From Pixels to Objects: Cubic Visual Attention for Visual Question Answering

Jingkuan Song, Pengpeng Zeng, Lianli Gao and **Heng Tao Shen***
Center for Future Media and School of Computer Science and Engineering,
University of Electronic Science and Technology of China, Chengdu 611731, China,
{lianli.gao, pengpeng.zeng}@uestc.edu.cn, jingkuan.song@gmail.com and shenhengtao@hotmail.com

Abstract

Recently, attention-based Visual Question Answering (VQA) has achieved great success by utilizing question to selectively target different visual areas that are related to the answer. Existing visual attention models are generally planar, i.e., different channels of the last conv-layer feature map of an image share the same weight. This conflicts with the attention mechanism because CNN features are naturally spatial and channel-wise. Also, visual attention models are usually conducted on pixel-level, which may cause region discontinuous problem. In this paper we propose a Cubic Visual Attention (CVA) model by successfully applying a novel channel and spatial attention on object regions to improve VQA task. Specifically, instead of attending to pixels, we first take advantage of the object proposal networks to generate a set of object candidates and extract their associated conv features. Then, we utilize the question to guide channel attention and spatial attention calculation based on the con-layer feature map. Finally, the attended visual features and the question are combined to infer the answer. We assess the performance of our proposed CVA on three public image QA datasets, including COCO-QA, VQA and Visual7W. Experimental results show that our proposed method significantly outperforms the state-of-the-arts.

1 Introduction

Visual Question Answering (VQA) is an interdisciplinary research problem, which has attracted extensive attention recently [Yang *et al.*, 2016; Lu *et al.*, 2016; Hyeonseob Nam and Kim, 2017; Lu *et al.*, 2018]. It has the potential to be applied for assisting the visually impaired people and automatically querying on large-scale image or video datasets [Xu and Saenko, 2016; Hyeonseob Nam and Kim, 2017]. Compared with image captioning task, the VQA task requires a deeper understanding of both image and question rather than a coarse understanding [Goyal *et al.*, 2017; Antol *et al.*, 2015]. It inspects intelligent system's ability by inferring a correct answer for the visual question.

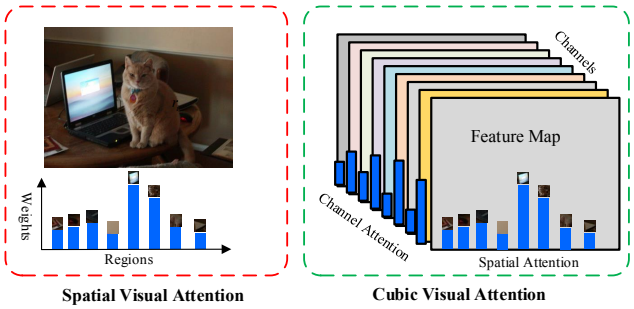

Figure 1: The illustration of spatial visual attention and cubic visual attention. Spatial visual attention is modelled as a weight matrix on the last conv-layer feature map of a CNN encoding an input image. Different channels share the same weights. Instead, cubic visual attention learns both spatial and channel attention, which conforms to the nature of conv features, i.e., spatial and channel-wise.

Existing works for VQA can be generally classified into two categories: 1) Typical CNN-RNN models, which transfer image captioning frameworks by integrating CNN with Recurrent Neural Networks (RNN) to solve VQA tasks [Gao *et al.*, 2015; Ren *et al.*, 2015; Malinowski *et al.*, 2015]; and 2) Question-guided visual attention mechanisms, which aim to discover the most important regions to answer a question by exploring their relationships [Fukui *et al.*, 2016; Lu *et al.*, 2016; Hyeonseob Nam and Kim, 2017; Shih *et al.*, 2016; Xu and Saenko, 2016; Yang *et al.*, 2016]. More specifically, question-guided visual attentions are conducted by concatenating the semantic representation of a question with each candidate region and then put them into a multiple layer perceptron (MLP) or applying the dot product of each word embedding and each spatial location's visual feature [Xu and Saenko, 2016]. In addition, Yang *et al.* [Yang *et al.*, 2016] proposed a stacked attention model by utilizing semantic representation of a question as query to search for the regions in an image multiple times to infer an answer progressively. In [Hyeonseob Nam and Kim, 2017], a dual attention was introduced to infer the answers by attending to specific regions in images and words in text. Lu *et al.* [Lu *et al.*, 2016] presented a co-attention that jointly performs question-guided vi-

