroller	286

Table 1. Different object types present in dataset

Split	Scene	Images	Lidar	Question
train	13122	13122x7	13122	170253
val	13122	13122x7	13122	42563
test	5015	5015x7	5015	80576

Table 2. **Dataset Statistics**

3.2. Dataset Attribute

3.2.1 Image Scene and Point Cloud Representation

For each generated question in the dataset, we have an image scene. Each image scene is a group of seven 2D images corresponding to seven different directions around the reference vehicle. The seven different directions are front left, centre, front right, side left, side right, rear left and rear right. These seven images give a 360-degree field of view around the reference vehicle. In addition to 2D image scene, the dataset also comprises of lidar sensor's 3D point cloud data corresponding to each image scene. Each point cloud consists of approx *107000* points around the reference vehicle.

3.2.2 Question Generation and Representation

Similar to diagnostic dataset CLEVR [10] we used functional programs to automatically generate questions from the scene graph. Unlike CLEVR dataset, the scene graph is not directly available in our case. As our dataset is built using Argoverse Dataset, in-order to generate a scene graph, we use already available 3D bounding box annotations given for lidar sensor point cloud data. We project these 3D bounding boxes to 2D images corresponding to all the seven directions. This gives us information about which object is present in which direction. Taking these seven directions as nodes, we create a scene graph around the reference vehicle. Thus each scene graph contains information about the different types of objects present in the seven directions.

Based on the scene graph we generate three different families of questions. Each question family consists of the template, parameters, nodes and constraints. All questions are generated taking a person inside the vehicle as a reference. For the generation of questions, we fill template parameters and execute this template over the scene graph to generate ground truth answer. Depth-first search is used to instantiate the template with valid parameters. Question is generated only if there is a unique answer to the question. Finally, parameter values are replaced to generate a natural language question. We have observed a maximum question length of 13 for our dataset. For example: consider "*exist type* question" *Is there a <object> to my <direction>?*. Unlike simple VQA, answering model will not only need to consider an object but also need to understand which direction the person inside the car is asking about.

Similarly, a *closest type* of question, *What is the nearest object to my side left?* can be generated using template *What is the nearest object to my <direction>?*. Here *my* refers to agent/passenger. These question would be of relevance to the passenger sitting inside vehicle looking for a specific entity nearby. Answer to such queries can be one of the 15 annotated objects. Count type question can be generated using templates, *How many <object> are on my <direction>?* or *What is the number of <object> to my <direction>?*. In all templates *direction* can take one of the seven directions *front left, centre, front right, side left, side right, rear left, rear right*. *<object>* can be one of the 15 different object as given in table 1. We thereby are able to generate around 4000 unique different questions by using different phrases for each type of question in order to create a sufficient variety of questions. An illustration of the various question distributions are provided in figure 2.

