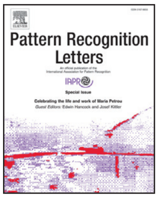
Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec



Dual-decoder transformer network for answer grounding in visual question answering

Liangjun Zhu, Li Peng*, Weinan Zhou, Jielong Yang

Engineering Research Center of Internet of Things Applied Technology, Jiangnan University, Wuxi 214122, China

ARTICLE INFO

Article history:
Received 15 September 2022
Revised 26 February 2023
Accepted 16 April 2023
Available online 18 April 2023

Edited by: Prof. S. Sarkar

Keywords:
Visual question answering
Answer grounding
Dual-decoder transformer

ABSTRACT

Visual Question Answering (VQA) have made stunning advances by exploiting Transformer architecture and large-scale visual-linguistic pretraining. State-of-the-art methods generally require large amounts of data and devices to predict textualized answers and fail to provide visualized evidence of the answers. To mitigate these limitations, we propose a novel dual-decoder Transformer network (DDTN) for predicting the language answer and corresponding vision instance. Specifically, the linguistic features are first embedded by Long Short-Term Memory (LSTM) block and Transformer encoder, which are shared between the Transformer dual-decoder. Then, we introduce object detector to obtain vision region features and grid features for reducing the size and cost of DDTN. These visual features are combined with the linguistic features and are respectively fed into two decoders. Moreover, we design an instance query to guide the fused visual-linguistic features for outputting the instance mask or bounding box. The classification layers aggregate results from decoders and predict answer as well as corresponding instance coordinates at last. Without bells and whistles, DDTN achieves state-of-the-art performance and even competitive to pretraining models on VizWizGround and GQA dataset. *The code is available at* https://github.com/zlj63501/DDTN.

1. Introduction

Visual Question Answering (VQA) plays a crucial role in multimedia interaction, which aims to answer related questions given an image [1–4]. It has substantial values for people with visual impairments to learn about their surroundings [5] or as an interactive diagnosis tool for medical images [6]. General computer vision or natural language processing tasks require the system to perform basic operations. In contrast, The advanced VQA systems are usually equipped with reasoning skills, such as color recognition, orientation determination, and object counting [7]. Therefore, VQA is more challenging than single-modality tasks.

Existing methods have made significant progress in accuracy metrics [8], but there are still some issues to address. Firstly, these systems do not provide sufficient evidence to demonstrate that they answer the question correctly by combining images and text. According to [9], the accuracy is about 90% if the answer to "Do you see a..." is "yes". Secondly, selecting the right word from a predefined list cannot really benefit people with vision impairments. As shown in Fig. 1(a), a textual answer to "what's it?" may not match the expected object. Thirdly, a valuable VQA system should be able to separate the object from its surroundings. When the system is asked, "Could you help me find the horse?", it should provide the picture "horse" along with the corresponding textual answer if the image really exists "horse".

To overcome these limitations, the answer grounding task was proposed in Hudson and Manning [10]. Unlike conventional VQA, it is closer to a practical application for visually impaired people, which aims to require the system to return the visual area corresponding to the answer. Prior tasks only required the area at the level of bounding boxes, and the recent [11] advances this task by requiring models to provide an instance mask for the answer. A common approach is to develop an attention map showing the area. Current state-of-the-art is MAC-Caps [12], which is built on a capsule network. In this method, each object entity or part of a whole is encoded by a capsule matrix. With a routing-by-agreement algorithm, every capsule is given a vote, modeling part-to-whole relationships. Despite high accuracy, the attention map also leads to low location precision. For other pretraining models, most of them cannot output both visual and linguistic modalities. Only a few models [13,14] attempt to accomplish this task, but their training costs are very expensive. DaVI [14] is also the most relevant method to our work. For modeling features of both vision

* Corresponding author.
E-mail addresses: zhuliangjun@stu.jiangnan.edu.com (L. Zhu), pengli@jiangnan.edu.cn (L. Peng), 6201924224@stu.jiangnan.edu.cn (W. Zhou), jyang022@e.ntu.edu.sg (J. Yang).

https://doi.org/10.1016/j.patrec.2023.04.003
0167-8655/© 2023 Published by Elsevier B.V.