*Corresponding author: Lianli Gao, Heng Tao Shen

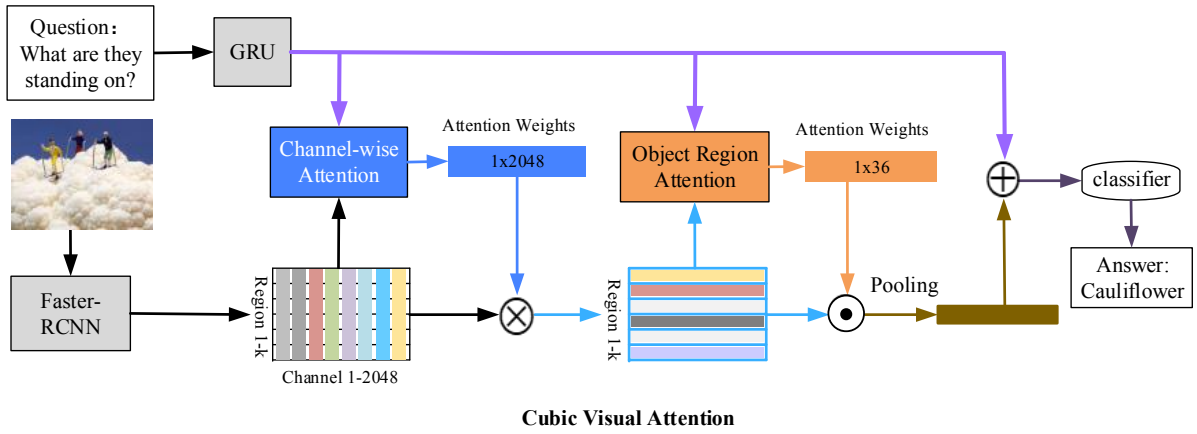

Figure 2: The framework of our proposed CVA for VQA. It consists of four main components, feature extraction, object proposal, cubic visual attention estimation and answer prediction.

sual attention and image-guided question attention to address 'where to look' and 'what words to listen to'.

Although promising results have been achieved, typical CNN-RNN models resort to a global image presentation which may contain noisy or unnecessary information for the related question. To some extent, question-guided visual attention mechanisms have tackled this problem generally by spatial attention, i.e., the attention is modeled as spatial probabilities that re-weight the last conv-layer feature map of a CNN encoding an input image. However, such spatial attention does not necessarily conform to the attention mechanism–a dynamic feature extractor that combines contextual fixations over time, as CNN features are naturally spatial and channel-wise [Chen *et al.*, 2017]. Image features are generally extracted by deep Convolutional Neural Networks (CNNs) [He *et al.*, 2016; Song *et al.*, 2017; Gao *et al.*, 2017]. Starting from an input color image of the size $W \times H \times 3$, the last convolutional layer consisting of C-channel filters output a $W' \times H' \times C$ feature map. Different channels of a feature map is essentially activation response maps of the corresponding filter, and channel-wise attention can be viewed as the process of selecting semantic attributes on the demand of the sentence context. For example, when we want to predict *cat*, our channel attention (e.g., in the conv5 3/conv5 4 feature map) will assign more weights on channel-wise feature maps generated by filters according to the semantics like furry texture, ear, and cat-like shapes. Channel attention plays a different role compared with spatial attention, and it is rarely addressed in previous works.

In this paper, we take the full advantage of two characteristics (i.e., channel and spatial) of object-based region features for visual attention-based VQA. Specifically, we propose a novel Cubic Visual Attention (CVA) framework by successfully applying a channel attention and a spatial attention to assist VQA task. An object detection network [Ren *et al.*, 2017], which has the potential to enable nearly cost-free object region proposal, is applied to extract top-k objects in an image and each object is represented by a D-dimensional vector. Next, the channel-wise attention learns to pay attention to specific channels of the last conv feature map. Thirdly, a region-based spatial attention is applied on the channel-attended features to select related objects. Finally, an answer is inferred by considering both the attended visual features and the question.

2 Cubic Visual Attention for VQA

The VQA task is to predict an answer from a question and a related image. In this section, we introduce our proposed CVA framework (shown in Fig. 2) for VQA, and it consists of 1) a feature extraction component, which extracts the features for question and input image; 2) a channel attention for selecting filters related to the high-level object semantic; 3) a region-based spatial attention for learning to focus on the regions of the image that are important and 4) an answer prediction layer to infer to answer. In this section, we describe each of them.

2.1 Input Representations

Image Features. For the input image, we use an existing state-of-the-art object detection network Faster R-CNN [Ren *et al.*, 2017] for object proposal and feature extraction. Specifically, each image is input into Faster R-CNN model to obtain the top-K candidate objects. For each selected region/box k, $\mathbf{v}_k$ is defined as the mean-pooled convolutional feature from this region and each $\mathbf{v}_k$ indicates a vector with D dimensions. Therefore, the input image $\mathbf{V}$ is defined as follow:

$$\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_K], \mathbf{v}_k \in \mathbb{R}^{1\times D} \quad (1)$$

Encoding Question Features. Suppose that $[\mathbf{q}_1, \mathbf{q}_2, ..., \mathbf{q}_T]$ represents an question, where $\mathbf{q}_t$ is an one-hot representation for the word at position t, and T is the length of the question. Each word representation $\mathbf{q}_t$ is transferred into a lower dimensional vector $\mathbf{x}_t$ with an embedding matrix $\mathbf{W}_e^{\mathbf{q}}$.