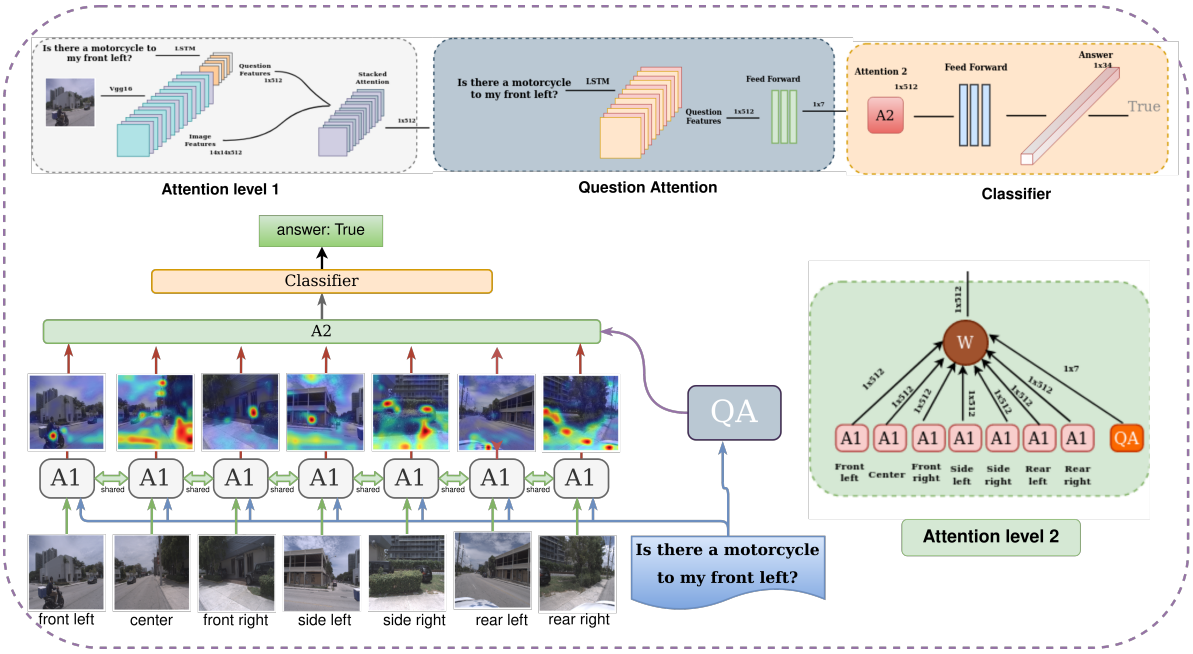


Figure 3. Image Based Model Diagram(A1 refers to attention level-1, A2 refers to attention level-2 and QA refers to Question Attention)

3.2.3 Answer Representation

Answer to exist type of question is a Boolean value i.e true or false. For count type question, the answer is a numeral. We have observed a max count of 18 for a single object, in our dataset. Answer to closest type question represents the nearby object queried corresponding to a direction. For our dataset, answer to these type of question is one of the 15 different objects for which annotation are available in Argoverse dataset.

4. Baseline Models

In order to solve the task, appropriate baselines are provided by us. Specifically, we consider image-based baselines, point-cloud based baselines as well as combined models. Further details about samples of our dataset, source code for all of our baseline experiments and more experiments results are present in our project page. [1]

4.1. Image Based model

Given an image scene consisting of seven images and question, we have used hierarchical attention over images using encoded question representation as context. Consider the question *"Is there a pedestrian to my side left?"*. In the hierarchical model, first-level attention will learn to attend the queried object in question while second level attention will learn, which image to attend to correctly answer the question. First level attention layer i.e *level-1 attention* attend over different regions of all seven images corresponding to different directions. This gives soft attention over image spatial regions. Thus bottom level attention will learn to attend specified entity in all seven images. We experimented with different existing VQA attention for bottom level attention, details of which are given in figure 3. After level-1 attention, for top-level attention i.e *level-2 attention*, we used self-attention over the question to learn seven attention weights, finally, we take a weighted sum of the output of level-1 attention and attention weights learned using self-attention over the question. Thus top-level attention will learn to attend the image corresponding to the specified direction. Finally, a classifier gives a classification score over candidate answer using the output of top-level attention.

4.2. Point Cloud Based model

VQA model on 2D images take as an input a 2D image, a question and generate distribution over all possible answer. In these models, first images are generally represented as feature vector over spatial regions. These feature vectors are extracted from the intermediate layer of pretrained ImageNet [7] classification models. Unlike VQA on 2D images, we don't have any pretrained model which can be used as transfer learning for feature extraction directly. For learning features using lidar point cloud data, we use PointNet++, as our backbone network. Just like CNN for images, PointNet++ extract features from local regions and grouped them to aggregate high-level features. Thus, it is a perfect replacement for CNN to learn spatial fea-