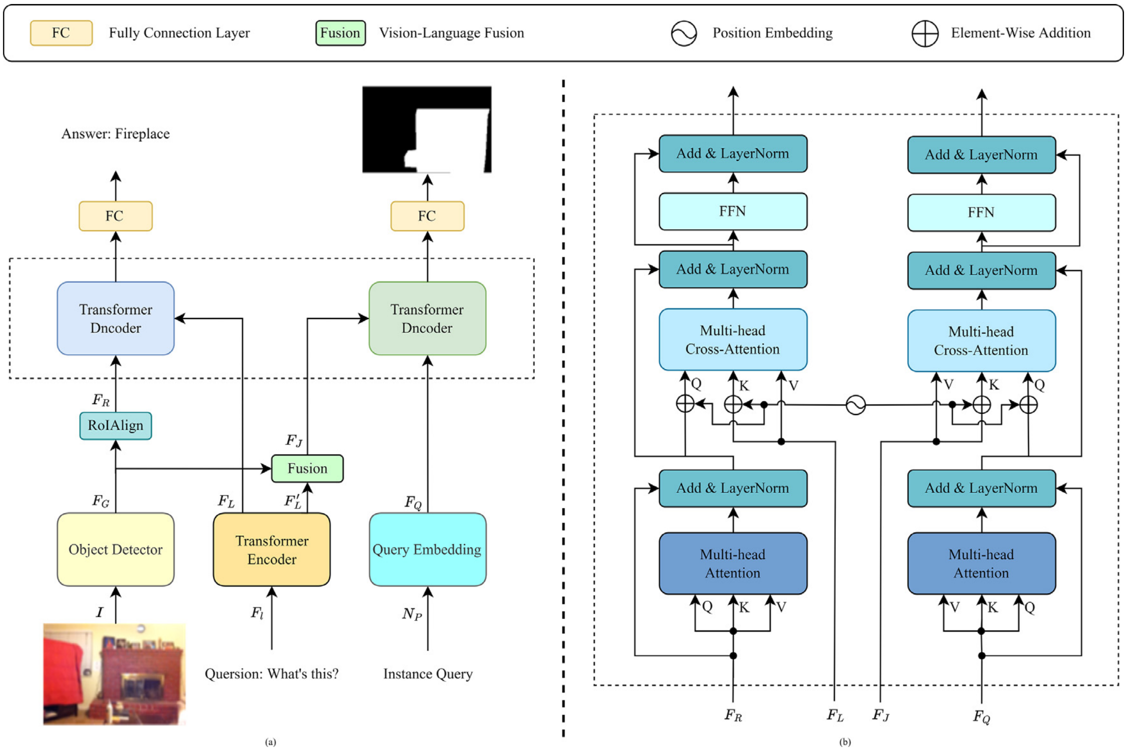


Fig. 1. (a) The overall architecture of DDTN for answer grounding. Image region features F_R and language features F_L are fed into one of the Transformer decoders to predict the answer. After image grid features F_G and shared language features F'_L are fused as joint features F_J, together with instance query embedding F_Q, they are fed into another decoder for predicting the corresponding instance coordinates. (b) The detail of two decoders in DDTN.

and language, it contains two encoders and two decoders, one of which is a ViT [15], while others are BERT [16] or models similar to BERT. In contrast, our proposed method achieves higher efficiency without relying on a web-scale dataset [17] and data augmentation due to the incorporation of instance query and RoIAlign. Instance query guides the generation of the instance coordinates, and RoIAlign generates the grid and area features for two decoders.

In this letter, we propose a novel dual-decoder Transformer network dubbed DDTN for predicting answers and providing corresponding instance coordinates. Different from attention map mechanism, DDTN is built on a Transformer framework consisting of the encoder-decoder [16]. The encoder extracts the shared linguistic features from textual questions, while the decoder, as opposed to classical Transformer, is a dual-decoder structure. Specifically, we exploit an object detector to collect the region and grid features from the image. The region features along with language features are fed into one of the decoders to predict answers. For another decoder to output the corresponding instance coordinates, we design an instance query to guide the joint features fused by grid features and shared language features. On the benchmark VizWiz-Ground [11] and GQA [10] datasets, experimental results demonstrate the remarkable efficiency of our proposed method. The main contributions are summarized as follows:

- In this letter, we propose a novel Transformer network named DTNN, which predicts linguistic answers and visual instances simultaneously by two decoupled decoders.
- We introduce a set of instance query embeddings and a sampling strategy around the mass center to precisely locate the object. By dividing images into bins, the complex coordinate regression can be transformed into a simpler classification prediction.
- By combining image region features and grid features, the model parameters and computational costs can be dramatically reduced, which allows DDTN to be trained and inferred on cheap devices.
- Without bells and whistles, DDTN achieves state-of-the-art performance and even competes with pretraining models on the benchmark VizWizGround and GQA datasets.

The rest of this letter is organized as follows: Section 2 discusses the related works on VQA and the Transformer network. Section 3 gives a detailed explanation of the proposed method. Section 4 describes the dataset, experimental details, ablation experiments and qualitative analysis. The conclusion is presented in Section 5.

2. Related works

2.1. Visual question answering

In the past, Visual Question Answering (VQA) has relied primarily on elaborate fusion frameworks [18] to overcome the semantic divide between vision and language. Kim et al. [19] proposed bilinear attention networks built on low-rank bilinear pooling to seamlessly exploit the input visual-linguistic information. Since the parameters of the bilinear model grows quadratically with the dimension of input features, it is difficult to deploy to practical applications. Ben-Younes et al. [20] developed a multimodal fusion method based on block-superdiagonal tensor decomposition to balance the expressiveness and complexity. Recent advances have been achieved with Transformer architecture and large-scale datasets as well as pretraining tasks. Tan and Bansal [21] proposed a two-stream Transformer framework to learn the relation between vision and language. Its backbone network consists of three encoders that have been pre-trained on five tasks, such as masked object prediction, label classification, and cross-modality matching. In order to mitigate the semantic gap between images and text, Li et al. [22] introduced the object tags detected from images to build the Word-Tag-Image triple as the Transformer inputs. Zhang et al. [23] proposed VinVL to improve the object-centered visual representation for demonstrating the importance of vision features in multimodal field. Instead, Kim et al. [24] developed a convolution-free method to speed up the model inference, but it performs slightly worse than the region-based approaches. To reduce the complexity of the pretraining process, Wang et al. [25] proposed a simple framework that only requires weakly aligned image-text pairs. Although these methods improve the performance of VQA models substantially, they do not directly provide visual evidence to demonstrate their inference ability. In this letter, we propose a dual-decoder Transformer network to output the object mask or box with the guidance of joint vision-language features.

2.2. Transformer network

Original Transformer network was proposed in Vaswani et al. [26] for natural language processing (NLP). It has an encoder-decoder structure stacked by Transformer layers. As self- and cross- attention mechanism cannot capture the ordering information, positional embedding is introduced to combine with the input sequences. After that, Transformer was applied to the field of computer vision (CV). ViT [15] is an influential milestone in CV research that converts images into sequence tokens by using linear projection of flattened patches. ViLT [24] applies this trick to encode images and text as a unified sequence to be fed into the Transformer encoder. TRAR [27] proposed a dynamic routing scheme that can adaptively select receptive fields in response to input samples of different sizes. Unlike convolutional neural networks that modify the convolutional kernels to change the receptive fields, TRAR accomplishes this by placing an adjacency mask in the attention mechanism. Self-attention is computed only once per layer, so there are no additional computational costs. For fair comparison, our proposed DDTN uses the original Transformer encoder and decoder structure. The only difference is that DDTN has two decoders with different number of Transformer layers.

