DUALNET: DOMAIN-INVARIANT NETWORK FOR VISUAL QUESTION ANSWERING

*Kuniaki Saito, Andrew Shin, Yoshitaka Ushiku, Tatsuya Harada**

The University of Tokyo
7-3-1 Hongo Bunkyo-ku, Tokyo, Japan
{k-saito,andrew,ushiku,harada}@mi.t.u-tokyo.ac.jp

ABSTRACT

Visual question answering (VQA) tasks use two types of images: abstract (illustrations) and real. Domain-specific differences exist between the two types of images with respect to "objectness," "texture," and "color." Therefore, achieving similar performance by applying methods developed for real images to abstract images, and vice versa, is difficult. This is a critical problem in VQA, because image features are crucial clues for correctly answering the questions about the images. However, an effective, domain-invariant method can provide insight into the high-level reasoning required for VQA. We thus propose a method called DualNet that demonstrates performance that is invariant to the differences in real and abstract scene domains. Experimental results show that DualNet outperforms state-of-the-art methods, especially for the abstract images category.

Index Terms— Visual question answering, Multimodal learning, Abstract images

1. INTRODUCTION

Multimodal learning incorporating vision and language has become a popular research area in artificial intelligence. Image captioning and visual question answering (VQA) are tasks in which an artificial intelligence agent behaves as if it precisely understands the content of natural images and languages [1, 2]. In both tasks, an artificial intelligence agent needs to understand the relationship between image representations and sentence representations correctly. Particularly, in VQA, we need to construct a model that understands the question, locates or classifies the objects/scenes mentioned in the question, and generates appropriate answers.

VQA comprises two task categories: one deals with real images and the other handles abstract images [3]. All methods that involve real images, to the best of our knowledge, use convolutional neural networks (CNNs) [4] trained on ImageNet [5] to extract the features of real images. Conversely, for abstract images (illustrations), different techniques are employed, because objects in these images differ from those in real images with respect to their objectness, texture, and color. Furthermore, the number of abstract images is considerably smaller than that of real images because of their collection cost. For these reasons, it has widely been considered that successful methods for real images cannot be directly ported to the abstract scenes domain. Existing research has not applied the same methods to both real and abstract images. However, a method that is useful for both real and abstract images would provide insight into the domain-invariant high-level reasoning required for VQA.

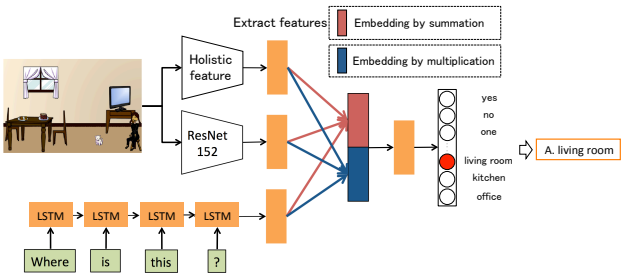


Fig. 1: Our proposed network: DualNet for abstract images. We conducted both elementwise multiplication and summation to combine multimodal features and achieved state-of-the-art performance on the VQA abstract image dataset.

We focus on the commonality in operations for real and abstract images in VQA, namely the combination of sentence and image representations for constructing multimodal features. This operation is essential because, given the question representation, it extracts image information that is useful for the reply. Therefore, building a multimodal feature from domain-invariant operations is important. In this paper, we introduce an effective VQA network architecture that includes a simple operation applicable to both real and abstract images, by performing separate summation and multiplication operations on input features to form a common embedding space. Our method, DualNet, results in good performance in both domains; our method achieves state-of-the-art performance on a VQA abstract scene dataset in particular. In Section 4, we demonstrate DualNet's clear advantage over methods that

*This work was funded by the ImPACT Program of Council for Science, Technology and Innovation (Cabinet Office, Government of Japan) and supported by JST CREST Grant Number JPMJCR1403, Japan.

perform only one operation, including many recent state-of-the-art methods. The contribution of this work is as follows:

- We propose an effective VQA network architecture that performs both summation and multiplication when combining features and is shown by experiments to be useful across domains.
- This method outperforms others for abstract VQA scenes and can serve as a new benchmark.