[1]Project Page: https://delta-lab-iitk.github.io/AUTO-QA/

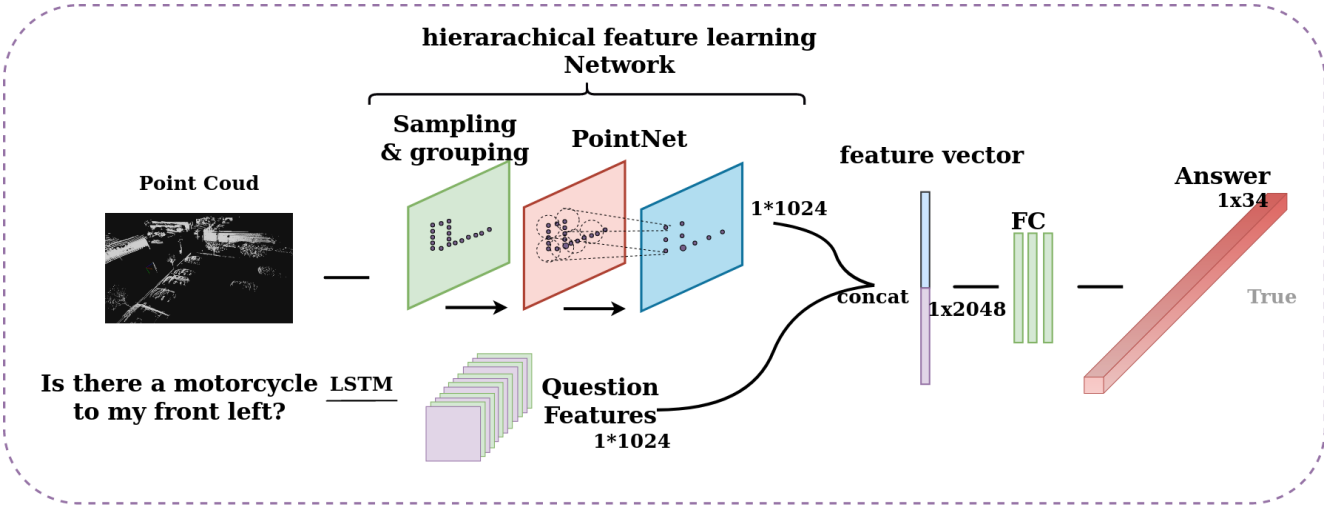


Figure 4. **Point Cloud based Model diagram** with PointNet++ as backbone network

tures for VQA model on the point cloud. We have used both single-scale and multi-scale grouping for learning feature vector.Further implementation details are explained in section 5.3. For question embedding, we first tokenize the question into words. Each word in question is encoded using learned embedding layer. Finally, we used single layer LSTM [9]. We used learned embedding layer followed by single layer LSTM for question features encoding. Finally, the output of the backbone network and the question feature vector is fed to a classifier which results out distribution over the possible answer (see fig 4). The whole model is trained in the end to end manner.

4.3. Image and Point Cloud Based Model

From table 4, 5, we can infer performance of individual image based and lidar based model. We take advantage of both image and lidar based model for answering queries. We take two pipelines one for each image and lidar. Lidar pipelines is similar to Point Cloud based model 4.2.Image pipeline is just like model described in 4.1. We concatenated *output of level-2 attention* of image based model and input of classifier in Point Cloud Based Model. Finally this concatenated feature vector(image+lidar) is used for classification into possible answers.Similar to Images Based Model, we experimented with level-1 attention in image pipeline and single and multi-scale grouping in point cloud based model.

5. Experiments

We evaluate Autonomous Question Answering (Auto QA) dataset in the following manner. First, we performed an ablation analysis of our dataset in section 5.1, next we experimented with existing different baseline attention model for level-1 attention discussed in section 4.1 in our image based model. Second, we experimented with the different architecture of backbone network for our Point Cloud Based Model explained in section 4.2. Finally we tested our Combination Model with different combination of point cloud backbone network with different level-1 attention model.

