# LGR-NET: Language Guided Reasoning Network for Referring Expression Comprehension

Mingcong Lu, Ruifan Li, *Member, IEEE*, Fangxiang Feng, Zhanyu Ma, *Senior Member, IEEE*, and Xiaojie Wang

***Abstract*— *Referring Expression Comprehension* (REC) is a fundamental task in the vision and language domain, which aims to locate an image region according to a natural language expression. REC requires the models to capture key clues in the text and perform accurate cross-modal reasoning. A recent trend employs transformer-based methods to address this problem. However, most of these methods typically treat image and text equally. They usually perform cross-modal reasoning in a crude way, and utilize textual features as a whole without detailed considerations (e.g., spatial information). This insufficient utilization of textual features will lead to sub-optimal results. In this paper, we propose a *Language Guided Reasoning Network* (LGR-NET) to fully utilize the guidance of the referring expression. To localize the referred object, we set a prediction token to capture cross-modal features. Furthermore, to sufficiently utilize the textual features, we extend them by our Textual Feature Extender (TFE) from three aspects. *First*, we design a novel coordinate embedding based on textual features. The coordinate embedding is incorporated to the prediction token to promote its capture of language-related visual features. *Second*, we employ the extracted textual features for Text-guided Cross-modal Alignment (TCA) and Fusion (TCF), alternately. *Third*, we devise a novel cross-modal loss to enhance cross-modal alignment between the referring expression and the learnable prediction token. We conduct extensive experiments on five benchmark datasets, and the experimental results show that our LGR-NET achieves a new state-of-the-art. Source code is available at https://github.com/lmc8133/LGR-NET.**

***Index Terms*— Vision and language, referring expression comprehension, transformer, cross-modal reasoning.**

Manuscript received 24 October 2023; revised 25 January 2024; accepted 5 March 2024. Date of publication 7 March 2024; date of current version 12 August 2024. This work was supported in part by Beijing Natural Science Foundation under Project Z200002; in part by the National Natural Science Foundation of China (NSFC) under Grant 62076032, Grant 62225601, and Grant U23B2052; in part by the Youth Innovative Research Team of Beijing University of Posts and Telecommunications (BUPT) under Grant 2023QNTD02; and in part by the High-Performance Computing Platform of BUPT. This article was recommended by Associate Editor F. Díaz-de-María. *(Corresponding author: Ruifan Li.)*

Mingcong Lu is with the School of Artificial Intelligence, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: lmc8133@bupt.edu.cn).

Ruifan Li, Fangxiang Feng, and Xiaojie Wang are with the School of Artificial Intelligence, the Engineering Research Center of Information Networks, Ministry of Education, the Key Laboratory of Interactive Technology and Experience System, Ministry of Culture and Tourism, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: rfli@bupt.edu.cn; fxfeng@bupt.edu.cn; xjwang@bupt.edu.cn).

Zhanyu Ma is with the School of Artificial Intelligence, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: mazhanyu@bupt.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSVT.2024.3374786.

Digital Object Identifier 10.1109/TCSVT.2024.3374786

## I. Introduction

REFERRING Expression Comprehension (i.e., REC) aims to locate an object in an image according to a natural language expression. In Fig. 1, we concisely show two types of REC frameworks generate the bounding boxes according to an image full of oranges with the referring expression. As a fundamental vision-language task, REC could advance various applications, including image captioning [1], [2], [3], [4], visual question answering (VQA) [5], [6], [7], [8], visual navigation [9], [10], [11], [12], and referring expression segmentation (RES) [13], [14], [15].

To address REC task, a key challenge is how to perform an accurate cross-modal reasoning. Generally, existing methods can be grouped into three categories: two-stage methods, one-stage methods, and transformer-based methods. Specifically, the former two types of methods are based on two-stage or one-stage object detectors, respectively. The two-stage pipelines [16], [17], [18], [19], [20], [21] often generate a set of region proposals of an image at first, and then perform reasoning by retrieving the region with the highest matching score for the given expression. In contrast, one-stage methods [22], [23], [24] perform cross-modal fusion while extracting image features and directly predict the box with maximal confidence score over pre-defined anchors. These two types of methods are both based on generic object detectors [25], [26] and predict the box over all candidates. Therefore, their performance is often limited by generated proposals or pre-defined anchors.

Very recently, several transformer-based frameworks [27], [28], [29], [30], [31] have been proposed for REC task. As shown in Fig. 1 a), those frameworks employ a stack of transformer layers to facilitate cross-modal reasoning and regress the bounding box directly without off-the-shelf detectors. Compared with previous two-stage and one-stage methods, these transformer-based methods are more elegant and achieve better performance.

However, these methods treat visual and textual features equally when performing cross-modal reasoning. They usually utilize visual and textual features in a connected-attention way for cross-modal reasoning as shown in Fig. 1 a). In addition, the textual features are used as a whole without specific distinctions. Nevertheless, the referring expression and image play different roles in REC. The referring expression is a crucial guidance for cross-modal reasoning, and image is the

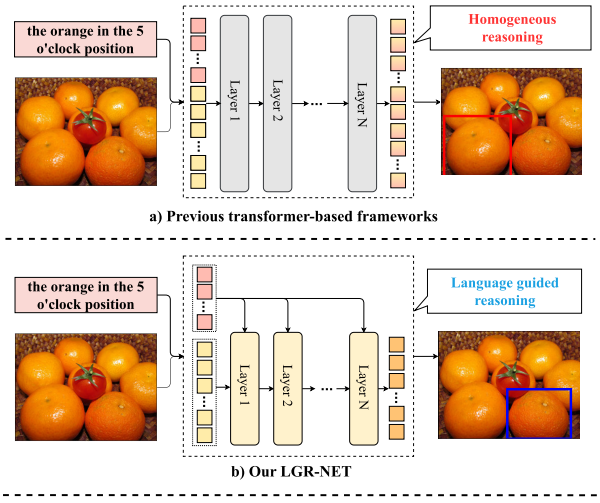



Fig. 1. An image with a referring expression is given to illustrate the definition of REC task and the major difference between two types of methods. a) Most existing transformer-based methods often use multi-modal features in a naïve fashion, i.e., connected-attention. b) Our LGR-NET fully utilizes the language features to guide the reasoning, obtaining an accurate localization.

carrier for target localization. REC model is asked to capture important clues in the text, fully use them to reason with the image progressively, and finally localize the referred object. However, due to the huge difference in the sequence length between two modalities, the connected-attention will lead to textual information overwhelmed by visual information. Further analysis is given in Sec. V-D (3). Therefore, textual features are insufficiently utilized in such a simple reasoning scheme. This would result in sub-optimal results, especially when the referring expression is complicated. To this end, we separate visual and textual modalities and fully utilize textual features to guide the cross-modal reasoning.

In this paper, we propose a Language Guided Reasoning Network (LGR-NET) to fully utilize textual features for effective cross-modal reasoning and to locate the referred object accurately. We utilize a learnable prediction token to capture the key visual and textual features for the bounding box prediction. Our LGR-NET mainly consists of visual and textual extractors, a Textual Feature Extender (TFE), and stacked Text-guided Cross-modal Alignment (TCA) and Fusion (TCF) modules. Visual and textual feature extractors are utilized for extracting corresponding features. To fully leverage the guidance of textual features in the process of cross-modal reasoning, we devise a novel TFE to extend textual features from three aspects. **First**, a novel coordinate embedding is generated from TFE. We incorporate it into the prediction token, to help capture necessary visual and textual features. **Second**, textual features output from TFE contain key clues in the referring expression. We utilize it to guide the cross-modal reasoning through stacked TCA and TCF modules. In each layer, visual features are firstly aligned with textual features by TCA, and then the aligned features are fused with the textual features in TCF. TCA and TCF work alternatively. Meanwhile, the learnable prediction token progressively captures key cross-modal features, which is then used for the bounding box regression. **Third**, a sentence embedding is generated from TFE. We construct an effective cross-modal loss between the sentence embedding and object features from the prediction token to enhance cross-modal alignment. Experimental results on benchmark datasets demonstrate that our LGR-NET method achieves a new state-of-the-art performance.

Our major contributions are highlighted as follows.