$$\mathbf{x}_t = \mathbf{W}_e^q \mathbf{q}_t \quad (2)$$

There are various approach to encode a question like bag-of-words. However, Long Short-term Memory (LSTM) and gated recurrent unit (GRU) are two of the most popular mechanisms to encode a sentence in machine translation and great results have been obtained. In this paper, we employ GRU to encode our questions, and the gradient chains of GRU do not vanish due to the length of questions. For the t-th time step, the GRU unit takes the embedding vector $\mathbf{x}_t$ as an input, updates the gate $\mathbf{z}_t$, resets gate $\mathbf{r}_t$, and then outputs a hidden state $\mathbf{h}_t$. After T steps, we obtain the T-th output $\mathbf{h}_T$ to represent the semantic information of a question. We formulate our encoding process as below:

$$\mathbf{z}_t = \sigma(\mathbf{W}_z\mathbf{x}_t + \mathbf{U}_z\mathbf{h}_{t-1} + \mathbf{b}_z) \quad (3)$$

$$\mathbf{r}_t = \sigma(\mathbf{W}_r\mathbf{x}_t + \mathbf{U}_r\mathbf{h}_{t-1} + \mathbf{b}_r) \quad (4)$$

$$\tilde{\mathbf{h}}_t = tanh(\mathbf{W}_h\mathbf{x}_t + \mathbf{U}_h(\mathbf{r}_t \circ \mathbf{h}_{t-1}) + \mathbf{b}_h) \quad (5)$$

$$\mathbf{h}_t = \mathbf{z}_t \circ \mathbf{h}_{t-1} + (1 - \mathbf{z}_t) \circ \tilde{\mathbf{h}}_t \quad (6)$$

$$\mathbf{Q} = \mathbf{h}_T \quad (7)$$

where $\mathbf{W}_{z,r,h}, \mathbf{U}_{z,r,h}, \mathbf{b}_{z,r,h}$ are the parameter which needed to be learn. Note that σ is a sigmiod activation function, and $\circ$ is used as the Hadamard product or element-wise multiplication. Through the question feature encoding process, we generate the question representation $\mathbf{Q}$, which equals to the last output of GRU ($\mathbf{h}_T$).

2.2 Channel Attention

In this section, we introduce a novel channel-wise attention mechanism to attend the visual features $\mathbf{V}$. Each $\mathbf{v}_i$ is obtained by mean pooling the conv features of a given box spatial location. Essentially, each channel of a feature map in CNN is correlated to a convolutional filter which performs as a pattern detector. For instance, the lower-level filters detect visual clues such as edges and color, while the higher-level filters detect semantic patterns, such as attributes or object components. In this work, our object region features are pooled from the last conv feature map, thus each channel represents the semantic patterns of the detected objects within an image. Therefore, conducting a channel-wise attention can be viewed as a process of choosing object semantic attributes.

For channel-wise attention, we first reshape $\mathbf{V}$ to $\mathbf{U}$ and $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_D]$, where $\mathbf{u}_i \in \mathbb{R}^k$ represents the i-th dimension of the whole object feature V, and D is the dimension of $\mathbf{v}_i$ or it is the total number of channels for each object region. Next, we apply a mean pooling for each channel to generate the channel vector

$$\overline{\mathbf{u}} = [\overline{u}_1, \overline{u}_2, ..., \overline{u_D}] \quad (8)$$

where $\overline{u_i}$ is the mean vector of $\mathbf{u}_i$, which represents the i-the channel features. Our channel-wise attention $\mathbf{A}_c$ is defined as below:

$$\mathbf{b} = tanh((\mathbf{W}_{vc}\overline{\mathbf{u}} + \mathbf{b}_{vc}) \otimes (\mathbf{W}_{qc}\mathbf{Q} + \mathbf{b}_{qc})) \quad (9)$$

$$\beta = softmax(\mathbf{W}_c\mathbf{b} + \mathbf{b}_c) \quad (10)$$

where $\mathbf{W}_{vc}$, $\mathbf{W}_{qc}$ and $\mathbf{W}_c$ are embedding matrices, $\mathbf{b}_{vc}$, $\mathbf{b}_{qc}$ and $\mathbf{b}_c$ are bias terms, and $\otimes$ indicates the outer product of vectors. To sum up, we obtain our channel-wise attention weight β through our channel-wise attention $\mathbf{A}_c$, defined as follow:

$$\beta = \mathbf{A}_c\left(\overline{\mathbf{U}}, \mathbf{Q}\right) \quad (11)$$

2.3 Object Region-based Spatial Attention

With above step, we obtain the channel-wise attention weight β, thus we can feed β to a channel-wise attention function $\mathbf{f}_c$ to calculate a modulated feature map $\mathbf{V}^c$:

$$\mathbf{V}^c = \mathbf{f}_c\left(\beta, \mathbf{V}\right) \quad (12)$$

where $\mathbf{f}_c$ is an channel-wise multiplication for region feature map channels and corresponding channel weights. In addition, eventually $\mathbf{V}^c$ is

$$\mathbf{V}^c = \{\mathbf{v}_1^c, \mathbf{v}_2^c, ..., \mathbf{v}_k^c\}, \mathbf{v}_i^c \in \mathbb{R}^D \quad (13)$$