2. RELATED WORK

2.1. VQA for Real Images

[3] introduced a large-scale dataset for the VQA Challenge, in addition to a baseline approach in which image and question features are embedded to common space at the last state prior to classification. Most methods used in VQA tasks essentially follow the pipeline of this method [6, 7], in which multimodal features are constructed from image and question representations to output a correct label. Many recent papers reporting competitive results have relied heavily on various attention mechanisms. [7] introduced stacked attention networks (SAN), which rely on the semantic representation of each question to search for relevant regions in the image. [8] proposed multimodal compact bilinear pooling for VQA with an attention mechanism to fuse features. Unlike most of the work, which concentrated on extracting information in regions of interest, we focus on building rich features when fusing image features and sentence features without spatial attention, as features of abstract images do not contain spatial information, in contrast with CNN features, which are discussed below.

2.2. VQA for Abstract Scenes

The VQA task for real images requires the use of a CNN to extract representations from complex or noisy images. Conversely, VQA tasks for abstract images require capturing a high-level concept from simple images because abstract scene images are constructed from limited clipart patterns. [9] converted the questions to a tuple containing essential clues to the visual concept of the images. Each tuple (P, R, S) consists of a primary object P, a secondary object S, and their relation R. As mentioned in Section 1, current VQA methods that use CNN features trained on real images perform poorly in the abstract category. We directly implemented the VQA baseline method provided by [10] on abstract images with ResNet152 [11] features. The total accuracy was 62.7%. On the other hand, when using holistic features, the accuracy was 65.0% [9]. This difference shows that the features of current state-of-the art CNNs are less discriminative than the holistic features for VQA. Therefore, in this paper, we propose to use a method that does not depend on the spatial information of CNN features to construct a commonly effective method for real and abstract images.

3. METHOD

In this section, we describe the details of our proposed network architecture, *DualNet*, and demonstrate its success in VQA tasks for both real and abstract images.

3.1. Insight and Motivation

It is necessary to determine how to combine visual features with sentence features, because a network cannot answer correctly unless it has sufficient knowledge of what is asked and which features are necessary to ascertain the answer. [3] employs elementwise multiplication for solving this problem. Another option for fusing the features is element-wise summation. Previous studies have examined and compared network behaviors based on the fusing mechanism, finding that network performance varies according to the method by which the image and sentence features are fused [12]. This indicates that, even with nonlinear networks, information can vary in accordance with the fusing method. Most architectures used only one method to fuse features; for example, summation or multiplication only.

Multiplication of two or three features is a nonlinear conversion of different features. In contrast, summation is a linear conversion that is easy to represent neural networks (expressed by multiplication between an aligned identity matrix and multimodal features). Importantly, expressing the projection of the multiplication from summation is difficult, for example, representing xy from $x + y$ and vice versa. Therefore, the information contained in features from multiplication and summation will be substantially different. From this insight, we propose a simple but effective network structure for VQA. We propose to integrate elementwise summation and elementwise multiplication; in other words, we implement a simple polynomial function. We obtain equations in our network that are in a similar form to the following:

$$(1 + x_1 + x_2 + x_3 + \cdots x_d)(1 + y_1 + y_2 + y_3 + \cdots y_d). \quad (1)$$

x_i and y_i denote the i-th dimension of an image and a sentence feature respectively. Unfortunately, we cannot express the product of different dimensions, such as $x_1 y_2$. However, other forms, such as $x_1 + y_2$ or $x_1 y_1 + x_3 y_3$, can be constructed. Moreover, we propose to use different types of image features to fully take advantage of the variety of information present. For example, holistic features used in abstract scenes [9] display completely different characteristics from CNN features. Likewise, for CNN features, different network structures result in extracted features with different characteristics. Therefore, DualNet benefits from exploiting a combination of features from different networks and methods.