- We propose a novel framework LGR-NET for REC task. LGR-NET uses a TFE to extend textual features from three aspects and fully utilizes them to guide the cross-modal reasoning.
- We design a novel coordinate embedding to enhance the spatial representation of the prediction token. We use textual features to guide the cross-modal alignment and fusion alternately. In addition, we devise a novel loss to enhance the cross-modal alignment.
- We conduct extensive experiments on five benchmark datasets. Experimental results show the effectiveness of our method. A significant improvement is gained over previous state-of-the-art methods, especially on the Ref-COCOg dataset with complex referring expressions.

## II. Related Work

### A. Two-Stage and One-Stage REC Methods

Two-stage methods generate a set of region proposals from an image at first, and then choose the region with the highest matching score. Often, the module used for proposal generating is selective search [32], or pre-trained two-stage detectors [25]. Then, a cross-modal similarity metric is used to measure the matching scores between candidate regions and the referring expression further to select the best one. Early works [16], [33] commonly treat the entire expression as a single unit and focus on region-query similarity. MattNet [17] uses three modular networks to capture the query information from three aspects: subject, location, and relationship to generate fine-grained similarity. Some methods [20], [21] construct multi-modal tree or graph to perform better multi-modal reasoning. Except for improving multi-modal fusion and reasoning performance, Ref-NMS [34] increases the recall of generated proposals at the first stage by query-guided Non-Maximum Suppression. Some other methods [35], [36], [37] focus on weakly supervised REC where the box annotation is not available during training.

One-stage methods usually perform multi-modal fusion while extracting visual features, and predict the bounding box over predefined anchors directly. FAOA [22] extends YOLOv3 [26] detector by concatenating sentence embedding with feature map on each spatial location, and chooses the anchor box with the highest score as result. Subsequently, ReSC [23] extends multi-round interaction with recursive sub-query construction module to address the unsatisfactory performance of FAOA on complex referring expressions. A few works [38] reformulate REC as a sequential reasoning process to adjust the prediction iteratively. Overall, these methods heavily rely on the performance of off-the-shelf object detectors.

### B. Transformer for REC

Transformer [39] is first proposed for machine translation and has been widely used for various NLP tasks [40]. Recently, transformer has been extended for CV tasks, such as image classification [41] and object detection [42], [43],

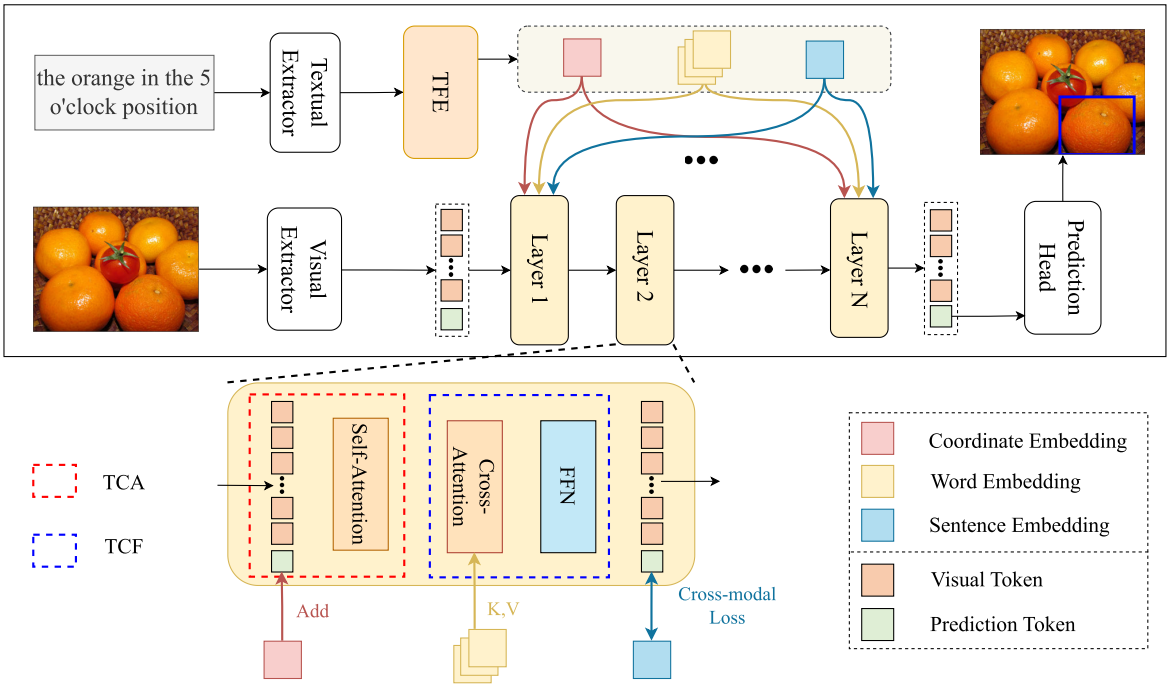



Fig. 2. The overall framework of our proposed LGR-NET. Visual and textual extractors are utilized to obtain multi-modal features, respectively (Sec. III-A). Textual Feature Extender (TFE, Sec. III-B) extends the original textual features from three aspects for subsequent modules. Text-guided Cross-modal Alignment (TCA, Sec. III-C) and Text-guided Cross-modal Fusion (TCF, III-D) work alternately and take the extended textual features for reasoning. The final prediction token is used for localization in prediction head (Sec. III-E). In addition, our loss (Sec. III-F) comprises of the traditional regression loss and our designed cross-modal loss.

[44]. For example, DETR [42] reformulates object detection as a set prediction task, and utilizes a set of learnable object queries, reasoning with visual features through the attention mechanism to predict objects in parallel. ViT [41] constructs a transformer visual backbone, and achieves excellent performance compared to the state-of-the-art convolutional backbone. Swin Transformer [43] conducts shifted window attention to reduce the complexity and makes it available for dense prediction tasks, such as object detection and semantic segmentation.

Transformer-based REC methods use transformer for feature extraction and multi-modal interaction. The pioneering work TransVG [27] encodes visual features by a CNN backbone and a transformer encoder, uses BERT [40] to extract language features, and constructs a transformer encoder (called visual-linguistic transformer) to fuse the concatenated visual-textual features. RefTR [28] and VG-LAW [45] devise models with two prediction heads for REC and RES to conduct multi-task learning and improves the performance of both tasks. QRNet [30] enhances the visual backbone to focus on the language-related region with the guidance of query. VLTVG [31] follows the same feature extraction as TransVG and enhances visual features with a visual-linguistic verification module and a language-guided context encoder.

### C. Vision-Language Pre-Training

Very recently, inspired by the transformer's remarkable success in both NLP and CV domains, several works [29], [46], [47], [48], [49], [50], [51] conduct vision-language pre-training. Basically, these models use large-scale image-text pairs to improve the model's performance in multi-modal understanding, and transfer it to downstream tasks like REC and VQA. CLIP [49] collects 400 million image-text pairs for pre-training and achieves remarkable image-text alignment performance through contrastive learning. OFA [46] unifies different cross-modal tasks in a simple sequence-to-sequence learning framework. MaPLe [52] adopts prompts that adapt both vision and language branches for improved generalization. The prompts are unshared across layers. BLIP2 [51] and MiniGPT4 [53] transfer the general abilities of Large Language Models [54], [55] into multi-modal settings.

## III. Methodology

In this section, we describe the details of our proposed method. The overview of our LGR-NET is shown in Fig. 2. Given an image and a referring expression, we use visual and textual feature extractors to obtain corresponding features. We set a prediction token to capture cross-modal features and use it for localizing the referred object. TFE extends the original textual features from three aspects, generating coordinate embedding, word embeddings, and sentence embedding. The extended textual features are fully utilized for cross-modal reasoning by being repeatedly fed into TCA and TCF modules. Simultaneously, the prediction token is well learned. At last, the prediction head uses the prediction token to generate the bounding box for the referred object. In following subsections, we will elaborate all the components in our method.