3. The proposed method

3.1. Overall network framework

Figure 1 illustrates the overall architecture of our proposed Dual-Decoder Transformer network (DDTN). It leverages two Transformer decoders to combine language and vision modalities for predicting textualized answers and visualized instances. To model long-range interactions and semantic correlation of each modality, DDTN first extracts image grid features F_G and language features F_L by the object detector and the Transformer encoder respectively. Meanwhile, an instance query F_q is defined as an embedding F_Q for guiding the generation of instance coordinates. Then, language features F_L are fed into one of the decoders along with image region features F_R obtained by performing RoIAlign on F_G. The shared language features F_L' and grid features F_G are reformed into joint features F_J, which are fed into another decoder together with the embed query F_Q. Lastly, two fully connected layers (FC) map the output of each decoder to respective classification space.

3.2. Features extraction and instance query generation

Given an image I with width W and height H, we adopt ResNet [28] as the object detector to extract the original grid feature map $F_G \in \mathbb{R}^{\frac{H}{16}\times\frac{W}{16}\times 2048}$ on stage-4. Flowing previous methods, RoIAlign [29] is implemented to produce salient regions features $F_R = \{r_i\}_{i=1}^{N_r}$ on F_G, where $r_i \in \mathbb{R}^{14\times14\times2048}$ represents the feature of the i-th region and N_r is the number of regions. Then, the grid and region features are transformed into $F_G \in \mathbb{R}^{\frac{H}{32}\times\frac{W}{32}\times 1024}$ and $F_R = \{R_i\}_{i=1}^{N_r} \in \mathbb{R}^{N\times7\times7\times1024}$ by ResNet stage-5. To obtain language features, the 300-D GloVe word embeddings and one-layer LSTM network with C_l hidden units are used to embed the input question with t words into high-dimensional embedding set $F_l \in \mathbb{R}^{N_t\times C_l}$. Taking F_l as input, the Transformer encoder performs further learning from word-to-sequence to output the language features F_L. Since F_G and F_L do not match in dimension, we compress F_L into a 1D vector via the self-attention mechanism. It can be formulated as:

$$F_L' = \text{softmax}\left(\frac{\omega_L F_L^T}{\sqrt{C_l}}\right)F_L \tag{1}$$

where $\omega_L \in \mathbb{R}^{1\times C_l}$ is the projection matrix. With prepared grid features F_G and shared linguistic features F_L', the joint features F_J are acquired via a fusion process, which is described as follows:

$$F_J = \tanh(F_G) \odot \tanh\left(F_L'\right) \tag{2}$$

where $\odot$ represents Hadamard product.

Inspired by DETR [30] and Pix2seq [31], we design an instance query N_p to learn the positional embeddings $F_Q \in \mathbb{R}^{N_p\times C_p}$ for guiding the locating of visual instances. $N_p = 2\times P$ represents the number of coordinates of all P reference points. The model does not need to regress the coordinates of instance masks or bounding boxes, but merely classifies it to a fixed position on the image. For a sequence of points $\left\{\hat{x}_i, \hat{y}_i\right\}_{i=1}^{P}$, it can be quantized into integer bins:

$$\begin{cases} x_i = \text{round}\left(\frac{\hat{x}_i}{W} * M\right) \\ y_i = \text{round}\left(\frac{\hat{y}_i}{H} * M\right) \end{cases} \tag{3}$$

where M is the number of bins. To choose appropriate referring locations, we sample P contour points uniformly around the mass center of each visual instance.

3.3. Dual-decoder structure and model training

Both Transformer decoders in DDTN have the same structure. The difference is the number of layers and inputs. As illustrated in Fig. 1(b), the decoder on the left receives image region features F_R and language features F_L, while the decoder on the right receives joint features F_J and instance embeddings F_Q. Before being fed into decoders, all features must be converted into sequences. We reshape F_R into $N_r \times 2048$ dimensions via an 2D adaptive pooling, and then map the channels to C_h by a fully connected layer. For joint features F_J, we transform the channel 2048 to C_h by a 1×1 convolution, and then flatten its width and height. To provide spatial information to the Transformer, we add the fixed *sine* position encodings to the grid features. For region features, we use learnable positional encodings for element-wise summation.

Subsequent processes follow the standard decoding procedure. The outputs of the two decoders are passed through two fully connected layers to get the final answers $\widehat{Y}_A$ and instance coordinates $\widehat{Y}_M$ respectively. Unlike many methods that use auxiliary losses in decoder [32–34], DDTN is trained uniformly by minimizing the binary cross-entropy loss. $\mathcal{L}_{bce}\left(\widehat{Y}, Y\right)$:

$$\mathcal{L}\left(\widehat{Y}_A, Y_A, \widehat{Y}_M, Y_M\right) = \lambda\mathcal{L}_{bce}\left(\widehat{Y}_A, Y_A\right) + \mathcal{L}_{bce}\left(\widehat{Y}_M, Y_M\right) \tag{4}$$

where λ is a balanced hyperparameter.

4. Experiments

4.1. Datasets and evaluation metrics

We conduct experiments on the VizWizGround [11] and GQA [10] datasets. VizWizGround contains 6494 images for training,

Table 1
Ablation experiment results of the sampling strategies for instance query points.