Models	Classifier Dim.	Accuracy
LSTM(only question)	1024	47.6
CNN_LSTM(dot prod.)	2048	53.8
CNN_LSTM(concat)	2048x7+1024	54.5

Table 3. Ablation Results

5.1. Ablation Analysis

To check whether there is any language bias in our dataset, we first used question only model. We used a learned embedding layer with hidden dimension 300 for encoding words which is then followed by a single layer LSTM with hidden dimension 1024. This 1024 dim question embedding is used for classification over the possible answer. Next, we used Resnet-152 to extract 2048 dimension feature vector for all seven images. We experimented with both concatenation and the dot product of these spatial image feature vector and encoded question vector. For concatenation, we concat seven vector corresponding to seven different directions with the question feature vector of dimension 1024. Thus we have a vector of dimension 2048x7+1024 *(seven attention feature vector and question feature vector)*, this is used for classification over possible answers.

In the case of the dot product of images and question feature vector, we used LSTM with hidden dimension 2048 and take the element-wise dot product of all seven image vector (2048 dim) and question vector. Finally, the resultant 2048 dim vector is used for classification. As we can see in table 3 concat perform slightly better than dot product. Also language only model perform poorly, indicating very little question bias in our dataset.

5.2. Experiments on Baseline Attention Methods

For level-1 attention we experimented with different existing attention model of visual question answering task. We have evaluated our dataset on various baseline VQA models such as SAN, MCB, DAN, MLB, MFB and MUTAN based grid attention method for our AutoQA dataset. We extracted 7x7x512 dimension feature vector from pretrained vgg-16. First, we tested only level-1 attention for our dataset. We take the output from the attention layer which is a weighted feature vector over spatial region based on attention weights. This is repeated for all seven images,with parameters of level-1 attention model shared for all seven images. Then we concat these seven feature vector. We also concat question feature vector with concatenated attention vector of seven images. This concatenated feature vector is used in classification over possible answers. We have used learned embedding layer with hidden dimension 300. Output of embedding layer is fed to single layer LSTM. In case of SAN, MCB and MFB, hidden dim for LSTM is 512 while for MLB and MUTAN hidden dim is 1024. Only for DAN, we have used bidirectional LSTM with hidden dim 1024. For SAN, we experimented with two stack of attention. Similarly with MLB and MUTAN, we experimented with multiple glimpsed over image feature vector. For MLB and MUTAN we experiment using 4 and 2 glimpses respectively.For DAN, we have used 2 attention steps.

Models	**Classifier Dim.**	**Acc %**
SAN(level-1)+concat	512x7+512	58.9
SAN(level-1)+level-2	512	71.4
MCB(level-1)+concat	512x7+512	57.2
MCB(level-1)+level-2	512	67.4
DAN(level-1)+concat	1024x7+512	52.0
DAN(level-1)+level-2	1024	55.6
MLB(level-1)+concat	4800x7+1024	62.8
MLB(level-1)+level-2	4800	66.5
MFB(level-1)+concat	500x7+512	66.0
MFB(level-1)+level-2	500	68.3
MUTAN(level-1)+concat	510x7+1024	68.2
MUTAN(level-1)+level-2	510	73.1

Table 4. Comparison of Image Based Model: using only *level-1 attention* and both *level-1 and level-2 attention* with dimension of input feature for Classifier

As we can observe from table 4, using SAN for level-1 attention and concat of all seven attention vector with question vector(512x7+512 dim), we saw improvement around 3% over baseline CNN_LSTM_concat 3, further incorporating our level-2 attention as described in section 4.1, we further get around 13 % improvement.Among other method MCB perform similar to SAN for level-1 attention followed by concat of attention and question vector(512x7+512 dim), approx 2% improvement, but with our level-2 attention its performance is slightly less as comparison to SAN.