### A. Visual and Textual Feature Extractors

Our multi-modal feature extractor is composed of two independent feature extractors of vision and language. The visual

feature extractor uses Swin Transformer [43] as the backbone. Specifically, given a RGB image $I \in \mathbb{R}^{3\times H\times W}$ with height $H$ and width $W$, an initial feature map $F_0 \in \mathbb{R}^{C\times \frac{H}{4}\times \frac{W}{4}}$ with $C$ channels is obtained using a patch partition. After that, four down-sampling stages generate $\times 4$, $\times 8$, $\times 16$, and $\times 32$ four types of down sampled feature maps $[F_1, F_2, F_3, F_4]$, respectively. Then, we use $1 \times 1$ convolution to unify the channel dimension to be $D_v$. By applying the mean pooling and average operation over adjacent feature maps alternately, we obtain an aggregated feature map $F_4^* \in \mathbb{R}^{D_v\times \frac{H}{32}\times \frac{W}{32}}$. Furthermore, to improve the detection performance on huge objects, a feature map $F_5^* \in \mathbb{R}^{D_v\times \frac{H}{64}\times \frac{W}{64}}$ is generated by applying $2\times 2$ max-pooling on the aggregated feature map $F_4^*$. Finally, we flatten and concatenate two previously obtained feature maps, i.e., $F_4^*$ and $F_5^*$ to generate the visual feature $F_{v'} \in \mathbb{R}^{D_v\times N_v}$, in which the feature length $N_v$ equals the number of $\frac{H}{32}\times\frac{W}{32}+\frac{H}{64}\times\frac{W}{64}$.

The textual feature extractor uses Bidirectional Encoder Representation from Transformers (BERT) [40]. Firstly, for a given referring expression, we add a *[CLS]* token at the beginning and a *[SEP]* token at the end of the tokenized expression. Then, we set the maximum length of the expression as $N_t$ and the channel dimension as $D_t$. Thus, we obtain the textual embeddings $F_{t'} \in \mathbb{R}^{D_t\times N_t}$. After extracting the multi-modal features, we use two Multi-Layer Perceptron (MLP) layers to project the vision and language features into a common vector space, keeping the channel dimension identical. Then, we obtain visual features $F_v \in \mathbb{R}^{D\times N_v}$ and textual features $F_t \in \mathbb{R}^{D\times N_t}$, respectively for subsequent modules.

### B. Textual Feature Extender (TFE)

To fully utilize the textual feature for cross-modal reasoning, we devise TFE module to extend textual features from three aspects, i.e., coordinate embedding for the spatial feature, word embeddings for the comprehensive feature, and a sentence embedding for the overall feature. The former two are used for TCA and TCF modules, respectively. The last one is used for the cross-modal loss. The details are illustrated in Fig. 2.

Firstly, a novel coordinate embedding is generated from textual feature to enhance the spatial representation of the prediction token. We find that the referring expression usually contains spatial words to refer target objects, such as "bottom right corner" or "5 o'clock position". Intuitively, using these spatial information can effectively enhance the spatial representation of the prediction token. This information can be captured by the *[CLS]* token. Therefore, we use *[CLS]* token to generate a two-dimensional coordinate through a Multi-layer Perceptron (MLP). The process is formulated as follows,

$$\begin{cases} coord = \text{sigmoid}\left(\text{FFN}_C(h_{cls})\right) \\ p_{\text{coord}} = \text{PE}(coord) \end{cases} \tag{1}$$

where the feed-forward network (FFN) comprises two linear projection layers with a Rectified Linear Unit (ReLU), i.e., $\max(0, z)$. In addition, $h_{\text{cls}}$ is the embedding of the *[CLS]* token and $coord$ is a normalized two-dimensional coordinate. The PE function encodes the coordinate into a $D$-dimensional sinusoidal positional embedding as Transformer [39] does.

Secondly, TFE directly outputs the whole textual features $F_t$ contained in the word embeddings. Thirdly, TFE generates a sentence embedding for cross-modal alignment loss computation (Sec. III-F). Specifically, the sentence embedding is generated from the *[CLS]* embedding using FFN as follows,

$$f_{\text{sent}} = \text{FFN}_S(h_{\text{cls}}) \tag{2}$$

### C. Text-Guided Cross-Modal Alignment (TCA)

As shown in Fig. 2, in TCA we first insert a learnable prediction token $f_p \in \mathbb{R}^{D\times 1}$ in front of the previously extracted visual features $F_v$ as the initial cross-modal representation $X^0 \in \mathbb{R}^{D\times(1+N_v)}$, i.e.,

$$X^0 = [f_p^0, \underbrace{f_1^0, f_2^0, \cdots, f_{N_v}^0}_{\text{visual features}}]. \tag{3}$$

To align the visual features with the textual feature, we utilize attention mechanism. Furthermore, before that we incorporate the spatial information from the referring expression to enhance the prediction token. Specifically, for each layer $i$, we add the generated coordinate embedding $p_{coord}$ from TFE into the prediction token.

$$\hat{f}_p^i = f_p^i + p_{\text{coord}} \tag{4}$$

Subsequently, we utilize multi-head self-attention and a residual connection with layer normalization to aggregate the visual features, i.e.,

$$\begin{aligned} &\underbrace{Q_a = W_{Q_a}^{\text{T}} X^i, K_a = W_{K_a}^{\text{T}} X^i, V_a = W_{V_a}^{\text{T}} X^i}\\ &\tilde{X^i} = \text{LN}\left(\text{softmax}\left(\frac{Q_a K_a^{\text{T}}}{\sqrt{d_k}}\right) V_a + X^i\right) \end{aligned} \tag{5}$$

where three matrices $W_{Q_a}$, $W_{K_a}$, and $W_{V_a}$ denote the linear projection weights for query, key, and value, respectively. TCA's output $\tilde{X^i}$ denotes the aligned visual feature. LN$(\cdot)$ means layer normalization and $d_k$ is the channel dimension.

### D. Text-Guided Cross-Modal Fusion (TCF)

We use a cross-attention mechanism to fuse textual and visual features. Here, the aligned visual feature $\tilde{X^i}$ of TCA's output is used as the query. The textual feature $F_t$ of the referring expression generated in TFE is used as the key and value. The cross-attention mechanism is formulated as follows,

$$\begin{aligned} &\underbrace{Q_f = {W_{Q_f}}^{\text{T}} \tilde{X^i}, K_f = {W_{K_f}}^{\text{T}} F_t, V_f = {W_{V_f}}^{\text{T}} F_t}\\ &\overline{X^i} = \text{LN}\left(\text{softmax}\left(\frac{Q_f \left(K_f\right)^{\text{T}}}{\sqrt{d_k}}\right) V_f + \tilde{X^i}\right)\\ &X^{i+1} = \text{LN}\left(\overline{X^i} + \text{FFN}_{\text{F}}\left(\overline{X^i}\right)\right) \end{aligned} \tag{6}$$

in which, the matrices $W_{Q_f}$, $W_{K_f}$, and $W_{V_f}$ denote the linear projection weights for query, key, and value, respectively. Thus, the output cross-modal representation $X^{i+1}$ captures the key textual information related to the referred object. Meanwhile, the prediction token $f_p^{i+1}$ aggregates the visual-related textual features for further alignment.

Furthermore, to learn better representations, we stack the TCA and TCF module with $N$ layers. In other words, the current output representation $X^{i+1}$ of TCF is used as the input of the next $(i+1)$-th layer of TCA. Thus, under the guidance of language, the prediction token is progressively learned with an accurate representation of the referred object.

### E. Prediction Head

Finally, based on the $N$-th learned prediction token $f_p^N$, we use a three-layer MLP with a Sigmoid activation function to generate the bounding box as follows,

$$(x, y, w, h) = \text{sigmoid}\left(\text{FFN}_H\left(f_p^N\right)\right) \tag{7}$$

in which, two symbols $x$ and $y$ represent the coordinates of its central point. In addition, two symbols $w$ and $h$ denote the corresponding width and height of the box, respectively.

### F. Loss and Training

To train our LGR-NET, we design a loss function with two terms. One is used for the bounding boxes regression and the other for the cross-modal alignment between the prediction token and the sentence embedding. The loss function is given as follows,

$$\mathcal{L}_{\text{total}} = \sum_{i=1}^{N} \mathcal{L}_{\text{box}_i} + \lambda \sum_{i=1}^{N} \mathcal{L}_{\text{align}_i} \tag{8}$$

The former is used to help the prediction token capture the referred object's extremity feature. The latter promotes to capture the representative feature of the object corresponding to the referring expression. Detailed illustration will be given in Sec. V-D (1). The hyper-parameter $\lambda$ is used to balance them.

Specifically, our model contains $N$ stacked TCA and TCF modules, and we denote the predicted result of the $i$-th layer as $b_i = (x_i, y_i, w_i, h_i)$. The corresponding ground-truth is denoted as $\hat{b} = (\hat{x}, \hat{y}, \hat{w}, \hat{h})$. Thus, the former bounding box loss in Eq. (8) of the $i$-th layer is given as follows,

$$\mathcal{L}_{\text{box}_i} = \mathcal{L}_{\text{GIoU}}\left(b_i, \hat{b}\right) + \mathcal{L}_{L_1}\left(b_i, \hat{b}\right), \tag{9}$$

in which, the two terms $\mathcal{L}_{GIoU}(\cdot,\cdot)$ and $\mathcal{L}_{L_1}(\cdot,\cdot)$ are GIoU loss [56] and $L_1$ loss, respectively.