Points	Params(M)	Uniform sampling					Center sampling				
		Accuracy	IoU	*AP*	AP_{50}	AP_{75}	Accuracy	IoU	*AP*	AP_{50}	AP_{75}
9	61.51	**41.01**	47.06	24.06	**53.13**	18.75	40.36	46.05	22.19	46.88	18.75
18	61.52	40.88	**49.92**	30.63	46.88	28.13	**40.84**	51.84	35.63	59.38	31.25
36	61.54	40.27	48.11	**33.44**	50.00	**37.50**	40.53	**60.69**	**40.94**	**75.00**	**37.50**
72	61.57	39.97	40.34	20.00	43.75	15.63	40.45	47.69	31.25	46.88	34.38

1131 images for validation, and 2373 images for online test. Each sample includes a question-answer pair and the corresponding instance mask. Compared with other VQA datasets, its image quality is poor, which makes the task more challenging. The GQA dataset is made up of 22M question-answer pairs for more than 113K images, providing box-level grounding annotations. Based on predicted answers, the standard VQA accuracy metric is used to evaluate the reasoning performance of models [35]. For grounding quality, except Intersection over Union (IoU), the standard COCO metrics [36] including average precision *AP* with IoU threshold ranges from 0.5 to 0.95 with a step size of 0.05, AP_{50} at IoU 0.5, and AP_{75} at IoU 0.75 are used to evaluate the segmentation performance.

4.2. Implementation details

Following previous works, we use the VinVL [23] framework as the object detector on VizWizGround with weights frozen until stage-5. The number of regions N_r is limited to 36. The input questions are trimmed to a maximum of $N_t = 14$ words. The feature channels of LSTM C_t, query embeddings C_p, and the input of Transformer decoders C_h are set to 256. The language encoder consists of 2 Transformer layers, in which the number of heads is restricted to 4. In the decoder for predicting answer, the Transformer layers are the same as those in the language encoder, while in the decoder for generation coordinates, there are three layers. The size of the answer vocabulary is set to 5254 using the strategy in Teney et al. [37], and the number of bins *M* is equal to 1000 based on Pix2seq [31]. On the GQA dataset, we directly use the image features provided by the official website and remove the object detector. Considering the size of the dataset, we expanded the feature dimension to 512. Additionally, we have increased the number of attention layers in the Transformer encoder and both decoders by one.

On the VizWizGround dataset, we train DDTN 35 epochs with batch size 32 using the Adam optimizer with an initial learning rate 1.0×10^{-4} for Transformer structure, and 1.0×10^{-5} for the object detector. All the models are trained on a single NVIDIA TITAN X GPU. After 30 epochs, the learning rate is decayed by 0.1. All ablation experiments are conducted on the val subset. On the GQA dataset, we trained DDTN for 3 epochs with batch size 128. After 17 K iterations, the learning rate decays by 0.1. The balance coefficient λ for both datasets is set to 1, and the weights of all transformer layers have been randomly assigned.

4.3. Ablation studies

4.3.1. Sampling strategies

In supervised learning, label has a decisive impact on model performance. Therefore, we first analyze the effect of the sampling strategy at different number of sampling points. Uniform sampling and center sampling are the most common sampling methods. The former takes out points uniformly on the object contour to form ground truth, while the latter selects points radially on the contour around the object mass center. Following most of the instance segmentation works [38], the sampling is conducted in multiples of 9, with each time being twice as large as the previous one, and the results are shown in Table 1.