Suppose, we have $\mathbf{V}^c$ where $\mathbf{v}_i^c$ indicates the visual feature of the i-th object region. In general, a question may only relate to one or several particular regions of an image. If we want to ask 'what the color of the dog', then only the dog object region contains the useful information, therefore typical CNN-RNN which employing the whole global visual feature may lead to sub-optimal results due to the irrelevant visual regions shown in the input image. Instead of considering each object region equally, our region-based spatial attention mechanism aims to target the most related region with an referred question. Given the previous calculated $\mathbf{V}^c$, a single-layer neural network is adopted to take both $\mathbf{V}^c$ and $\mathbf{Q}$ as inputs to generate a new feature $\mathbf{a}$, and then a softmax function is followed to compute the region-based spatial attention weight η. The object region-based spatial attention $\mathbf{A}_s$ is defined as :

$$\mathbf{a} = tanh(\mathbf{W}_{vo}\mathbf{V}^c + \mathbf{b}_{vo}) \oplus (\mathbf{W}_{qo}Q + \mathbf{b}_{qo}) \quad (14)$$

$$\eta = softmax(\mathbf{W}_o\mathbf{a} + \mathbf{b}_o) \quad (15)$$

where $\mathbf{W}_{vo}$ and $\mathbf{W}_{qo}$ are the embedding matrices that project both visual and question features into a common latent space. In addition, $\mathbf{W}_o$ is a set of parameters that needs to be learn. $\mathbf{b}$ is the model bias. $\oplus$ is the addition of a matrix and a vector. Moreover, $\eta \in \mathbb{R}^k$ is a k-dimensional vector, which represents the importance of each object region. Therefore, the weights η can be calculated by:

$$\eta = \mathbf{A}_s\left(\mathbf{V}^c, \mathbf{Q}\right) \quad (16)$$

Furthermore, to deal with multiple object regions, a simple strategy usually is used to compute the average of features across the whole image, and this generated feature is used as input to integrate question feature to generate an answer:

$$\mathbf{V}^s = \frac{1}{k}\sum\nolimits_{i=0}^{k} \mathbf{v}_i^c \quad (17)$$

However, as mentioned above this strategy effectively effectively integrate multiple regions into a single vector, neglecting the inherent spatial structure and leading to the loss of information. Instead of using above simple strategy, we apply η to attend where to look at and it is defined as bellow:

$$\mathbf{V}^s = \mathbf{f}_s\left(\eta, \mathbf{V}^c\right) = \frac{1}{k}\sum\nolimits_{i=0}^{k} \eta_i\mathbf{v}_i^c \quad (18)$$

2.4 Answer Prediction

Following previous work [Wang *et al.*, 2017; Lu *et al.*, 2016], we treat the answer prediction process as a multi-class classification problem, in which each class corresponds to a distinct

answer. We predict the answer based on the stacked attended image visual feature $\mathbf{V}^s$ and a question feature $\mathbf{Q}$, and a multi-layer perceptron (MLP) is used for classification:

$$h = tanh(\mathbf{W}_v\mathbf{V}^s + \mathbf{W}_q\mathbf{Q} + \mathbf{b}_h) \quad (19)$$

$$p = softmax(\mathbf{W}_h\mathbf{h} + \mathbf{b}_p) \quad (20)$$

where $\mathbf{W}_v, \mathbf{W}_q, \mathbf{W}_h$ are parameters. $\mathbf{b}_h, \mathbf{b}_p$ are bias terms and $\mathbf{p}$ is the probability of the final answer.

2.5 A Variant of CVA

The previous introduced stacked attention mechanism applies channel-wise attention before spatial attention. Given an initial question feature $\mathbf{Q}$ and visual region features $\mathbf{V}$, we adopt a channel-wise attention $\mathbf{A}_c$ to compute the channel-wise attention weights β for obtaining a channel-wised weighted object region features $\mathbf{V}^c$. Next, we apply the object region based spatial attention $\mathbf{A}_s$ by taking $\mathbf{V}^c$ as inputs to obtain region spatial weights η. The pipeline of this framework can be summarized as follows:

$$\beta = \mathbf{A}_c\left(\overline{\mathbf{U}}, \mathbf{Q}\right) \quad (21)$$

$$\mathbf{V}^c = \mathbf{f}_c\left(\beta, \overline{\mathbf{U}}\right) \quad (22)$$

$$\eta = \mathbf{A}_s\left(\mathbf{V}^c, \mathbf{Q}\right) \quad (23)$$

$$\mathbf{V}^s = \mathbf{f}_s\left(\eta, \mathbf{V}^c\right) \quad (24)$$

In order to further study the effect of the oder of channel-wise and spatial attentions, we propose an CVA variant which exchange the order of two attentions by firstly applying spatial attention $\mathbf{A}_s$ and then following by a channel-wise attention $\mathbf{A}_c$. This pipeline can be summarized as follows:

$$\eta = \mathbf{A}_s\left(\mathbf{V}, \mathbf{Q}\right) \quad (25)$$

$$\mathbf{V}^s = \mathbf{f}_s\left(\eta, \mathbf{V}\right) \quad (26)$$

$$\beta = \mathbf{A}_c\left(\mathbf{U}^s, \mathbf{Q}\right) \quad (27)$$

$$\mathbf{V}^c = \mathbf{f}_c\left(\beta, \mathbf{V}^s\right) \quad (28)$$

where $\mathbf{U}^s$ is obtained by reshape $\mathbf{V}^s$, seen channel-wise reshape operation. Further more, $\mathbf{V}^c$ and $\mathbf{Q}$ are utilized to predict the final answer. For simplicity, we name this pipeline as CVA-V.