Similarly using MUTAN, MLB and MFB just for level-1 attention for all seven images show improvement over baseline method CNN_LSTM_concat, 14%, 8% and 12% respectively. DAN for only level-1 attention doesn't perform well(52%) but shows approx 4% increase with level-2 attention. These method further show further improvement using our model with level-2 attention, around 5%,4% and 2% improvement. As we can observe from table-4, using MUTAN for level-1 attention and then using our level-2 attention outperform all methods with accuracy of 73.1%. Also all methods show improvement using our hierarchical attention model(level-1+level-2 attention).

Models	**Radii**	**Acc.**
SSG+LSTM(dot prod)	0.2	52.7
SSG+LSTM(concat)	0.2	54.2
MSG+LSTM(dot prod)	0.1,0.2,0.4	53.1
MSG+LSTM(concat)	0.1,0.2,0.4	56.5

Table 5. Comparison of VQA results with 3D Point Cloud Data using PointNet++ SSG(single scale grouping) and MSG (multi scale grouping)

5.3. Backbone Network for Point Cloud Model

For each 3D point cloud corresponding to a given question, we first sub-sample 16000 points, using farthest point sampling. Now for hierarchical feature learning network as used in PointNet++. For single-scaling grouping, we have used four abstraction layer with group sizes of (4096,512,256,1) to obtain a single dimension feature vector. For multi-scale grouping, group sizes are same as single-scale grouping with three different radii (0.1,0.2,0.4). For question feature, we have used LSTM with hidden dimension 1024 for question feature encoding. We experimented with both concatenation and the dot product of the question feature vector and point cloud feature vector.

As we can infer from table 5, using multi-scale grouping perform slightly better as compared to single-scale grouping(2%).Also by comparing results of table 3 with these results, we can see that using PointNet++ as the backbone network for VQA using Point Cloud data, perform similar to CNN_LSTM(concat) baseline model(54% and 56%). Thus can be used as a baseline model for VQA on 3D point cloud data.

5.4. Combination Model Analysis

We experimented with level-1 attention for image pipeline using each different attention as in 5.2 in combination with single and multi scale grouping. From tables 6 and 4, we observe that using SAN for image pipeline and point-cloud model pipeline with single-scale grouping (512+512+1024, 512 dimension vector from image

Models	Classifier Dim.	Acc %
SAN(level-1,2)+SSG	512+512+1024	72.8
SAN(level-1,2)+MSG	512+512+1024	**74.5**
MCB(level-1,2)+SSG	512+512+1024	67.6
MCB(level-1,2)+MSG	512+512+1024	68.2
DAN(level-1,2)+SSG	1024+512+1024	55.6
DAN(level-1,2)+MSG	1024+512+1024	57.0
MLB(level-1,2)+SSG	4800+1024+1024	66.5
MLB(level-1,2)+MSG	4800+1024+1024	67.6
MFB(level-1,2)+SSG	500+512+1024	67.4
MFB(level-1,2)+MSG	500+512+1024	69.2
MUTAN(level-1,2)+SSG	510+1024+1024	73.7
MUTAN(level-1,2)+MSG	510+1024+1024	**74.8**

Table 6. Comparison of **Combined Image and Point Cloud Based Model**

pipeline, 512+1024 dimension vector from lidar pipeline), we get an improvement of around 1.5%, while with multi-scale grouping the improvement is approximately 3% taking it to 74.5%. Similarly, MLB, MFB, DAN and MCB show improvement using point-cloud data with both single scale and multi-scale grouping. With single-scale grouping improvement is not much (0.5-1%) improvement. But with multi-scale grouping, we see an improvement of around 3% in case of MUTAN and DAN, while MLB, MFB, MCB show an improvement of around 2%. MUTAN shows the best results among all types of models(image, point cloud and combined) with 74.8%.