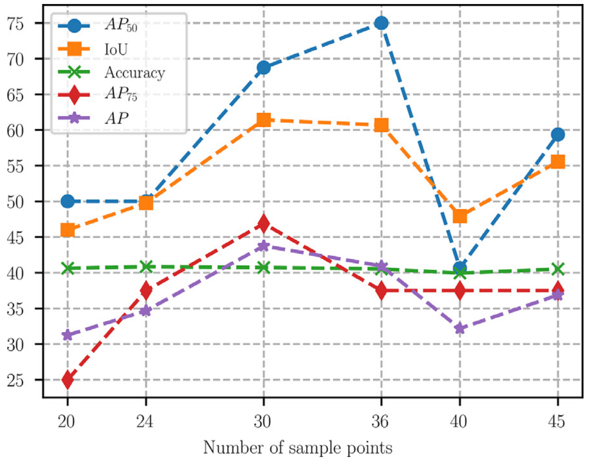
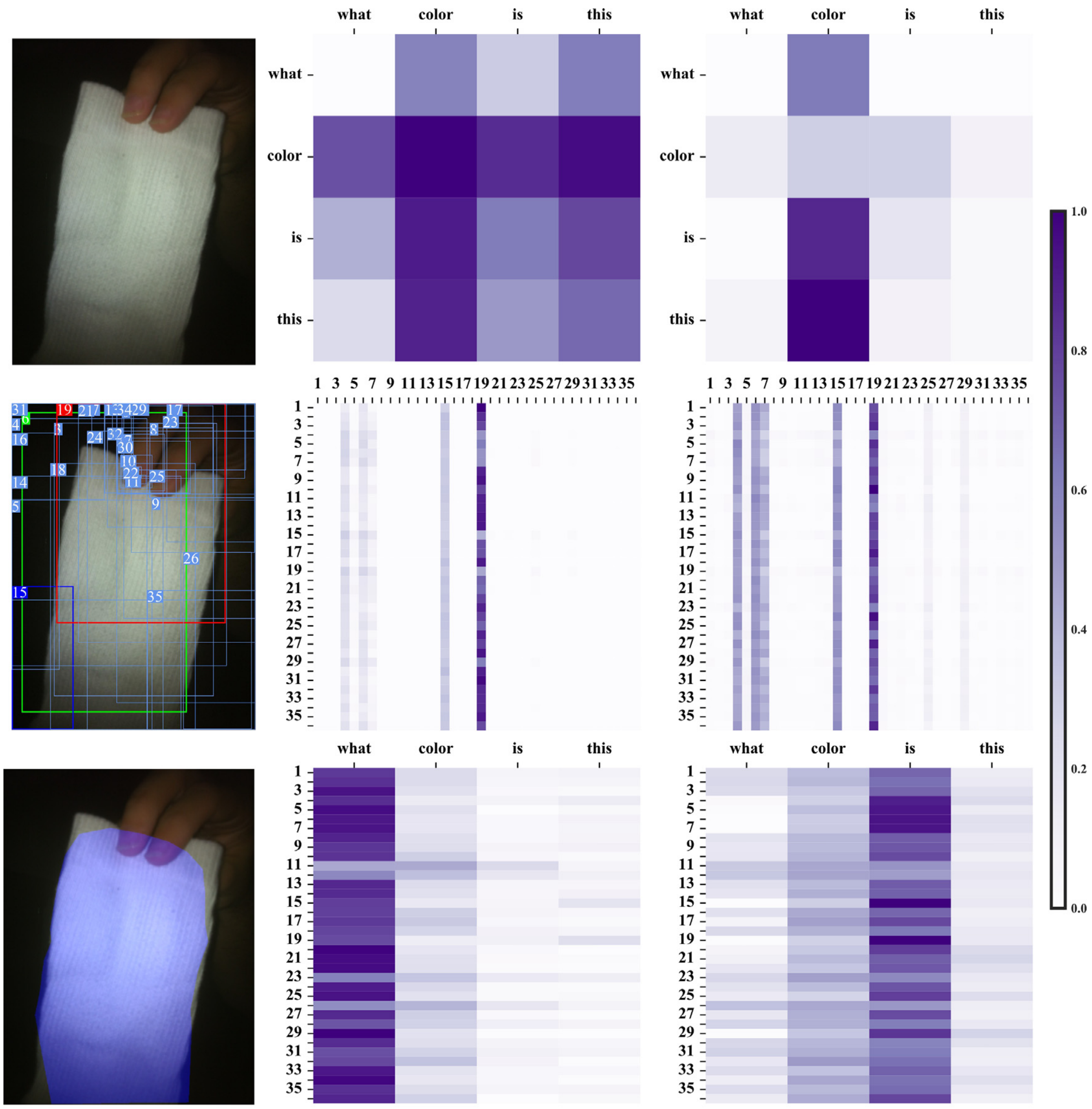


Fig. 2. Ablation experiment results of the number of sample points for answer accuracy and segmentation performance.

Regardless of the sampling method, the total parameters increase slightly as the change in labels only affects the parameters of the classification layer in the network. For uniform sampling, the model with 9 points achieved the maximum answer accuracy and AP_{50}, but its scores are below other models on IoU, *AP*, and AP_{75}. The model with 18 points and 36 points achieved the maximum IoU and *AP* scores, respectively. Meanwhile, the model with 36 points performs best on the AP_{75} metric, which requires higher segmentation performance. These results can be attributed to the following reasons: 1) Fewer points allow the model to focus on the more difficult VQA task; 2) Fewer points construct a mask that occupies more area, but loses fine-grained edge information; 3) As the number of points increases, the constructed mask becomes more and more complex, even to the detriment of model performance. Similar results are seen in the center sampling. However, the latter is higher than uniform sampling in overall performance. Therefore, we choose the center sampling method as the default setting, and explore the optimal number of sampling points in detail.

4.3.2. The number of sample points

For center sampling, we further explore the effect of the number of sampling points *N* on model performance. Depending on the angle around the mass center, *N* must be divisible by 360°. We sampled from 20 to 45 points and the results are shown in Fig. 2. With the increasing number of sampling points, the segmentation performance of the model gradually improves, while the correct answering rate remains about 40%, which indicates that our dual decoder is decoupled. When $N = 30$, the model performance reaches saturation. As *N* continues to increase, the rough edges also rise, resulting in decreased *AP* and AP_{75}. Hereinafter, we keep the number of instance query points at 30 in the following experiments unless otherwise stated.



(a)   (b)

Fig. 3. (a) From top to bottom are the original image, the detected 36 object image, and the output instance mask. (b) From top to bottom are the two layer attention maps of the language encoder, and the two layer self- and cross-attention maps of the decoder that outputs the answer. Each index within [1–36] corresponds to an object in subfigure (a). We highlight three top scoring objects for a better visual effect.

Table 2
Ablation experiment results of the decoder and image features for answering grounding in visual question answering.

Feature type		Single-decoder						Dual-decoder					
Region	Grid	Params(M)	Accuracy	IoU	*AP*	*AP*$_{50}$	*AP*$_{75}$	Params(M)	Accuracy	IoU	*AP*	*AP*$_{50}$	*AP*$_{75}$
✓		58.90	37.54	35.28	17.90	31.48	15.83	61.53	40.56	47.25	23.44	53.13	18. 75
	✓	58.90	37.75	39.78	21.30	38.20	19.19	61.53	38.30	58.90	42.81	62.50	46.55
✓	✓	58.90	36.91	40.50	21.38	39.70	17.77	61.53	**40.73**	**61.42**	**43.75**	**68.75**	**46.88**

4.3.3. Decoder and image feature

To investigate the effect of the number of decoders as well as region and grid features, we tested answer accuracy and segmentation performance while keeping the language features constant. When comparing single and dual decoders, we maintain a single decoder with three attention layers since the only difference between two decoders is the number of attention layers. When comparing image region and grid features, 2D adaptive pooling is utilized to compress the width and height of grid features. As shown in Table 2, DDTN performs poorly when only a single decoder is used. For the following reasons, we believe that a single decoder does not perform as well as a dual decoder. In order to output both the textualized answer and the instance mask, two classification layers are required. Consequently, if only a single decoder is employed, the module before the classification layer should learn both textualized answer features and visualized location features. In contrast, two individual decoders tackle this task more efficiently. In terms of region or grid features, the former contain evi-