3 Experiments

3.1 Datasets

We evaluate our proposed model on three public image QA datasets: the COCO-QA dataset, the VQA dataset and Visual7W dataset.

COCO-QA dataset. This dataset [Ren *et al.*, 2015] is proposed to enable training large complex models due to the reason that the previous DAQUAR dataset only contains approximately 1,500 images and 7,000 questions on 37 common object classes. Therefore, COCO-QA is created to produce a much larger number of QA pairs and a more evenly distributed answers based on the MS-COCO dataset. It contains in total 117,684 samples with 78,736 as training and 38,948 as testing. There are 23.29% and 18.7% overlap in training questions and training question-answer pairs. COCO-QA consists of four question categories: *Object* (54,992 Training vs 27,206 Testing), *Number* (5,885 vs 2,755), *Color* (13,059 vs 6,509) and *Location* (4,800 vs 2,478). In addition, all answers in this dataset are a single word.

Methods	Y/N	Num.	Other	All
CA	80.55	39.10	52.45	62.54
RA	83.41	39.01	55.91	65.37
CVA	**83.73**	**40.91**	**56.36**	**65.92**
R-CVA	83.39	40.89	55.86	65.54

Table 1: Ablation study on the VQA test-dev dataset

VQA dataset. It is a large-scale dataset presenting both open-ended answering task requiring a free-form response and multiple-choice task requiring an approach to pick from a predefined list of possible answers. Specifically, 204,721 real images (123,287 training and validation vs 81,434 testing) are collected from the newly-released Microsoft Common Objects in Context (MS COCO) dataset. For this VQA dataset challenge, it contains two test categories: 1) test-dev for debugging and validation purpose; and 2) test standard indicates the 'standard' test data for the VQA competition. More specifically, we use 248,349 training questions, 121,512 validation questions and 244,302 test questions. The question types include *Y/N* 38.37%, *Number* 12.31% and *Other*. Compared with COCO-QA, the answer form is diversity with an answer containing one, two or three words respectively being 89.32%, 6.91%, and 2.74%. In addition, for each image, three questions are collected and each question is answered by ten subjects alone with confidence. Following [Hyeonseob Nam and Kim, 2017], we set th number of possible answer for VQA as 2,000.

Visual7W. This dataset is collected recently by Zhu *et al* [Zhu *et al.*, 2016] with 327,939 QA pairs on 47,300 COCO images, which is a subset of Visual Genome image dataset. With the Amazon Mechanical Turk (AMT), they collected 1,311,756 human-generated multiple-choices and 561,459 object groundings from 36,579 categories. For each QA pair, it has four human-generated multiple-choices and only one of them is correct. In addition, for Visual7W, it consists of seven types of question including *what*,*where*,*when*, *who*,*why*,*how* and *which*. The first six type of questions are proposed to examine the capability of a model of visual understanding. Compared with VQA, Visual7W contains rich questions and longer answers. Moreover, Visual7W establishes an explicit link between QA pairs and image regions by providing complete grounding annotations and providing diverse question to acquire detailed visual information. Following [Yu *et al.*, 2017], we only test our model in the settings of multiple-choices.

3.2 Evaluation Metrics

The VQA task is usually regarded as a multi-class classification problem, and thus accuracy is an important evaluation metric for evaluating the performance of VQA models. Following [Antol *et al.*, 2015], we use the following equation to

Approach	test-dev					test-standard				
	Open Ended				MC	Open Ended				MC
	Y/N	Num	Other	All	All	Y/N	Num	Other	All	All
LSTM Q +I	78.9	35.2	36.4	53.7	57.2	79.0	35.6	36.8	54.1	57.8
deeper +norm	80.5	36.8	43.1	57.8	62.7	80.6	36.5	43.7	58.2	63.1
DPPnet	80.7	37.2	41.7	57.2	-	-	-	-	58.9	-
SAN	79.3	36.6	46.1	58.7	-	-	-	-	59.5	-
FDA	81.1	36.2	45.8	59.2	-	-	-	-	60.4	-
DMN+	80.5	36.8	48.3	60.3	-	-	-	-	-	-
MCB	81.2	35.1	49.3	60.8	65.4	-	-	-	-	-
MCB-7	83.4	39.8	58.5	66.7	70.2	83.2	39.5	58.0	66.5	70.1
Our CVA	**83.73**	**40.91**	**56.36**	**65.92**	**70.3**	**83.79**	**40.41**	**56.77**	**66.20**	**70.41**
AC	79.8	36.8	43.1	57.5	-	79.7	36.0	43.4	57.6	-
ACK	81.0	38.4	45.2	59.2	-	81.1	37.1	45.8	59.4	-
HieCoAtt	79.7	38.7	51.7	61.8	65.8	-	-	-	62.1	66.1
DAN	83.0	39.1	53.9	64.3	69.1	82.8	39.1	54.0	64.2	69.0
MLAN	82.9	39.2	52.8	63.7	68.9	-	-	-	-	-