3.2. Network Architecture

We now detail the theoretical background of our network. For clarity, we will skip the notation for bias parameters in the following equations and consider a situation in which two types

of image features are used. The VQA task is formulated as

$$\hat{a} = \arg\max_{a \in R} p(a|\boldsymbol{i}_1, \boldsymbol{i}_2, \boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3, \cdots \boldsymbol{x}_t; \boldsymbol{\theta}), \tag{2}$$

where $\hat{a}$ denotes the output answer, $\boldsymbol{i}_1, \boldsymbol{i}_2$ denotes image features, $\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3, \cdots \boldsymbol{x}_t$ denotes a one-hot vector of question words, R denotes possible answers, and $\boldsymbol{\theta}$ denotes model parameters.

$$\boldsymbol{q} = \mathrm{LSTM}(\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3, \cdots \boldsymbol{x}_t) \tag{3}$$

First, we input the one-hot word vector to obtain question vector $\boldsymbol{q}$ from the last hidden layer of the LSTM [13]. We used the same activation function as in [13].

$$\boldsymbol{i}_{M_1} = \tanh(W_{M_1}\boldsymbol{i}_1) \tag{4}$$
$$\boldsymbol{i}_{M_2} = \tanh(W_{M_2}\boldsymbol{i}_2) \tag{5}$$
$$\boldsymbol{q}_M = \tanh(W_{M_q}\boldsymbol{q}) \tag{6}$$
$$\boldsymbol{f}_M = \boldsymbol{i}_{M_1} \circ \boldsymbol{i}_{M_2} \circ \boldsymbol{q}_M \tag{7}$$

W_{M_1}, W_{M_2} and W_{M_q} encode different features into the space with the same dimension. Eqs. (4) - (7) correspond to the fusing of image and text features by multiplication. $\circ$ refers to elementwise multiplication.

$$\boldsymbol{i}_{S_1} = \tanh(W_{S_1}\boldsymbol{i}_1) \tag{8}$$
$$\boldsymbol{i}_{S_2} = \tanh(W_{S_2}\boldsymbol{i}_2) \tag{9}$$
$$\boldsymbol{q}_S = \tanh(W_{S_q}\boldsymbol{q}) \tag{10}$$
$$\boldsymbol{f}_S = \boldsymbol{i}_{S_1} + \boldsymbol{i}_{S_2} + \boldsymbol{q}_S \tag{11}$$

W_{S_1}, W_{S_2} and W_{S_q} encode different features into the space with the same dimension. Eqs. (8) - (11) correspond to the fusing of features by summation. Our network does not share weights between multiplication and summation because we expect each operation to extract different information.

$$\boldsymbol{f} = \mathrm{Concat}(\boldsymbol{f}_M, \boldsymbol{f}_S) \tag{12}$$
$$p(a|\boldsymbol{i}_1, \boldsymbol{i}_2, \boldsymbol{x}; \boldsymbol{\theta}) = \boldsymbol{w}_{f_2}\tanh(W_{f_1}\boldsymbol{f}) \tag{13}$$

We concatenate the features from elementwise multiplication and elementwise summation, and then input a fully-connected layer with weight, W_{f_1}, to obtain the prediction. In this example, we have shown a case using two kinds of image features. We can alter the number of image features based on need, but the overall workflow remains the same.

For real images, we used three kinds of image features: L2-normalized features from the first fully-connected layer (fc6) of VGG-19 [4] trained on ImageNet [5], and the uppermost fully-connected layers from ResNet-152 and ResNet-101 [11]. The proposed model architecture for abstract images is described in Fig. 1. It uses the L2-normalized holistic feature and the fully-connected layer of ResNet-152.