In addition, to enhance the alignment of the referred object and the referring expression, we design an alignment loss. We take the matched image-text pair in the batch as positive samples. In contrast, the other unmatched pairs are regarded as negative samples. Formally, our alignment loss $\mathcal{L}_{\text{align}}$ is given as follows,

$$\mathcal{L}_{\text{align}} = -\frac{1}{B}\sum_{j=1}^{B} \log \frac{\exp\left(f_{\text{obj}}^j \otimes f_{\text{sent}}^j / \tau\right)}{\sum_{k=1}^{B} \exp\left(f_{\text{obj}}^j \otimes f_{\text{sent}}^k / \tau\right)} \tag{10}$$

in which, $B$ is the training batch size and $\otimes$ denotes the inner product. The learnable temperature parameter $\tau$ controls the smoothness of the distribution. For the object feature, we use the prediction token $f_p$. The sentence embedding $f_{\text{sent}}$ is one of the outputs from TFE module. To keep an identical dimension, we apply a FFN on the prediction token, i.e.,

$$f_{\text{obj}} = \text{FFN}_{\text{obj}}(f_p) \tag{11}$$

Note that the index of TCA and TCF modules in Eq. (10) and (11) is omitted for simplicity.

Thus, with the total loss $\mathcal{L}_{\text{total}}$, we can train the LGR-NET model using the back-propagation algorithm. Once the training is finished, we use the model to inference the the bounding box for given images and their referring expressions.

TABLE I
STATISTICS OF FIVE BENCHMARK DATASETS, INCLUDING REFCOCO, REFCOCO+, REFCOCOG,REFERITGAME, AND FLICKR30K ENTITIES

| Dataset | Images | Expressions | Instances |
|---|---|---|---|
| RefCOCO | 19,994 | 142,210 | 50,000 |
| RefCOCO+ | 19,992 | 141,564 | 49,856 |
| RefCOCOg | 25,799 | 95,010 | 49,822 |
| ReferItGame | 20,000 | 120,072 | 19,987 |
| Flickr30K Entities | 31,783 | 427,000 | 427,000 |

TABLE II
STATISTICS ON THE "SPATIAL EXPRESSION" IN DIFFERENT DATASETS

| Datasets | RefCOCO | RefCOCO+ | RefCOCOg | Flickr30K Entities | ReferIt Game |
|---|---|---|---|---|---|
| Proportion | 54.82% | 2.65% | 17.96% | 1.08% | 45.92% |

## IV. EXPERIMENTS

### A. Datasets

To evaluate our proposed method, we adopt five benchmark datasets, i.e., RefCOCO [16], RefCOCO+ [16], RefCOCOg [33], ReferItGame [57] and Flickr30K Entities [58]. The statistics are shown in Table I. Specifically, **1) RefCOCO/+/g** are re-annotated from MSCOCO [59]. In RefCOCO and RefCOCO+, *testA* only contains people's annotation, while *testB* contains other objects. Location words like "left" or "bottom" are forbidden in RefCOCO+. RefCOCOg contains two splits called RefCOCOg-google [33] and RefCOCOg-umd [16] and the referring expressions are longer and more complex. **2) ReferItGame** includes 20,000 images from SAIAPR12 [60]. Each image has one or more expressions for different regions. **3) Flickr30K Entities** contains images from Flickr30K [61] and enriches the annotations with short region phrases.

### B. Spatial Proportion Statistics

To reveal the influence of our proposed Coordinate Embedding on different datasets, we calculate the proportion of spatial words in the datasets. Specifically, if a referring expression contains following words which are usually utilized in the annotation: "left", "right", "top", "bottom", "middle", "o'clock", the referring expression is regarded as a "spatial expression". The statistical results are shown in Table II. The ablation results of Coordinate Embedding over different datasets are illustrated in Sec. V-C

TABLE III

EXPERIMENTAL RESULTS (ACC@0.5) ON THREE BENCHMARK DATASETS, I.E., REFCOCO, REFCOCO+, AND REFCOCOG. THE BEST PERFORMANCE IS HIGHLIGHTED IN BOLDFACE. THE SECOND BEST IS UNDERLINED

| Method | Backbone | Refcoco | | | Refcoco+ | | | Refcocog | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | *val* | *testA* | *testB* | *val* | *testA* | *testB* | *val-g* | *val-u* | *test-u* |
| **Two-stage** | | | | | | | | | | |
| VC [18] | VGG16 | - | 73.33 | 67.44 | - | 58.40 | 53.18 | 62.30 | - | - |
| MattNet [17] | ResNet-101 | 76.65 | 81.14 | 69.99 | 65.33 | 71.62 | 56.02 | - | 66.58 | 67.27 |
| LGRANs [62] | VGG16 | - | 76.60 | 66.40 | - | 64.00 | 53.40 | 61.78 | - | - |
| RvG-Tree [21] | ResNet-101 | 75.06 | 78.61 | 69.85 | 63.51 | 67.45 | 56.66 | - | 66.95 | 66.51 |
| NM-Tree [20] | ResNet-101 | 76.41 | 81.21 | 70.09 | 66.46 | 72.02 | 57.52 | 64.62 | 65.87 | 66.44 |
| DGA [19] | VGG16 | - | 78.42 | 65.53 | - | 69.07 | 51.99 | - | - | 63.28 |
| Ref-NMS [34] | ResNet-101 | 80.70 | 84.00 | 76.04 | 68.25 | 73.68 | 59.42 | - | 70.55 | 70.62 |
| **One-stage** | | | | | | | | | | |
| FAOA [22] | DarkNet-53 | 72.54 | 74.35 | 68.50 | 56.81 | 60.23 | 49.60 | 56.12 | 61.33 | 60.36 |
| RCCF [63] | DLA-34 | - | 81.06 | 71.85 | - | 70.35 | 56.32 | - | - | 65.73 |
| ReSC-Large [23] | DarkNet-53 | 77.63 | 80.45 | 72.30 | 63.59 | 68.36 | 56.81 | 63.12 | 67.30 | 67.20 |
| SAFF [64] | DarkNet-53 | 79.26 | 81.09 | 76.55 | 64.43 | 68.46 | 58.43 | - | 68.94 | 68.91 |
| LBYL-Net [65] | DarkNet-53 | 79.67 | 82.91 | 74.15 | 68.64 | 73.38 | 59.49 | 62.70 | - | - |
| **Transformer-based** | | | | | | | | | | |
| Word2Pix [66] | ResNet-101 | 81.20 | 84.39 | 78.12 | 69.46 | 76.81 | 61.57 | - | 70.81 | 71.34 |
| TransVG [27] | ResNet-101 | 81.02 | 82.72 | 78.35 | 64.82 | 70.70 | 56.94 | 67.02 | 68.67 | 67.73 |
| RefTR [28] | ResNet-101 | 82.23 | 85.59 | 76.57 | 71.58 | 75.96 | 62.16 | - | 69.41 | 69.40 |
| YORO [67] | Linear | 82.90 | 85.60 | 77.40 | 73.50 | 78.60 | 64.90 | - | 73.40 | 74.30 |
| SeqTR [68] | DarkNet-53 | 81.23 | 85.00 | 76.08 | 68.82 | 75.37 | 58.78 | - | 71.35 | 71.58 |
| QRNet [30] | Swin-S | 84.01 | 85.85 | 82.34 | 72.94 | 76.17 | 63.81 | 71.89 | 73.03 | 72.52 |
| VLTVG [31] | ResNet-101 | 84.77 | 87.24 | 80.49 | 74.19 | 78.93 | 65.17 | 72.98 | 76.04 | 74.18 |
| VLTVG(Swin) | Swin-S | 84.69 | 87.54 | 82.32 | 74.28 | 79.22 | 67.95 | 73.98 | 74.86 | 75.11 |
| VG-LAW [45] | ViT-B | 86.06 | 88.56 | **82.87** | **75.74** | 80.32 | 66.69 | - | 75.31 | 75.95 |
| TransVG++ [69] | ViT-B | **86.28** | 88.37 | 80.97 | 75.39 | 80.45 | 66.28 | 73.86 | 76.18 | 76.30 |
| Dynamic MDETR [70] | CLIP-B | 85.97 | **88.82** | 80.12 | 74.83 | **81.70** | 63.44 | 72.21 | 74.14 | 74.49 |
| Ours | | | | | | | | | | |
| LGR-NET | ResNet-101 | 83.69 | 86.42 | 79.25 | 73.50 | 78.36 | 65.02 | 71.38 | 74.14 | 74.23 |
| LGR-NET | Swin-S | 85.63 | 88.24 | 82.69 | 75.32 | 80.60 | **68.30** | **75.48** | **76.82** | **77.03** |

### C. Evaluation Metrics

We follow previous works, such as [23] and [27] and use Accuracy (**Acc@0.5**) as our evaluation metric. A prediction bounding box is regarded as true if its Intersection over Union (IoU) with the ground truth box is larger than 0.5.