Table 2: Comparison results on VQA dataset. According to different attention mechanisms, all the approaches are divided into five categories and each row represents one category. Row One indicates *No Attention*. Row Two utilizes *Visual Attention only*. Row Three applies *Semantic Attention*. Row Four includes both *Visual and Question Attentions*. Row Five applies both *Semantic and Visual Attentions*.

compute the classification accuracy:

$$Acc(ans) = min\left\{\frac{\#humans\ that\ said\ ans}{3}, 1\right\} \quad (29)$$

where ans is the answer predicted by a VQA model.

In addition, for MS COCO dataset, we also report the performance in terms of the Wu-Palmer similarity (WUPS), which accounts for word-level ambiguities in the answer words. The equation is defined as [Malinowski *et al.*, 2015] and it contains a thresholded taxonomy-based Wu-Palmer similarity parameter. For COCO-QA, we report WUPS at two extremes, 0.0 and 0.9.

3.3 Implementation Details

For extracting visual object features, we first integrate Faster R-CNN with ResNet-101 retrained on the ImageNet dataset by following an image captioning approach [Anderson *et al.*, 2017] and then select top 36 ($k = 36$) object regions and each region is represented as 2,048 dimensional features. For sentence encoding, a pre-trained GloVe word embedding of dimension (300) and a single layer GUR are utilized. In addition, the dimension of every hidden layer including GRU, attention models and the final joint feature embedding is set as 1,024.

In our experiments, our models are trained with Adam. The batch size is set to 256, and the epoch is set as 30. More specifically, gradient clipping technology and dropout are exploited in training.

3.4 Ablation Study

For VQA challenge, it contains the test-dev, which is proposed to debug and validate VQA models, thus VQA competition evaluation server allows for unlimited submission. In this section, we perform ablation study on the VQA dataset to qualify the role of each component in our model. Specifically, we re-train our approach by ablating certain components: 1) channel-wise attention only (CA); 2) object region attention only (RA); 3) our stacked attention with both channel-wise and region-based attention (CVA); and 4) reversed stacked attention (R-CVA) by changing the order of channel-wise and region-based attention to test whether their order effect the VQA performance.

The experimental results are shown in Tab.1 (test-standard is not recommended to be used for VQA ablation study). From the experimental results, we can see that object region attention alone performs better than channel-wise attention only on the *Y/N*, *Other* and *All* with an increase of 2.86%, 3.46% and 2.83% respectively. In terms of *Number*, channel-wise attention performs slightly better. By stacking those two attention into a VQA model in any order, we find that the stacked attention models improve the performance of VQA, especially for *Number* by approximately 1.9%. In addition, changing order would slightly effect the performance.

3.5 Comparing on the VQA dataset

Compared Methods. We compare our opposed CVA with the state-of-the-art VQA approaches, which can be divided into four categories:1) No attention approaches (LSTM Q+I [Antol *et al.*, 2015] and deeper+norm [Antol *et al.*, 2015] and DPPnet [Noh *et al.*, 2016]); 2) visual attention based methods (SAN [Yang *et al.*, 2016], FDA [Ilievski *et al.*, 2016], DMN+ [Xiong *et al.*, 2016], MCB+Att. [Fukui *et al.*, 2016] and MCB-7 (ensemble of 7 Att. models) [Fukui *et al.*, 2016]); 3) Utilizing high-level concepts as visual features or semantic attention based methods (AC [Wu *et al.*, 2016] and ACK [Wu *et al.*, 2016]); 4) methods with both image attention and question attention (HieCoAtt [Lu *et al.*, 2016] and DAN [Hyeonseob Nam and Kim, 2017]) and 5) jointly learning semantic attention and visual attention (MLAN (ResNet) [Yu *et al.*, 2017]). Our proposed method CVA belongs to the second category, visual attention only.

Results. The experimental results are shown in Tab.2. We have the following observations: CVA obtains the best 'all'

Q:What kind of cake is it?
Our Prediction: chocolate
Human Answer: Chocolate cake

Q:What is this person laying on?
Our Prediction: bench
Human Answer: bench

Q:Is that a fuel efficient truck?
Our Prediction: no
Human Answer: no

Q:Is the laptop turned on?
Our Prediction: yes
Human Answer: yes

Figure 3: Four qualitative results from visual question answering.