4. EXPERIMENTS

In this section, we describe and discuss the results from experiments using our DualNet architecture on a VQA dataset. There are two tasks on the VQA dataset: open-ended and multiple choice. In the open-ended task, the answer must be determined solely from the question and the image. In the multiple choice task, 18 candidate answers per question are given. We evaluated our method on both the tasks. We used the same metric as used in other works. Multiple ground truth answers are attached to one question by some researchers. If we obtain one answer for one question, we calculate the accuracy by the following metric. $accuracy = min(\text{the number of humans that provided that answer})/3, 1)$

4.1. Real Images

The VQA dataset consists of 82,783 training images, 40,504 validation images, and 81,434 test images. Three questions are attached to each image. We evaluated our model using a subset of the data, called the test-dev split on the VQA evaluation server. In this experiment, we used both the training and validation splits for training. As the number of test submissions for the complete test split is limited, evaluation on the complete test split was restricted to selected key methods. We tuned the parameter using the split of test-dev. We set the learning rate of all methods (summation, multiplication, and proposed method) as 0.0004. We tuned it on test-dev and all methods showed better performance on that learning rate. The LSTM network in our model consists of two layers with 512 hidden units. We used the 2,000 most frequent answers as labels and relied on rms-prop to optimize our model. The batch size was 300. All methods show slightly different performance depending on the parameters, but the difference is very small. The margin of the score between the proposed method and baseline methods is more significant than what can be expected by simply tuning hyperparameters.

Model performance changed slightly based on the dimension of common space. We present the result of 512 dimensions as a single DualNet's result. For the ensemble of DualNets, we set the common space dimensions differently for each unit. We changed the common space dimensions from 500 to 3000 for each DualNet unit in our ensemble, which consisted of 19 DualNet units. We tuned the weight for each unit in the ensemble based on its result on test-dev splits. As we increased the number of units, we could observe improvements in accuracy on test-dev, and the accuracy saturated at around 19 units.

As baseline methods, we used networks that implemented only summation or multiplication. For fair comparison, we set the dimension of common space as 1,024 in summation- and multiplication-only networks, because DualNet can be considered to have a 1,024-dimensional common space. We further compare our method with ensemble of summation- and multiplication-only networks. We also show the result of our method that uses only VGG network features.

Table 1: Performance on a test-dev split of real images. Sum only means using summation to combine image and sentence features; likewise, Mul only uses only multiplication. DualNet* has two fully-connected layers after fusing features.

	Open-Ended				Multiple Choice			
	All	Y/N	Num	Others	All	Y/N	Num	Others
deeper LSTM Q+norm [10]	57.75	80.5	36.8	43.1	62.70	80.5	38.2	53.0
SAN [7]	58.70	79.3	36.6	46.1	-	-	-	-
DMN+[14]	60.30	80.5	36.8	48.3	-	-	-	-
MCB[8] (w/o attention)	60.8	81.2	35.1	49.3	-	-	-	
MCB[8] (with attention)	**64.2**	83.4	39.8	58.5	-	-	-	
Sum only	58.78	78.6	36.0	47.0	63.64	78.6	37.7	56.7
Mul only	59.29	80.7	37.1	46.0	64.74	80.8	39.0	56.9
Ensemble of Sum and Mul	60.27	80.7	37.4	47.9	63.77	78.7	37.9	56.9
DualNet (only VGG)	59.03	80.7	36.6	45.6	64.52	80.8	38.9	56.4
DualNet	60.34	81.3	37.0	47.7	65.54	81.4	39.1	58.0
DualNet*	60.47	81.0	37.1	48.2	65.80	80.8	39.8	58.9
DualNet (ensembled)	**61.47**	**82.0**	**37.9**	**49.2**	**66.66**	**82.1**	**39.8**	**59.5**
Evaluation on test-std split								
	Open-Ended				Multiple Choice			
	All	Y/N	Num	Others	All	Y/N	Num	Others
deeper LSTM Q+norm [10]	58.2	80.6	36.5	43.7	63.09	80.6	37.7	53.6
AYN [12]	58.43	78.2	36.3	46.3	-	-	-	-
SAN [7]	58.90	-	-	-	-	-	-	-
DMN+ [14]	60.36	80.4	36.8	48.3	-	-	-	-
MRN [15]	61.84	82.4	38.2	49.4	66.33	82.4	39.6	58.4
MCB[8] (att models ensemble)	66.5	83.2	39.5	58.0	70.1	-	-	
DualNet (ensembled)	61.72	81.9	37.8	49.7	66.72	82.0	39.7	59.6