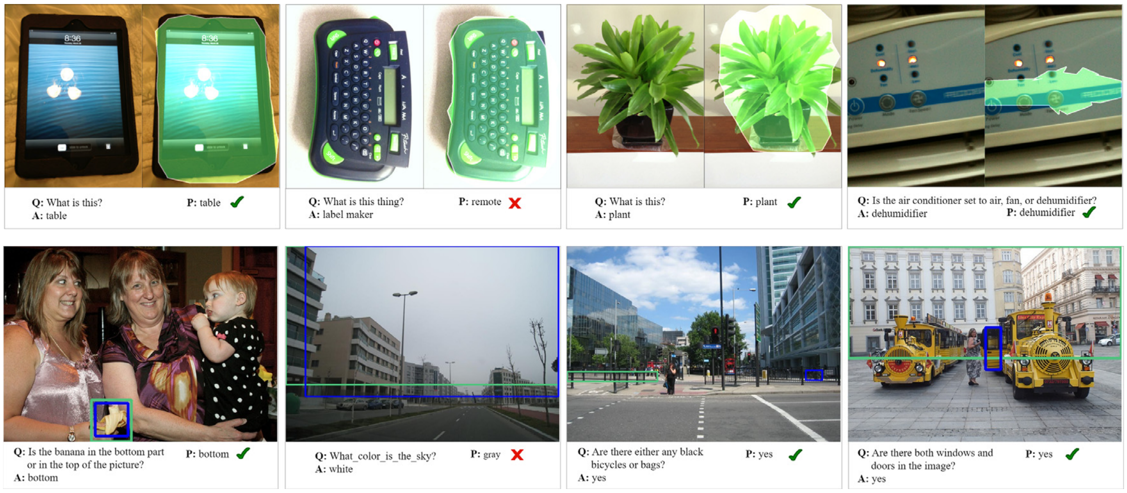

Q: What is this?
A: table
P: table ✓
Q: What is this thing?
A: label maker
P: remote ✗
Q: What is this?
A: plant
P: plant ✓
Q: Is the air conditioner set to air, fan, or dehumidifier?
A: dehumidifier
P: dehumidifier ✓
Q: Is the banana in the bottom part or in the top of the picture?
A: bottom
P: bottom ✓
Q: What_color_is_the_sky?
A: white
P: gray ✗
Q: Are there either any black bicycles or bags?
A: yes
P: yes ✓
Q: Are there both windows and doors in the image?
A: yes
P: yes ✓

Fig. 4. Typical examples of the model inputs and outputs. The first row is a sample of images and predicted results from VizWizGround. The image, question (Q), and answer (A) for each example are shown on the left; the mask of the output object and prediction (P) are present on the right. The second row shows sample images and prediction results within GQA. The blue line represents ground truth, and the green line or mask is the predicted result. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 3
Results of the fusion method between the vision and language features on the VizWizGround val set.

Methods	Params(M)	Accuracy	IoU	AP	AP_{50}	AP_{75}
Pro.	61.53	39.93	57.85	39.69	59.38	43.75
Sum.	61.53	39.89	59.61	42.81	62.50	40.63
Tanh. & Sum.	61.53	39.86	60.23	**43.75**	65.63	**46.88**
Tanh. & Pro.	61.53	**40.73**	**61.42**	**43.75**	**68.75**	**46.88**

dent semantic information, while the latter contain rich spatial information. When we use both region and grid features, the performance of DDTN is improved, which validates the effectiveness of our method. For parameters, the adaptive pooling operation does not increase the computational cost. Similarly, replacing grid features with region features is conventional because the dimensionality meets the Transformer requirement, which has no effect on the number of total parameters.

4.3.4. Fusion module

Table 3 summarizes the effect of visual and linguistic feature fusion methods on model performance. The first line Pro. (Product) means that the grid features and language features are fused directly by Hadamard products under the broadcast mechanism. Similarly, the second line Sum. (Summation) denotes the element-by-element summation of these features. Tanh. & Sum., i.e. hyperbolic tangent function and summation, represents the summation of the region features and language features after tanh activation respectively. The last line Tanh. & Pro. is the fusion method used in Eq. (2). We did not consider more complex fusion approaches for simplicity and leave them for future exploration. In Table 3, Tanh. & Pro. achieves the best performance, and we use it for further comparison.

4.4. Comparison with others

On the VizWizGround test set, we compare the performance and parameters of the proposed method with other state-of-the-art methods. Since the current version only supports comparison of IoU, we select the $N = 30$ model based on the results of Fig. 2. MAC-Caps [12] is a capsule network built on an attention map mechanism without pretraining, while UNIFIED-IO [13] is a sequence-to-sequence model with pre-training and multi-tasking on large-scale data, then tested on the VizWizGround dataset. Additionally, MAC-Caps also uses the VinVL model to extract the image features. As shown in Table 4, our proposed DDTN achieves state-of-the-art performance without any pretraining, even exceeding the base pre-training model by 3.4 percent. Furthermore, it has the fewest parameters than other models. On the GQA evaluation set, we set $N = 2$ to predict bounding boxes. MAC-Caps and DDTN use publicly available image features to perform training and validation, while UNIFIED-IO is performing validation experiments on GQA directly since its training data contains the original dataset of GQA. In Table 4, DDTN with a minimum number of parameters outperforms other models by a wide margin in answer accuracy. In addition, it has the highest IoU and AP_{50} scores for grounding accuracy. These results illustrate the effectiveness of our proposed method.