Method	All	Obj.	Num.	Color	Loc.	WUPS0.9	WUPS0.0
2VIS+BLSTM	55.09	58.19	44.79	49.53	47.34	65.34	88.64
IMG-CNN	58.40	-	-	-	-	68.50	89.67
DDPnet	61.60	-	-	-	-	70.84	90.61
SAN	61.60	65.40	48.60	57.90	54.00	71.60	90.90
QRU	62.50	65.06	46.90	60.50	56.99	72.58	91.62
HieCoAtt	65.40	68.00	**51.00**	62.90	58.80	75.10	92.00
CVA	**67.51**	**69.55**	50.76	**68.96**	**59.93**	**76.70**	**92.41**

Table 3: Evalution results by our proposed method and compared methods on the COCO QA dataset.

Methods	Wht.	Whr.	Whn.	Who	Why	How	Avg
LSTM-Att	51.5	57.0	75.0	59.5	55.5	49.8	54.3
MCB+Att	60.3	70.4	79.5	69.2	58.2	51.1	62.2
MLAN	60.5	**71.2**	79.6	69.4	58.0	50.8	62.4
CVA	**64.8**	59.4	**80.1**	**70.0**	**65.2**	**55.7**	**63.8**

Table 4: Evaluation results on Visual7W dataset.

accuracies on test dev (Open-end 65.92% vs multiple-choice 70.3%) and test-standard (66.20% vs 70.41%). Our method belongs to the second category (visual attention only). Compared with the second category methods, the improvement over the best approach MCB is significant, by 5.12% (test-dev, open-ended, all) and 4.9% (test-dev, multiple-choice, all). In addition, we also report the results of MCB-7 (ensemble of 7 Att. models), although it is not comparable because each model in ensemble of 7 models uses MCB with attention. Comparing with the third category, again our approach outperforms the best approach ACK, especially with an increase of 6.8% in terms of test-standard opened-end all. Although HieCoAtt and DAN integrate both visual and question attentions, our CVA performs better. For standard-test dataset, tt surpass the DAN by 2.0% (in open-ended, all) and 1.14% (in multiple-choice, all), respectively. In addition, compared with MLAN involving both semantic attention and visual attention, our approach is also better, with a clear performance gap on the test-dev dataset. The results in Tab.2 clearly demonstrate the advantage of our method.

3.6 Comparing on the COCO-QA dataset

Compared Methods. In this section, we compare our methods with the state-of-the art approaches on the COCO-QA dataset. We compare it with 2-VIS+BLSTM [Ren *et al.*, 2015], IMG-CNN [Ma *et al.*, 2016], DDPnet[Noh *et al.*, 2016], SAN(2, CNN)[Yang *et al.*, 2016], QRU [Li and Jia, 2016] and HieCoAtt [Lu *et al.*, 2016]. In addition, Fig.3 demonstrates some qualitative results from VQA.

Results. The experimental results are demonstrated in Tab. 3. Our CVA achieves the highest performance with an accuracy of 67.51% on all and a WUPS 0.9 of 76.7%. Compared with the best counterpart HioCoAtt and the second best counterpart QRU, the improvement of accuracy is 2.11% and 5.01% on all. Specifically, for *Object*, *Color* and *Loc.* three types questions, CVA increases the accuracy to 69.55% and 68.96%, and 59.93%, respectively. The table indicates that the *Number* type question performs worst in all methods, with approximately 10 % to 19% lower than other types. This might be caused by the unbalanced training dataset. The number of training samples of *Object* type is 54,992, which is nine times more than the number of training samples for *Number*.

3.7 Comparing on the Visual7w dataset

In this section, we further assess our model on the recent released dataset Visual7w. For the experiments, we compare our CVA with previous work LSTM-Att. [Zhu *et al.*, 2016], MCB+Att. [Fukui *et al.*, 2016] and MLAN [Yu *et al.*, 2017] and the experimental results are shown in Tab.3. Both LSTM-Att. and MLAN are competitive. However, compared with LSTM-Att., MLAN utilizes a much lower fusion methods (2400-dimension vs 16,000-dimension)[Yu *et al.*, 2017]. Tab.3 shows that our CVA achieves the highest scores. In particular, the MLAN employs both visual attention and text attention, while our CVA only exploits the visual attention.

4 Conclusion

In this paper, we propose a novel cubic visual attention network for visual question answering task. CVA takes the full advantage of characteristics of CNN to obtain visual channel-wise features representing semantic attributes and object region based visual features representing rich semantic information to support visual question answering and it achieves the state-of-the-art on three public standard datasets across various question types, such as multiple-choices and open-ended questions. The contribution of CVA is not only provide a powerful VQA model, but also a better mechanism to understand the visual information for predicting answers.

Acknowledgments

This work is supported by the Fundamental Research Funds for the Central Universities (Grant No. ZYGX2014J063, No. ZYGX2014Z007) and the National Natural Science Foundation of China (Grant No. 61772116, No. 61502080, No. 61632007, No. 61602049).

References

[Anderson *et al.*, 2017] 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. *arXiv preprint arXiv:1707.07998*, 2017.