4.2. Abstract Scene

The abstract scenes category contains 20,000 training images, 10,000 validation images, and 20,000 test images, and each image is accompanied by three questions. Unlike the real image category, there is no test-dev split. We set the learning rate as 0.004. We first tuned hyperparameters on validation split, then trained our model on train and validation splits using the hyperparameters from the beginning.Our proposed model is shown in Fig. 1. We used L2-normalized ResNet152 and holistic features as image features. We used a two-layer LSTM for question encoding and 500 possible answers. For the ensemble model, we combined five models with different common space dimensions.

We compared our method with summation- and multiplication-only networks, as we did for real images. We set the number of hidden units to 1024 when combining features. However, we assume that the number of units can easily affect model performance, since the number of samples is considerably smaller than that of real images. Then, we implemented summation- and multiplication-only networks with 512 units.

5. RESULT AND ANALYSIS

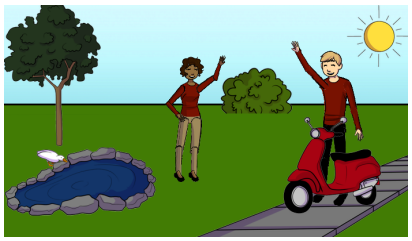
Table 1 shows the results of each method on the test-dev split and on the test-std split. As shown in these tables, the proposed method performed better than Sum only, Mul only, and their ensemble model. We can clearly see the effectiveness of our network structure in comparison with the summation- and multiplication-only networks, which achieved performances of 58.78% and 59.29%, respectively. The performance of the summation network is poorer than that of the multiplication network. However, when combining the two paths, we were able to improve performance up to 60.34%. For both category images, a simple ensemble of two networks results in poorer performance than our method as seen from Table 2, and is obvious in the multiple choice task. This indicates that fusing features during training as in our method is important. Compared with methods such as DMN [14], SAN [7], and FDA [18], which used attention mechanisms, our model achieves better performance. Compared with MCB (w/o attention) [8], the performance of our model is comparable. The implementation of our model is considerably simpler than that of their methods. Although it is true that combining multiple image features improves the performance, we observed that our model achieves performance comparable to that of other methods even when we have only the VGG feature. Fig. 2 shows examples of questions and generated answers in the case of real images, along with the images.

We further analyzed the output from summation and mul-

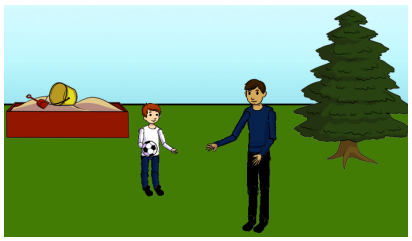
Table 2: Evaluation of each method on abstract scene test data from the VQA test server.

	Open-Ended				Multiple Choice			
	All	Y/N	Num	Others	All	Y/N	Num	Others
holistic feature [9]	65.02	77.5	52.5	56.4	69.21	77.5	52.9	66.7
holistic + vlad + deep[16]	67.39	79.6	57.1	58.2	71.18	79.6	56.2	67.9
MRN [15]	62.56	79.1	51.6	48.9	67.99	79.1	52.6	62.0
Sum 512 units	66.24	78.6	53.1	58.0	-	-	-	-
Mul 512 units	67.87	79.1	57.5	59.7	-	-	-	-
Sum 1024 units	66.87	78.8	53.4	59.2	71.41	78.8	54.1	70.2
Mul 1024 units	68.00	79.5	57.1	59.8	72.65	79.5	57.4	71.2
Ensemble of Sum and Mul	68.80	79.9	57.3	61.3	72.60	79.5	57.9	70.9
DualNet	68.87	80.0	57.9	61.1	73.29	80.0	58.5	71.8
DualNet (ensembled)	**69.73**	**80.7**	**58.8**	**62.1**	**74.02**	**80.8**	**59.2**	**72.4**