4.5. Qualitative analysis

In Fig. 3, we visualize the intermediate processing of the DDTN and the output mask. The language encoder focuses on the keywords in the question. In contrast, the decoder that outputs the answer looks at the interrogative word as well as the global objects in the image. By observing the mask, we find that the proposed instance query adequately models the object contour. These results indicate that DDTN is able to accurately locate the object relevant to the answer.

In Fig. 4, we also visualize the typical output masks or boxes and predicted answers. In the first row, the masks usually contain visual evidence that supports the answers, even if the image quality is extremely poor (e.g., the “dehumidifier” in the last example). The second and third sample illustrates some weaknesses of our method. For example, it hardly distinguishes between objects with identical appearance (e.g., the “label maker” and “remote”). In addition, our method has less ability to model contours of complex objects (e.g., the “plant” in the third sample). For complex street scenarios within GQA, we have found that the model tends to predict an oversize box (e.g., the “black bicycles” or “windows and doors” in the first or fourth sample). These observations are instructive for our future research. We can see that, in the second or third column, a better VQA or visual grounding model can improve the overall accuracy or segmentation performance.

Table 4
Results of the Comparison with state-of-the-art methods on the VizWizGround test set and GQA val set.

Methods	VizWizGround		GQA					
	Params(M)	IoU	Params(M)	Accuracy	IoU	*AP*	*AP*$_{50}$	*AP*$_{75}$
MAC-Caps [12]	74.3	27.3	74.3	55.13	7.03	–	–	-
UNIFIED-IO$_{\text{SMALL}}$ [13]	71.0	35.5	71.0	38.13	28.23	19.58	26.24	19.74
UNIFIED-IO$_{\text{BASE}}$ [13]	241.0	50.0	241.0	41.12	33.60	25.51	32.32	26.21
UNIFIED-IO$_{\text{LARGE}}$ [13]	776.0	**54.7**	776.0	39.32	37.17	**29.08**	36.36	**30.01**
DDTN (Ours)	**61.5**	53.4	**67.4**	**58.54**	**38.18**	28.25	**37.36**	28.88

5. Conclusion

In this letter, we propose a dual decoder Transformer network (DDTN) for grounding answer in VQA, which uses decoupled dual decoders to predict the answer and instance separately. To reduce the training costs, DDTN applies RoIAlign on the image grid features to generate regional features. Benefiting from instance query embeddings and image quantization, we transform the complex object coordinate regression into a classification task. Ablation experiment results demonstrate the effectiveness of instance query and the interpretability of the overall framework. The visualization results show that our method can accurately locate the visual evidence that supports the answer. In addition, a suitable VQA model and sampling strategy deserve further exploration to improve answer correct rate and segmentation performance.

Declaration of Competing Interest

Authors declare that they have no conflict of interest.

Data availability

The code is available at https://github.com/zlj63501/DDTN

Acknowledgments

This research was funded by the National Natural Science Foundation of China (No. 61873112) and the Young Scientists Fund of the National Natural Science Foundation of China (No. 62106082).