[Antol *et al.*, 2015] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. VQA: visual question answering. In *ICCV*, pages 2425–2433, 2015.

[Chen *et al.*, 2017] Long Chen, Hanwang Zhang, Jun Xiao, Liqiang Nie, Jian Shao, Wei Liu, and Tat-Seng Chua. Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning. In *CVPR*, 2017.

[Fukui *et al.*, 2016] 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. In *EMNLP*, pages 457–468, 2016.

[Gao *et al.*, 2015] Haoyuan Gao, Junhua Mao, Jie Zhou, Zhiheng Huang, Lei Wang, and Wei Xu. Are you talking to a machine? dataset and methods for multilingual image question. In *NIPS*, pages 2296–2304, 2015.

[Gao *et al.*, 2017] Lianli Gao, Zhao Guo, Hanwang Zhang, Xing Xu, and Heng Tao Shen. Video captioning with attention-based LSTM and semantic consistency. *IEEE Trans. Multimedia*, 19(9):2045–2055, 2017.

[Goyal *et al.*, 2017] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the V in VQA matter: Elevating the role of image understanding in visual question answering. In *CVPR*, pages 6325–6334, 2017.

[He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[Hyeonseob Nam and Kim, 2017] Jung-Woo Ha Hyeonseob Nam and Jeonghee Kim. Dual attention networks for multimodal reasoning and matching. In *CVPR*, 2017.

[Ilievski *et al.*, 2016] Ilija Ilievski, Shuicheng Yan, and Jiashi Feng. A focused dynamic attention model for visual question answering. *CoRR*, abs/1604.01485, 2016.

[Li and Jia, 2016] Ruiyu Li and Jiaya Jia. Visual question answering with question representation update (QRU). In *NIPS*, pages 4655–4663, 2016.

[Lu *et al.*, 2016] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Hierarchical question-image co-attention for visual question answering. In *NIPS*, 2016.

[Lu *et al.*, 2018] Pan Lu, Hongsheng Li, Wei Zhang, Jianyong Wang, and Xiaogang Wang. Co-attending free-form regions and detections with multi-modal multiplicative feature embedding for visual question answering. In *AAAI*, 2018.

[Ma *et al.*, 2016] Lin Ma, Zhengdong Lu, and Hang Li. Learning to answer questions from image using convolutional neural network. In *AAAI*, pages 3567–3573, 2016.

[Malinowski *et al.*, 2015] Mateusz Malinowski, Marcus Rohrbach, and Mario Fritz. Ask your neurons: A neural-based approach to answering questions about images. In *ICCV*, pages 1–9, 2015.

[Noh *et al.*, 2016] Hyeonwoo Noh, Paul Hongsuck Seo, and Bohyung Han. Image question answering using convolutional neural network with dynamic parameter prediction. In *CVPR*, pages 30–38, 2016.

[Ren *et al.*, 2015] Mengye Ren, Ryan Kiros, and Richard S. Zemel. Exploring models and data for image question answering. In *NIPS*, pages 2953–2961, 2015.

[Ren *et al.*, 2017] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(6):1137–1149, 2017.

[Shih *et al.*, 2016] Kevin J. Shih, Saurabh Singh, and Derek Hoiem. Where to look: Focus regions for visual question answering. In *CVPR*, pages 4613–4621, 2016.

[Song *et al.*, 2017] Jingkuan Song, Lianli Gao, Zhao Guo, Wu Liu, Dongxiang Zhang, and Heng Tao Shen. Hierarchical LSTM with adjusted temporal attention for video captioning. In *IJCAI*, pages 2737–2743, 2017.

[Wang *et al.*, 2017] Peng Wang, Qi Wu, Chunhua Shen, and Anton van den Hengel. The vqa-machine: Learning how to use existing vision algorithms to answer new questions. In *CVPR*, pages 3909–3918, 2017.

[Wu *et al.*, 2016] Qi Wu, Peng Wang, Chunhua Shen, Anthony R. Dick, and Anton van den Hengel. Ask me anything: Free-form visual question answering based on knowledge from external sources. In *CVPR*, pages 4622–4630, 2016.

[Xiong *et al.*, 2016] Caiming Xiong, Stephen Merity, and Richard Socher. Dynamic memory networks for visual and textual question answering. In *ICML*, pages 2397–2406, 2016.

[Xu and Saenko, 2016] Huijuan Xu and Kate Saenko. Ask, attend and answer: Exploring question-guided spatial attention for visual question answering. In *ECCV*, 2016.

[Yang *et al.*, 2016] Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng, and Alexander J. Smola. Stacked attention networks for image question answering. In *CVPR*, 2016.

[Yu *et al.*, 2017] Dongfei Yu, Jianlong Fu, Tao Mei, and Yong Rui. Multi-level attention networks for visual question answering. In *CVPR*, pages 4187–4195, 2017.

[Zhu *et al.*, 2016] Yuke Zhu, Oliver Groth, Michael S. Bernstein, and Li Fei-Fei. Visual7w: Grounded question answering in images. In *CVPR*, pages 4995–5004, 2016.