(a) **Q**: What is the boy playing with? **A**: teddy bear



(b) **Q**: Are there any animals swimming in the pond? **A**: No



(c) **Q**: How many trees? **A**: 1

Fig. 2: Examples of questions and generated answers for abstract scenes

Table 3: Analysis of network performance inside DualNet. $\boldsymbol{f}_m$ and $\boldsymbol{f}_s$ denote a feature from multiplication and summation respectively. w_m and w_m denote weights on each feature inside DualNet.

	Open-Ended			
	All	Y/N	Num	Others
DualNet $w_m \boldsymbol{f}_m$	42.7	51.3	25.6	39.3
DualNet $w_s \boldsymbol{f}_s$	54.5	77.0	35.5	39.6
DualNet	60.34	81.3	37.0	47.7

tiplication networks inside DualNet, as shown in Table 3. From this result, we observe that summation and multiplication inside DualNet extract completely different and less discriminative representations, but combining them via DualNet clearly improves their performance. It is very interesting to see that the result of yes/no questions in the summation result is lower than chance, which is about 70%, but performance improves 77% to 81% when combined with multiplication. In Fig. 3, we show examples of how each operation evaluates the candidate answers and how DualNet combines them to extract the correct answer. We visualized a feature space using t-SNE [17] in Fig. 4. The embedded feature space seems easily separable by the question type.

As for abstract images, our DualNet method significantly outperformed the other method.Our model performed better than other methods both on open-ended and multiple choice tasks. Although we cannot obtain a good result only by ResNet image features as in Section 2, combining holistic features and image features through our method improved the performance. Our model outperformed the state-of-the-art models by about 2.4%, and the multiplication network performed better than the summation network. From this result, we can say that our simple network structure helps to extract discriminative features.

6. CONCLUSION

We proposed DualNet to efficiently and fully account for discriminative information in images and textual features by performing separate operations for input features and building ensembles with varying dimensions. Experimental results demonstrate that DualNet outperforms many previous state-of-the-art methods, and that it is applicable to both real images and abstract scenes, despite their fundamentally different characteristics. In particular, our model outperformed the previous state-of-the-art methods for abstract scenes. Since our method was able to perform well even without an attention mechanism, an interesting future work would be to examine the combination of DualNet with an attention mechanism.

7. REFERENCES

[1] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan, "Show and tell: A neural image cap-