### D. Implementation Detail

We set the input image size as $640 \times 640$ and the maximum length of referring expressions as 40. Specifically, we resize the image to keep the longer edge equal to 640 and pad the shorter one. For referring expressions longer than 38 (except two special tokens, *[CLS]* and *[SEP]*), we cut the tail, else we pad. The padding part of images and referring expressions are both masked during the attention computing process. We set the number of layers $N = 6$, the channel dimension $D = 256$, and the weight factor $\lambda$ as 2. We initialize the textual feature extractor with BERT-base-uncased.[1] The backbone of our visual extractor is initialized as QRNet[2] did. The parameters in other components are randomly initialized with Xavier. We use AdamW optimizer [71] with a weight decay of $10^{-4}$. We set the initial learning rate of BERT and Swin Transformer as $10^{-5}$, and that of the other components as $10^{-4}$. The learnable temperature parameter $\tau$ is initialized to 0.07 [49]. Following previous works [22], [23], [27], [63], we perform data augmentation. We train our model for 90 epochs and the learning rate is multiplied by a factor of 0.1 after 60 epochs.

For pre-training setting, we follow [28] to train our model on Visual Genome [72] for six epochs and fine-tune it on corresponding datasets for 50 epochs. The dataset contains about 100K images. We remove the images which appear in ReferItGame, Flickr30K Entities and RefCOCO/RefCOCO+/RefCOCOg's validation and test splits to avoid data leak. All experiments are conducted using eight NVIDIA V100 GPUs with the global batch size of 128. For all the experimental results, we run experiments five times with different random seeds and the error bars are within $\pm 0.5\%$. The number of our LGR-NET's parameter is 274.18M, the computation complexity is 79.87 GFLOPS. For the inference time, we randomly sample 1000 samples and compute the average value on a server with one 1080Ti GPU, the result is 60.25ms.

[1] https://huggingface.co/bert-base-uncased

[2] https://github.com/LukeForeverYoung/QRNet

## V. RESULTS AND ANALYSIS

In this section, we report the quantitative results on five benchmark datasets. Then, we report the results of ablation studies and analyze qualitative results.

### A. Main Results

We report the experimental results of our method on three benchmark datasets, including RefCOCO, RefCOCO+, and RefCOCOg in Table III. Our method achieves remarkable

performance on the three datasets for all splits. Compared with the best two-stage method Ref-NMS and one-stage method LBYL-Net, our method outperforms them by significant margins. For RefCOCO and RefCOCO+ datasets, the improvements on *testB* split are more remarkable. This demonstrates that our model achieves better cross-modal alignment especially when the referred objects are diverse. For RefCOCOg where the referring expressions are more complicated, our method still outperforms LBYL-Net 12.78% on *val-g*, and Ref-NMS 6.34% on *val-u*, *test-u* on average. This indicates that our method could perform much more accurate reasoning when facing complicated referring expressions.

In addition, we observe that the performance of our model is still superior to most recently proposed transformer-based methods from Table III. Compared to TransVG, our LGR-NET with resnet-101 [82] gets absolute improvements up to 3.70%, 8.08%, and 6.50% on RefCOCO, RefCOCO+, and RefCOCOg, respectively. It demonstrates the superiority of our LGR-NET compared to the concatenated visual-textual reasoning method. Compared with VLTVG which performs best with resnet-101, our method obtains comparable results with the same backbone. When both using a better backbone swin-s, our method outperforms VLTVG comprehensively. It means our LGR-NET achieves a better adaptability to stronger backbone. Furthermore, compared to those baselines with even stronger visual backbones like ViT-B and CLIP-B, our LGR-NET achieves competitive results, especially on RefCOCOg where the referring expressions are longer and more complex. It further demonstrates the reasoning performance of our LGR-NET when facing complicated query.

Furthermore, we report the experimental results on ReferItGame and Flickr30K Entities in Table IV. We observe that our model outperforms two-stage and one-stage methods by a large margin consistently. Compared with the best transformer-based method, our method shows comparable improvements. Note that the referring expressions in ReferItGame and Flickr30K Entities are largely simple noun phrases, which may not be suitable for exhibiting the superior reasoning ability of our method. Therefore, the improvements gained on these two datasets are less than those on RefCOCOg.

TABLE IV

EXPERIMENTAL RESULTS (ACC@0.5) ON REFERITGAME AND FLICKR30K ENTITIES. THE BEST PERFORMANCE IS HIGHLIGHTED IN BOLDFACE. THE SECOND BEST IS UNDERLINED. OUR LGR-NET WITH A STAR SYMBOL DENOTES ITS PRE-TRAINING VERSION

| Method | Backbone | ReferItGame *test* | Flickr30K *test* |
|---|---|---|---|
| **Two-stage** | | | |
| VC [18] | VGG16 | 31.13 | - |
| MattNet [17] | ResNet-101 | 29.04 | - |
| Similarity Net [73] | ResNet-101 | 34.54 | 60.89 |
| CITE [74] | ResNet-101 | 35.07 | 61.33 |
| DDPN [75] | ResNet-101 | 63.00 | 73.30 |
| DIGN [76] | VGG16 | 65.15 | 78.73 |
| **One-stage** | | | |
| FAOA [22] | DarkNet-53 | 60.67 | 68.71 |
| RCCF [63] | DLA_34 | 63.79 | - |
| ReSC-Large [23] | DarkNet-53 | 64.60 | 69.28 |
| SAFF [64] | DarkNet-53 | 66.01 | 70.71 |
| LBYL-Net [65] | DarkNet-53 | 67.47 | - |
| **Transformer-based** | | | |
| TransVG [27] | ResNet-101 | 70.73 | 79.10 |
| RefTR [28] | ResNet-101 | 71.42 | 78.66 |
| YORO [67] | Linear | 71.90 | - |
| SeqTR [68] | DarkNet-53 | 69.66 | 81.23 |
| QRNet [30] | Swin-S | 74.61 | <u>81.95</u> |
| VLTVG [31] | ResNet-101 | 71.98 | 79.84 |
| VLTVG (Swin) | Swin-S | 70.26 | 80.57 |
| VG-LAW [45] | ViT-B | **76.60** | - |
| TransVG++ [69] | ViT-B | <u>74.70</u> | 81.49 |
| Dynamic MDETR [70] | CLIP-B | 70.37 | 81.89 |
| Ours | | | |
| LGR-NET | ResNet-101 | 71.03 | 79.61 |
| LGR-NET | Swin-S | 74.64 | **81.97** |
| LGR-NET* | Swin-S | **77.78** | **82.18** |

## B. Pre-Trained Results

To compare with pre-trained methods, we use the pre-training strategy described in Sec. IV-D to pre-train our model. The pre-training baselines in Table V can be grouped into two categories. The first class includes MDETR [29] and RefTR [28], which are designed for detection or segmentation tasks and pre-trained on these tasks' datasets like VG, COCO, and Flickr 30K. Due to the model framework, the available pre-training datasets and feasible down-stream tasks are relatively limited. Other baselines belong to the second category, like OFA [46] and ONE PEACE [81]. These methods convert different kinds of multi-modal tasks as an unified sequence-to-sequence task and apply an unified framework to handle them. In this way, they can collect a variety of datasets from different tasks like VQA, Image Caption, and REC. As for REC dataset, they will convert the bounding box label into a "sentence" like $[x_{min}, y_{min}, x_{max}, y_{max}]$. Thus, these methods can utilize much more datasets for pre-training as shown in Table V.