References

[1] W. Li, J. Sun, G. Liu, L. Zhao, X. Fang, Visual question answering with attention transfer and a cross-modal gating mechanism, Pattern Recognit. Lett. 133 (2020) 334–340.
[2] V. Lioutas, N. Passalis, A. Tefas, Explicit ensemble attention learning for improving visual question answering, Pattern Recognit. Lett. 111 (2018) 51–57.
[3] A. Al-Sadi, M. Al-Ayyoub, Y. Jararweh, F. Costen, Visual question answering in the medical domain based on deep learning approaches: a comprehensive study, Pattern Recognit. Lett. 150 (2021) 57–75.
[4] H. Zhong, J. Chen, C. Shen, H. Zhang, J. Huang, X. Hua, Self-adaptive neural module transformer for visual question answering, IEEE Trans. Multimed. 23 (2021) 1264–1273.
[5] D. Gurari, Q. Li, A.J. Stangl, A. Guo, C. Lin, K. Grauman, J. Luo, J.P. Bigham, Vizwiz grand challenge: answering visual questions from blind people, in: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, 2018, pp. 3608–3617.
[6] A.B. Abacha, S.A. Hasan, V.V. Datla, J. Liu, D. Demner-Fushman, H. Müller, Vqa-med: overview of the medical visual question answering task at imageclef 2019, in: Working Notes of CLEF 2019 - Conference and Labs of the Evaluation Forum, 2019, in: CEUR Workshop Proceedings, vol. 2380, 2019.
[7] S. Whitehead, H. Wu, H. Ji, R. Feris, K. Saenko, Separating skills and concepts for novel visual question answering, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, 2021, pp. 5632–5641.
[8] Q. Wu, D. Teney, P. Wang, C. Shen, A.R. Dick, A. van den Hengel, Visual question answering: a survey of methods and datasets, Comput. Vis. Image Underst. 163 (2017) 21–40.
[9] Y. Niu, K. Tang, H. Zhang, Z. Lu, X. Hua, J. Wen, Counterfactual VQA: a cause-effect look at language bias, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, 2021, pp. 12700–12710.
[10] D.A. Hudson, C.D. Manning, GQA: a new dataset for real-world visual reasoning and compositional question answering, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, 2019, pp. 6700–6709.
[11] C. Chen, S. Anjum, D. Gurari, Grounding Answers for Visual Questions Asked by Visually Impaired People, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 19098–19107.
[12] A.U. Khan, H. Kuehne, K. Duarte, C. Gan, N. da Vitoria Lobo, M. Shah, Found a reason for me? weakly-supervised grounded visual question answering using capsules, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, 2021, pp. 8465–8474.
[13] J. Lu, C. Clark, R. Zellers, R. Mottaghi, A. Kembhavi, Unified-io: a unified model for vision, language, and multi-modal tasks, CoRR abs/2206.08916 (2022).
[14] J. Pan, G. Chen, Y. Liu, J. Wang, C. Bian, P. Zhu, Z. Zhang, Tell me the evidence? Dual visual-linguistic interaction for answer grounding, CoRR abs/2207.05703 (2022).
[15] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, N. Houlsby, An image is worth 16 × 16 words: transformers for image recognition at scale, in: 9th International Conference on Learning Representations, ICLR 2021, 2021.
[16] J. Devlin, M. Chang, K. Lee, K. Toutanova, BERT: pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, vol. 1, 2019, pp. 4171–4186.
[17] J. Li, D. Li, C. Xiong, S.C.H. Hoi, BLIP: bootstrapping language-image pre-training for unified vision-language understanding and generation, in: International Conference on Machine Learning, ICML 2022, in: Proceedings of Machine Learning Research, vol. 162, 2022, pp. 12888–12900.
[18] H. Sharma, A.S. Jalal, A survey of methods, datasets and evaluation metrics for visual question answering, Image Vis. Comput. 116 (2021) 104327.
[19] J. Kim, J. Jun, B. Zhang, Bilinear attention networks, in: Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 2018, pp. 1571–1581.
[20] H. Ben-Younes, R. Cadène, N. Thome, M. Cord, BLOCK: bilinear superdiagonal fusion for visual question answering and visual relationship detection, in: The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, 2019, pp. 8102–8109.
[21] H. Tan, M. Bansal, LXMERT: learning cross-modality encoder representations from transformers, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, 2019, pp. 5099–5110.
[22] X. Li, X. Yin, C. Li, P. Zhang, X. Hu, L. Zhang, L. Wang, H. Hu, L. Dong, F. Wei, Y. Choi, J. Gao, Oscar: Object-semantics aligned pre-training for vision-language tasks, in: Computer Vision - ECCV 2020 - 16th European Conference, in: Lecture Notes in Computer Science, vol. 12375, 2020, pp. 121–137.
[23] P. Zhang, X. Li, X. Hu, J. Yang, L. Zhang, L. Wang, Y. Choi, J. Gao, Vinvl: revisiting visual representations in vision-language models, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, 2021, pp. 5579–5588.
[24] W. Kim, B. Son, I. Kim, Vilt: vision-and-language transformer without convolution or region supervision, in: Proceedings of the 38th International Conference on Machine Learning, ICML 2021, in: Proceedings of Machine Learning Research, vol. 139, 2021, pp. 5583–5594.
[25] Z. Wang, J. Yu, A.W. Yu, Z. Dai, Y. Tsvetkov, Y. Cao, Simvlm: simple visual language model pretraining with weak supervision, in: The Tenth International Conference on Learning Representations, ICLR 2022, 2022.
[26] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 2017, pp. 5998–6008.
[27] Y. Zhou, T. Ren, C. Zhu, X. Sun, J. Liu, X. Ding, M. Xu, R. Ji, TRAR: routing the attention spans in transformer for visual question answering, in: 2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, 2021, pp. 2054–2064.
[28] Z. Lu, Y. Bai, Y. Chen, C. Su, S. Lu, T. Zhan, X. Hong, S. Wang, The classification of gliomas based on a pyramid dilated convolution resnet model, Pattern Recognit. Lett. 133 (2020) 173–179.
[29] K. He, G. Gkioxari, P. Dollár, R.B. Girshick, Mask R-CNN, in: IEEE International Conference on Computer Vision, ICCV 2017, 2017, pp. 2980–2988.
[30] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, S. Zagoruyko, End-to-end object detection with transformers, in: Computer Vision - ECCV 2020 - 16th European Conference, in: Lecture Notes in Computer Science, vol. 12346, 2020, pp. 213–229.

[31] T. Chen, S. Saxena, L. Li, D.J. Fleet, G.E. Hinton, Pix2seq: a language modeling framework for object detection, in: The Tenth International Conference on Learning Representations, ICLR 2022, 2022.

[32] S. Liu, F. Li, H. Zhang, X. Yang, X. Qi, H. Su, J. Zhu, L. Zhang, DAB-DETR: dynamic anchor boxes are better queries for DETR, in: The Tenth International Conference on Learning Representations, ICLR 2022, 2022.

[33] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, J. Dai, Deformable DETR: deformable transformers for end-to-end object detection, in: 9th International Conference on Learning Representations, ICLR 2021, 2021.

[34] G. Zhang, Z. Luo, Y. Yu, K. Cui, S. Lu, Accelerating DETR convergence via semantic-aligned matching, CoRR abs/2203.06883 (2022).

[35] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C.L. Zitnick, D. Parikh, VQA: visual question answering, in: 2015 IEEE International Conference on Computer Vision, ICCV 2015, 2015, pp. 2425–2433.

[36] T. Lin, M. Maire, S.J. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C.L. Zitnick, Microsoft COCO: common objects in context, in: Computer Vision - ECCV 2014 - 13th European Conference, in: Lecture Notes in Computer Science, vol. 8693, 2014, pp. 740–755.

[37] D. Teney, P. Anderson, X. He, A. van den Hengel, Tips and tricks for visual question answering: learnings from the 2017 challenge, in: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, 2018, pp. 4223–4232.

[38] S. Minaee, Y. Boykov, F. Porikli, A. Plaza, N. Kehtarnavaz, D. Terzopoulos, Image segmentation using deep learning: a survey, IEEE Trans. Pattern Anal. Mach. Intell. 44 (7) (2022) 3523–3542.