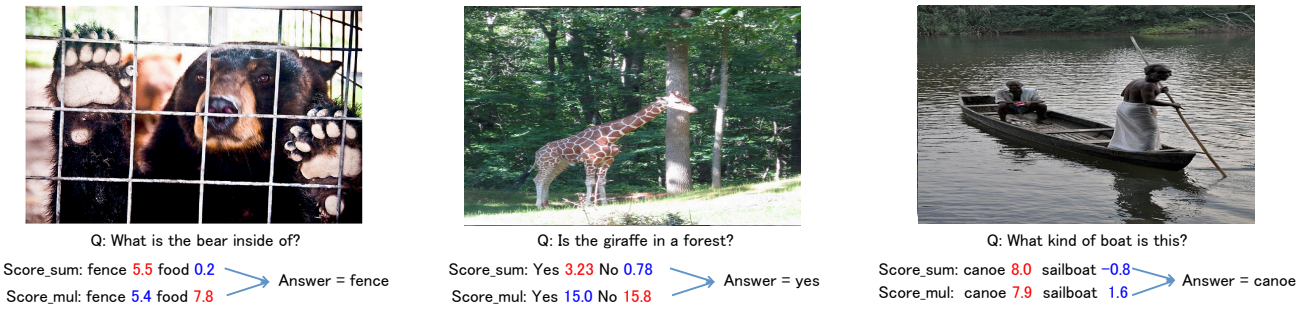
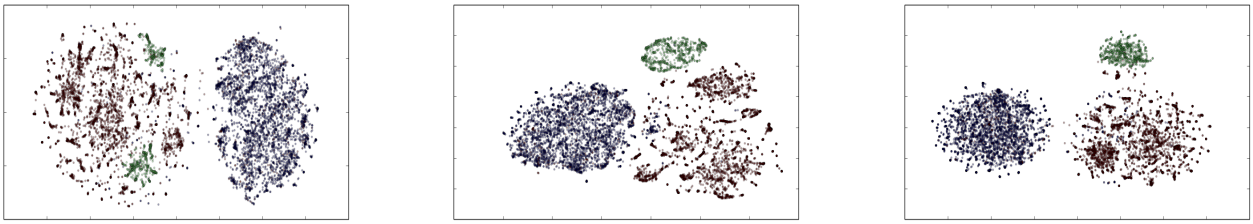


Fig. 3: Contribution of summation and multiplication networks inside DualNet



(a) Features of summation-only network (b) Features of multiplication-only network (c) Features of DualNet

Fig. 4: Visualization of t-SNE embedding [17] from the last fully-connected layer of each network. (c) is from DualNet. (a) and (b) are from summation and multiplication, respectively. **Blue** plots indicate yes/no questions, **red** are other questions, and **green** are numerical questions.

tion generator," in *CVPR*, 2015.

[2] Lin Ma et al., "Learning to answer questions from image using convolutional neural network," in *AAAI*, 2016.

[3] Stanislaw Antol et al., "Vqa: Visual question answering," in *ICCV*, 2015.

[4] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *ICLR*, 2014.

[5] J. Deng et al., "ImageNet: A Large-Scale Hierarchical Image Database," in *CVPR*, 2009.

[6] Kevin J. Shih et al., "Where to look: Focus regions for visual question answering," in *CVPR*, June 2016.

[7] Zichao Yang et al., "Stacked attention networks for image question answering," in *CVPR*, 2016.

[8] Akira Fukui et al., "Multimodal compact bilinear pooling for visual question answering and visual grounding," *arXiv 1606.01847*, 2016.

[9] Peng Zhang et al., "Yin and yang: Balancing and answering binary visual questions," in *CVPR*, 2016.

[10] Jiasen Lu et al., "Deeper lstm and normalized cnn visual question answering model," `https://github.com/VT-vision-lab/VQA_LSTM_CNN`, 2015.

[11] Kaiming He et al., "Deep residual learning for image recognition," in *CVPR*, 2015.

[12] Mateusz Malinowski et al., "Ask your neurons: A deep learning approach to visual question answering," *arXiv 1605.02697*, 2016.

[13] Felix A Gers et al., "Learning to forget: Continual prediction with lstm," *Neural computation*, vol. 12, no. 10, pp. 2451–2471, 2000.

[14] Caiming Xiong et al., "Dynamic memory networks for visual and textual question answering," in *ICML*, 2016.

[15] Jin-Hwa Kim et al., "Multimodal residual learning for visual qa," *arXiv 1606.01455*, 2016.

[16] Kuniaki Saito et al., "Mil-ut presentation on abstract image challenge," `http://visualqa.org/static/slides/MIL_UT_slide.pdf`, 2016, VQA Challenge 2016 Workshop.

[17] Laurens van der Maaten and Geoffrey Hinton, "Visualizing data using t-sne," *JMLR*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[18] Ilija Ilievski et al., "A focused dynamic attention model for visual question answering," *arXiv 1604.01485*, 2016.