Our pre-trained LGR-NET belongs to the first type as it only works for REC. Compared with RefTR and MDETR, our pre-trained LGR-NET achieves better performance with same or less pre-trained data. It demonstrates the scalability of our model. Compared with the second type of methods like ONE-PEACE or mPLUG, our method shows lower performance. It means a generic cross-modal framework pre-trained by a variety of datasets owns a higher upper limit.

## C. Ablation Study

We conduct ablation experiments to further analyze our LGR-NET. To this aim, we choose RefCOCOg as our dataset. The reason is that compared with the other four datasets, RefCOCOg contains longer referring expressions. This requires REC models to perform more complex reasoning. Specifically, we consider five aspects of ablation as follows.

*1) On Coordinate Embedding:* To verify the effectiveness of our Coordinate Embedding (CE), we conduct ablation experiments on RefCOCO/+/g whose proportions of the "spatial expression" vary obviously as shown in Table II. The ablation results are shown in Table VI. We can find that due to the low proportion of the "spatial expression" in RefCOCO+, the improvements by our CE are insignificant. By contrast, the improvements over RefCOCO and RefCOCOg are obvious, especially on RefCOCO testB which has a high proportion of spatial expression, our CE brings 1.77% improvement.

TABLE V

EXPERIMENTAL RESULTS IN OF PRE-TRAINING SETTING (ACC@0.5) ON THREE BENCHMARK DATASETS, I.E., REFCOCO, REFCOCO+, AND REFCOCOG. THE BEST PERFORMANCE IS HIGHLIGHTED IN BOLDFACE. WE REPORT THE SIZE OF PRE-TRAIN IMAGES USED IN VARIOUS METHODS. OUR LGR-NET WITH A STAR SYMBOL DENOTES THE PRE-TRAINING VERSION

| Method | Backbone | Pre-train Images | RefCOCO | | | RefCOCO+ | | | RefCOCOg | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | *val* | *testA* | *testB* | *val* | *testA* | *testB* | *val-g* | *val-u* | *test-u* |
| VilBERT [77] | ResNet-101 | 3.3M | - | - | - | 72.34 | 78.52 | 62.61 | - | - | - |
| UNITER_L [78] | ResNet-101 | 4.6M | 81.41 | 87.04 | 74.17 | 75.90 | 81.45 | 66.70 | - | 74.86 | 75.77 |
| RefTR [28] | ResNet-101 | 100k | 85.65 | 88.73 | 81.16 | 77.55 | 82.26 | 68.99 | - | 79.25 | 80.01 |
| MDETR [29] | ResNet-101 | 200k | 86.75 | 89.58 | 81.41 | 79.52 | 84.09 | 70.62 | - | 81.64 | 80.89 |
| OFA [46] | ResNet-152 | 20M | 90.05 | 92.93 | 85.26 | 85.80 | 89.87 | 79.22 | - | 85.89 | 86.55 |
| Shikra [79] | ViT-L | - | 87.83 | 91.11 | 81.81 | 82.89 | 87.79 | 74.41 | - | 82.64 | 83.16 |
| m-PLUG [80] | ViT-L | 14M | 92.40 | **94.51** | 88.42 | 86.02 | 90.17 | 78.17 | - | 85.88 | 86.42 |
| ONE-PEACE [81] | ViT-g | 1.5B | **92.58** | 94.18 | **89.26** | **88.77** | **92.21** | **83.23** | **-** | **89.22** | **89.27** |
| Our LGR-NET* | Swin-S | 100k | 88.16 | 90.01 | 84.26 | 78.66 | 83.51 | 73.18 | 80.94 | 82.60 | 82.58 |

TABLE VI

ABLATION RESULTS FOR COORDINATE EMBEDDING (CE) ON DIFFERENT DATASETS

| Models | RefCOCO | | RefCOCO+ | | RefCOCOg | |
|---|---|---|---|---|---|---|
| | testA | testB | testA | testB | val-u | test-u |
| LGR-NET(w/o CE) | 87.16 | 80.92 | 80.11 | 68.30 | 75.47 | 75.91 |
| LGR-NET | **88.24** | **82.69** | **80.60** | **68.30** | **76.82** | **77.03** |
| Δ | +1.08 | +1.77 | +0.49 | +0 | +1.35 | +1.12 |

TABLE VII

ABLATION RESULTS FOR DIFFERENT OPERATIONS TO INCORPORATE THE COORDINATE EMBEDDING

| Models | RefCOCOg | |
|---|---|---|
| | val-u | test-u |
| LGR-NET (concatenation) | 76.76 | 76.90 |
| LGR-NET (multiply) | 76.63 | 76.49 |
| LGR-NET | **76.82** | **77.03** |

TABLE VIII

INFLUENCE OF BATCH SIZE ON THE PERFORMANCE OF CROSS-MODAL LOSS (CL). "BS" REPRESENTS BATCH SIZE

| Models | RefCOCOg | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | val-u (bs= 16, 32, 64, 128) | | | | test-u (bs= 16, 32, 64, 128) | | | |
| LGR-NET(w/o CL) | 74.38 | 75.94 | 75.28 | 75.71 | 75.29 | 75.32 | 75.59 | 75.76 |
| LGR-NET | 74.57 | 76.47 | 76.43 | **76.82** | 75.33 | 75.98 | 76.65 | **77.03** |

In addition, we conduct ablation experiments to explore the influence of different operations to incorporate the CE. The results are shown in Table VII. For "concatenation", we concatenate the prediction token and the CE, then apply a linear layer to project them into the original prediction token's dimension. We can find the "add" operation gets the best performance.

*2) On Cross-Modal Loss:* As our Cross-modal Loss (CL) is a form of contrastive loss which benefits from a large batch size [49], we conduct ablation experiments to ascertain the impact of batch size on CL. The results are shown in Table VIII. We can find that when we enlarge the batch size, the improvements from CL get larger. Specifically, when the batch size is 16, the CL only brings 0.12% improvement on average; when we set batch size as 128, it raises to 1.19%. Furthermore, the performance of our LGR-NET increases more significantly than LGR-NET without CL when enlarging the batch size from 16 to 128. The former improves 1.98% on average, while when without CL, it's only 0.9%. Overall, the results demonstrate that when enlarging the batch size, our CL brings more significant improvements.

TABLE IX

JOINT ABLATION RESULTS OF CE AND CL

| Coordinate Embedding | Cross-modal Loss | RefCOCOg | |
|---|---|---|---|
| | | *val-u* | *test-u* |
| ✗ | ✗ | 75.40 | 75.63 |
| ✗ | ✓ | 75.47 | 75.91 |
| ✓ | ✗ | 75.71 | 75.76 |
| ✓ | ✓ | **76.82** | **77.03** |

In addition, our CL is implemented between the *[CLS]* token and the prediction token which captures referred object's visual feature rather than the whole image's feature. It has been demonstrated that the CE plays a positive role when the prediction token captures the referred object's visual feature. To verify the performance of the combination of the two components, we conduct ablation experiments on both of them. The results are shown in Table IX. We can find that the CE and CL bring only 0.31% and 0.07% improvements on RefCOCOg *val-u* without each other, respectively, while the improvements rise to 1.35% and 1.11% with each other. Therefore, the ablation results not only demonstrate the effectiveness of our CE and CL, but also indicate that they complement each other.

*3) On TCA and TCF Modules:* Actually, our TCA and TCF can be viewed as a standard transformer cross-attention block with custom input, while we argue that the input manner of the two modalities is important.