5.5. Level-1 and Level-2 Attention Visualisation

Our idea behind use of level-1 attention is to attend the specified object in the object.Thus in order to correctly answer a question ,level-1 attention must attend the image region corresponds to specified object in question. In order to visualize attention,we overlay attention probability distribution matrix, obtained for all seven images using a given query. Consider example given in figure 5. Here question is *"Is there a vehicle to my side right?"*, we can see that SAN based attention, correctly attend over vehicle in images as compared to other methods.

We conduct a quantitative Analysis on level-2 Attention Localization on our Auto QA dataset. Each question about an object in our dataset is associated with some direction. We have employed level-2 attention in our models 4.1 to correctly identify which direction answering model must look into for given question. From table 7 , we can see that our level-2 attention module focuses on the correct image.

6. Conclusions

Through this work, we introduce a novel semantic challenge 'AutoQA' that aims to evaluate the functional aspect of ability to localize accurately while answering questions. Our work includes a practical aspect of answering a question by incorporating all the views and point-cloud data that are available for answering the question. This enables us to consider various aspects for the task, i.e. to what extent is an image-based approach alone able to solve the task and is the image-based approach able to attend appropriately to the right image while answering the question. Our analysis also shows that combined models that incorporate image and point-cloud information perform marginally better as a baseline. Further research could help explore the use of multiple modalities. Moreover, we could analyse the failure modes for this task to empirically evaluate the performance of autonomous driving AI agents. It could also serve as an aid for advanced driver assistance systems that incorporate more sensors for safer driving. We believe that through this work we make a meaningful contribution for the autonomous driving task.

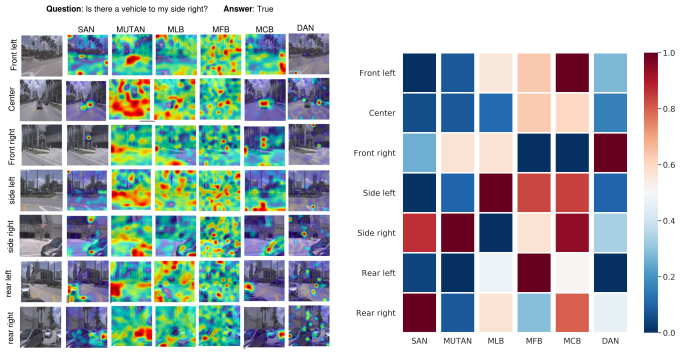


Figure 5. In this figure we visualise level-1 and 2 attention corresponding to instance of Auto QA Sample. We also have shown visualisation results for different model along all directions.

Models	precison	recall	f1 score
SAN(level-1,2)	64.79	81.57	72.21
SAN(level-1,2)+SSG	78.47	72.76	74.22
SAN(level-1,2)+MSG	78.21	73.40	75.4
MCB(level-1,2)	63.67	80.58	71.12
MCB(level-1,2)+SSG	72.15	73.48	72.1
MCB(level-1,2)+MSG	72.28	71.45	71.8
DAN(level-1,2)	64.33	78.32	70.6
DAN(level-1,2)+SSG	70.22	68.76	69.4
DAN(level-1,2)+MSG	70.49	69.34	69.91
MLB(level-1,2)	64.8	79.1	71.2
MLB(level-1,2)+SSG	72.59	69.7	71.03
MLB(level-1,2)+MSG	73.2	69.81	71.46
MFB(level-1,2)	63.25	79.46	70.04
MFB(level-1,2)+SSG	71.49	68.56	70.51
MFB(level-1,2)+MSG	71.40	68.2	68.76
MUTAN(level-1,2)	68.79	79.83	73.89
MUTAN(level-1,2)+SSG	78.25	74.65	76.4
MUTAN(level-1,2)+MSG	78.13	74.2	76.1

Table 7. Comparison of level-2 localization analysis

References

[1] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6077–6086, 2018.

[2] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. VQA: Visual Question Answering. In *International Conference on Computer Vision (ICCV)*, 2015.

[3] Yalong Bai, Jianlong Fu, Tiejun Zhao, and Tao Mei. Deep attention neural tensor network for visual question answering. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 20–35, 2018.