To verify this, we firstly design ablation experiment on QRNet. Specifically, for all the visual-linguistic transformer encoder layers, we replace the visual or textual part of the concatenated features with visual or textual features output from the corresponding extractor, respectively. The results are reported in Table X. In the first row, the visual and textual features are input once corresponding to QRNet. When we repeatedly input visual features, namely keep visual features fixed, we observe a significant performance decline, 2.18% accuracy drop on *val-u* split. This means the alignment of visual features is indispensable for cross-modal reasoning. In contrast, we obtain consistent improvements when we repeatedly input textual features, 1.40% and 1.30% on *val-u* and *test-u*, respectively. This means that repeatedly fused

TABLE X
ABLATION RESULTS ON MODALITIES INPUT MANNER

| Repeatedly input visual features | Repeatedly input textual features | RefCOCOg *val-u* | RefCOCOg *test-u* |
|---|---|---|---|
| ✗ | ✗ | 73.03 | 72.52 |
| ✓ | ✗ | 70.85 | 72.21 |
| ✓ | ✓ | 70.14 | 69.44 |
| ✗ | ✓ | **74.43** | **73.82** |

TABLE XI
ABLATION OF TCA AND TCF

| Models | RefCOCOg va-u | RefCOCOg test-u |
|---|---|---|
| LGR-NET(text-attn-img) | 75.06 | 75.21 |
| LGR-NET | **76.82** | **77.03** |

TABLE XII
ABLATION RESULTS ABOUT THE NUMBER OF PREDICTION TOKEN

| Number of prediction token | RefCOCOg val-u | RefCOCOg test-u |
|---|---|---|
| 1 | 76.82 | 77.03 |
| 2 | 77.10 | 76.84 |
| 3 | 76.45 | 76.28 |

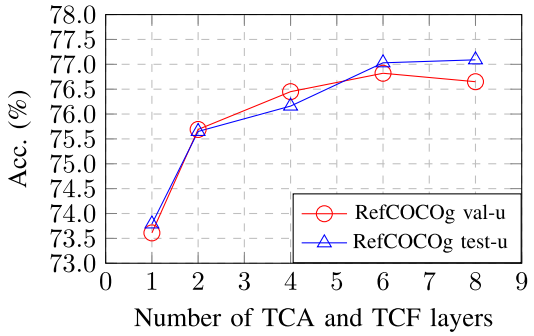



Fig. 3. Effect of the number of TCA and TCF layers.

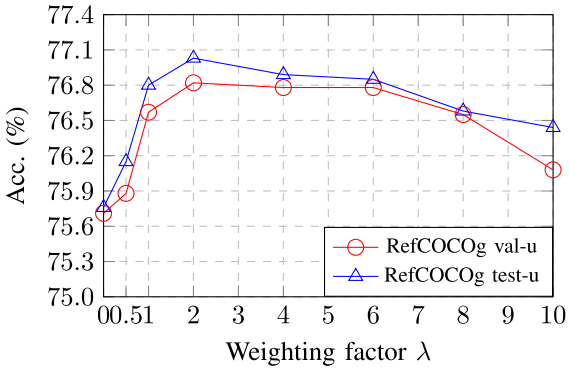



Fig. 4. Effect of the weighting factor λ for our loss.

textual features play an important role on reasoning guidance. It needs to be noted that when the visual and textual features are both repeatedly input, the cross-modal module degenerates to a single layer transformer, and naturally leads to worse performance. Based on this, we set the modalities input manner of our LGR-NET as described in Sec. III.

In addition, we exchange the visual and textual input in our LGR-Net and name it LGR-NET(text-attn-img), as shown in Fig. 8. Ablation results are shown in Table XI. We can find that when exchanging the input of visual and textual modalities, the performance drops consistently and significantly, which demonstrates the effectiveness of our TCA and TCF. Further visualization analysis is provided in Sec. V-D (4).

*4) On the Number of Layer and Prediction Token:* We evaluate our model with different number of layers in the TCA and TCF modules. The experimental results are shown in Fig. 3. The accuracy on RefCOCOg *val-u* and *test-u* increases consistently from one layer (73.61% and 73.79%) to six layers (76.90% and 76.71%). This demonstrates that the fusion of textual features with multiple layers can enhance the effect of language guidance, leading to better performance. When setting more layers, we observe that the accuracy improves inconsistently, with a little drop on *val-u* split. Therefore, we choose the model with six layers as our final implementation.

In addition, to reveal the impact of the number of prediction token, we set different numbers of them to conduct experiments. The results are shown in Table XII. Specifically, for 2 or more tokens, we utilize the mean pooling of them to predict the bounding box. We can find that the number of PT as 2 brings 0.28% improvement on *val-u* but 0.19% decline on *test-u*, which is insignificant and inconsistent. When we set 3 tokens, performance on both splits decreases. Therefore, setting one prediction token for bounding box prediction is intuitive and better.

*5) On the Loss Weight λ:* To investigate the impact of our cross-modal loss, we evaluate our LGR-NET with different values of weighting factor λ in Eq. 8. As shown in Fig. 4, we notice that when the weighting factor $\lambda = 2.0$, the model achieves the best performance on both splits. In addition, we observe that all the values of hyper-parameter λ bring performance improvements, compared with λ equals to zero. This demonstrates the effectiveness of our cross-modal loss.

### D. Visualization Analysis

*1) Language Guidance:* To show the effectiveness of our LGR-NET, we visualize the language guidance from the following three aspects. The visualization results are shown in Fig. 5. First, in the first column, we visualize the 2D coordinate generated by textual features from the last cross-modal alignment and fusion layer. The generated points (marked in blue) correctly hit the referred objects. Take the second row as an example. The point hits the correct orange corresponds to the 2D coordinate (0.77, 0.77). This demonstrates that our CE can help the prediction token to capture key visual features.

Second, to understand the reasoning process we visualize the attention score between the prediction token and visual tokens from layer 2, 4 and 6. In addition, we visualize the attention score between the prediction token and textual tokens from the sixth layer. Note that words such as "zebras" or "o'clock" are split by the tokenizer. We use mean pooling to combine them. Take the bottom as an example. We observe that the prediction token gradually attends to the correct orange

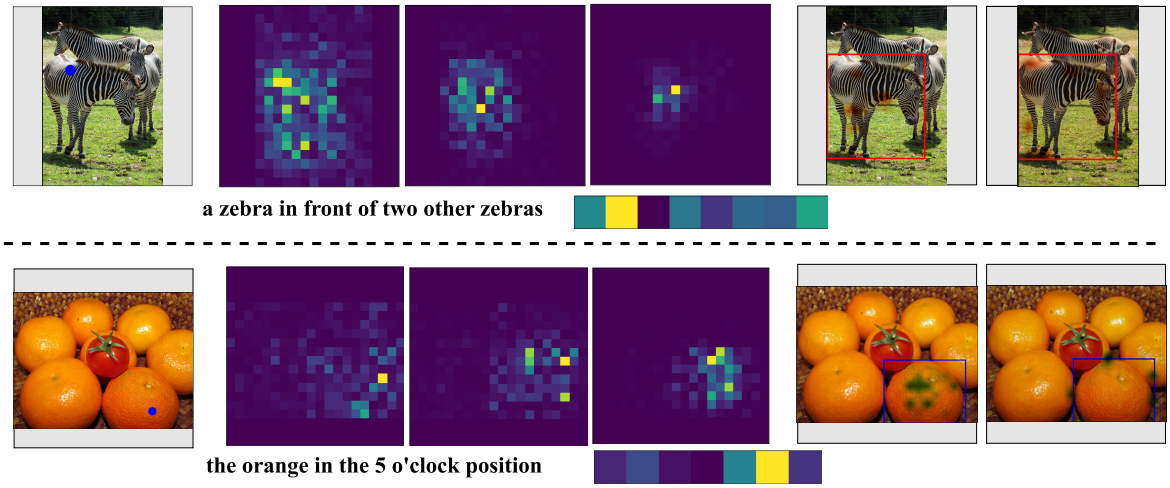



Fig. 5. Visualization of the language guidance of our LGR-NET by two examples. *Column 1*: The 2D coordinates generated from textual features are marked in images. We map the normalized coordinates to blue dots. The coordinate $(0, 0)$ means the top left, and $(1, 1)$ means the bottom right of an image (including padded mask, shown in gray). *Column 2-4*: The prediction token's attention scores on visual tokens from layer 2, 4 and 6, and that of textual tokens from layer 6 are shown. *Column 5-6*: The attention scores of the prediction token on the visual tokens from layer 6 are shown. The weights are obtained from 5 heads out of 8 attention heads. The fifth column comes from one head, and the sixth column combines the other four head. They highlight regions inside the predicted boxes and four extremities, respectively.

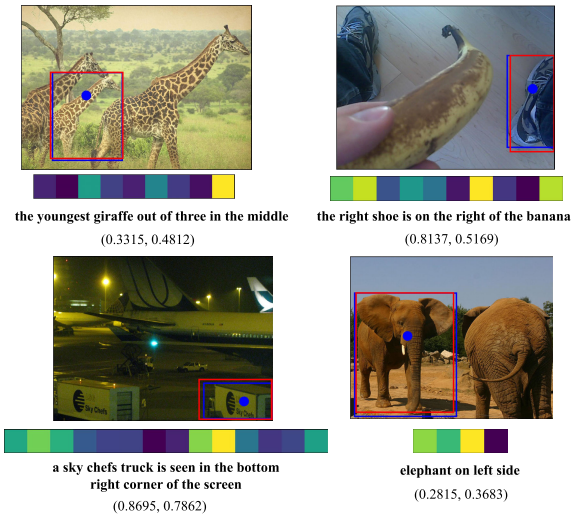



Fig. 6. Visualization of the CE. For each example, the first line is the image with predicted result by our LGR-NET (red box) and the ground truth (blue box); the second line is the word-level attention between the CE and the query; the third line is the query; the fourth line is the 2D coordinate, it's visualized in the image (blue point).