[4] Hedi Ben-younes, Remi Cadene, Matthieu Cord, and Nicolas Thome. Mutan: Multimodal tucker fusion for visual question answering. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[5] Ming-Fang Chang, John W Lambert, Patsorn Sangkloy, Jagjeet Singh, Slawomir Bak, Andrew Hartnett, De Wang, Peter Car, Simon Lucey, Deva Ramanan, and James Hays. Argoverse: 3d tracking and forecasting with rich maps. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[6] Yilun Chen, Shu Liu, Xiaoyong Shen, and J. Jia. Fast point r-cnn. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9774–9783, 2019.

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[8] Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach. Multimodal compact bilinear pooling for visual question answering and visual grounding. *arXiv preprint arXiv:1606.01847*, 2016.

[9] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[10] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2901–2910, 2017.

[11] Jin-Hwa Kim, Jaehyun Jun, and Byoung-Tak Zhang. Bilinear attention networks. In *Advances in Neural Information Processing Systems*, pages 1571–1581, 2018.

[12] Jin-Hwa Kim, Kyoung-Woon On, Woosang Lim, Jeonghee Kim, Jung-Woo Ha, and Byoung-Tak Zhang. Hadamard product for low-rank bilinear pooling, 2016.

[13] A. Lang, Sourabh Vora, H. Caesar, Lubing Zhou, J. Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12689–12697, 2019.

[14] B. Li, Tianlei Zhang, and Tian Xia. Vehicle detection from 3d lidar using fully convolutional network. *ArXiv*, abs/1608.07916, 2016.

[15] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Hierarchical question-image co-attention for visual question answering. In *Advances In Neural Information Processing Systems*, pages 289–297, 2016.

[16] Mateusz Malinowski and Mario Fritz. A multi-world approach to question answering about real-world scenes based on uncertain input. In *Advances in Neural Information Processing Systems (NIPS)*, 2014.

[17] Hyeonwoo Noh, Paul Hongsuck Seo, and Bohyung Han. Image question answering using convolutional neural network with dynamic parameter prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 30–38, 2016.

[18] Badri Patro and Vinay P. Namboodiri. Differential attention for visual question answering. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[19] Charles R Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J Guibas. Frustum pointnets for 3d object detection from rgb-d data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 918–927, 2018.

[20] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. *arXiv preprint arXiv:1612.00593*, 2016.

[21] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in neural information processing systems*, pages 5099–5108, 2017.

[22] Mengye Ren, Ryan Kiros, and Richard Zemel. Exploring models and data for image question answering. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2953–2961, 2015.

[23] Shaoshuai Shi, Chaoxu Guo, L. Jiang, Zhe Wang, Jianping Shi, X. Wang, and Hongsheng Li. Pv-rcnn: Point-voxel feature set abstraction for 3d object detection. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10526–10535, 2020.

[24] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointrcnn: 3d object proposal generation and detection from point cloud. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[25] Kevin J Shih, Saurabh Singh, and Derek Hoiem. Where to look: Focus regions for visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4613–4621, 2016.

[26] Huijuan Xu and Kate Saenko. Ask, attend and answer: Exploring question-guided spatial attention for visual question answering. In *European Conference on Computer Vision*, pages 451–466. Springer, 2016.

[27] Huijuan Xu and Kate Saenko. Dual attention network for visual question answering. In *ECCV 2016 2nd Workshop on Storytelling with Images and Videos (VisStory)*, 2016.

[28] Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng, and Alex Smola. Stacked attention networks for image question

answering. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[29] 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 conference on computer vision and pattern recognition*, pages 6281–6290, 2019.

[30] Zhou Yu, Jun Yu, Jianping Fan, and Dacheng Tao. Multi-modal factorized bilinear pooling with co-attention learning for visual question answering. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[31] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4490–4499, 2018.