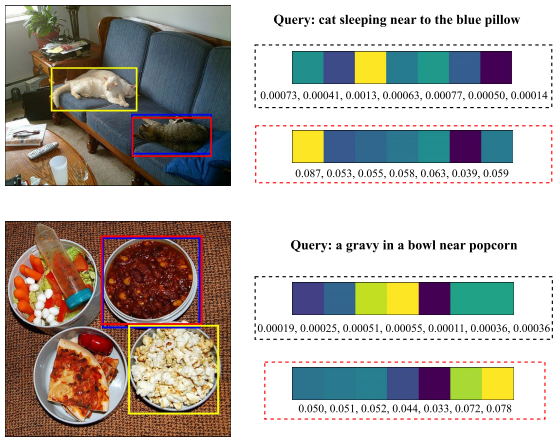



Fig. 7. The attention visualization result comparison between QRNet and our LGR-NET. For each example, the left shows the image and the predicted results by QRNet (yellow box), LGR-NET (red box) and the ground truth (blue box); the right shows the query and the attention score between prediction token and query, QRNet's results are in the black dashed boxes, our LGR-NET's results are in the red dashed boxes. We remove the attention scores over [CLS] and [SEP] tokens and average the scores over words that split by the tokenizer.

highly-related to the "5 o'clock position". In addition, the attention scores on textual tokens show that the prediction token focuses on the key expression "5 o'clock".

Third, to reveal the visual features captured by the prediction token, we visualize the attention weights between the prediction token and visual tokens from 8 attention heads. We show five attention maps. In the fifth column, one of the five maps highlights some regions inside the box, like the highlighted orange (the bottom). In the sixth column, the other four maps highlight the referred object's extremities. These extremities often correspond to the four edges of the predicted boxes, such as the zebra's feet and head (the top). This demonstrates that the prediction token captures two types of features. One is the feature of the referred object's extremities for box prediction. The other is the representative feature of the object corresponding to the referring expression. The analysis results are consistent with our designed loss.

*2) Coordinate Embedding on Query:* To further show the effectiveness of our Coordinate Embedding (CE), we visualize the attention score between the CE and query, and the results are shown in Fig. 6. We can find that the CE indeed helps for capturing spatial information. For instance, in the first example, we can find that the CE pays more attention on the spatial word "middle", which is the key spatial information for localizing the correct giraffe. In addition, when facing a more complex query in example 3, the CE attends to the key spatial expression "bottom right corner" correctly as well. Overall, the CE can help the prediction token capture necessary spatial information from the referring expression and promote to attend to the correct object.

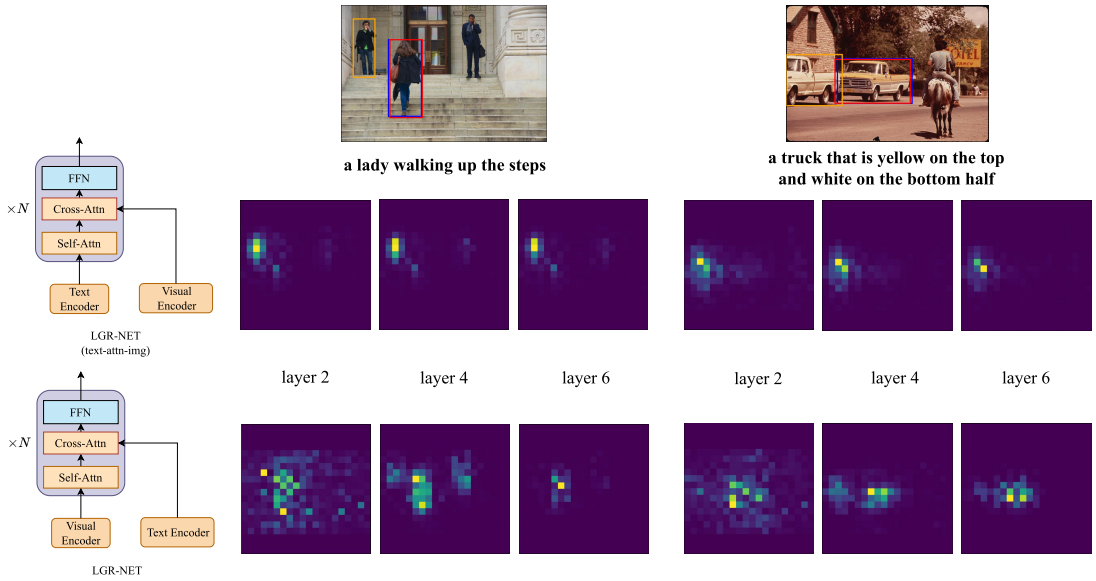



Fig. 8. Visualization comparison between our LGR-NET and LGR-NET (text-attn-img), the inputs of them are exchanged. Each case includes the image and the query on the top, the boxes in blue, red, orange represent the ground truth, LGR-NET's result and LGR-NET (text-attn-img)'s result; on the bottom are corresponding attention maps between the prediction token and visual tokens from layer 2, 4, 6.

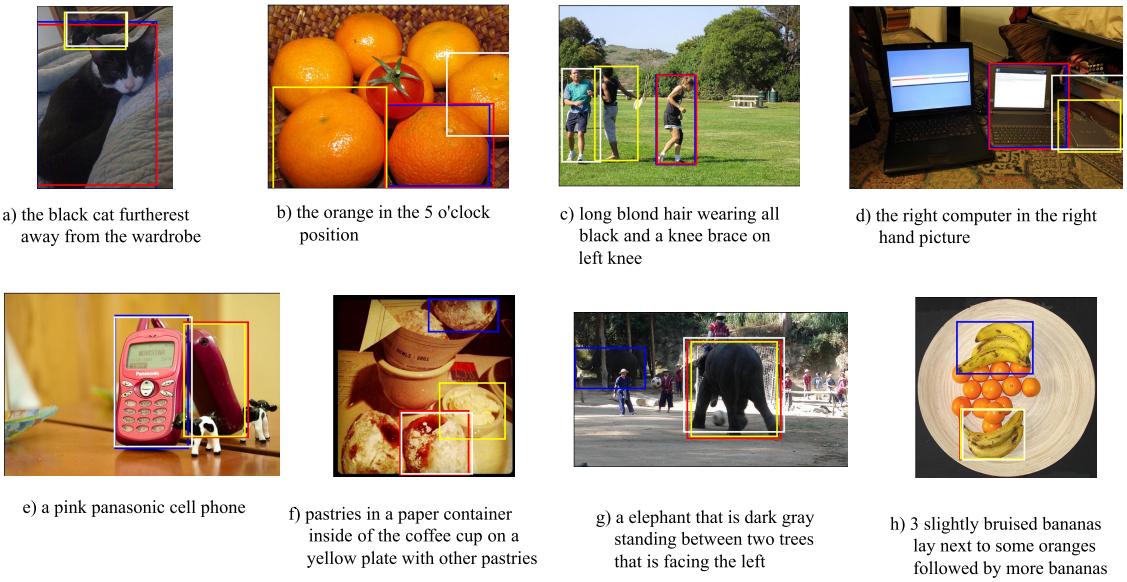



Fig. 9. Qualitative results of our LGR-NET compared with those of TransVG and QRNet from RefCOCOg. The bounding boxes in blue, red, white, and yellow are used for the results of ground-truth, our LGR-NET, TransVG, and QRNet, respectively. At the top our model correctly predicts the results. In contrast, our LGR-NET fails at the bottom.

*3) Overwhelmed Language Information:* To reveal the language overwhelming problem as mentioned in Sec. I, we compare the visualization results between QRNet and our LGR-NET in Fig. 7. The main difference between them is the cross-modal reasoning module as illustrated in Fig. 1. For QRNet, it applies self-attention over concatenated tokens $[f_p, F_L; F_V]$ (i.e., prediction token, textual sequences and visual sequences). As the length of $F_L$ is only 40, it's much smaller than the length of $F_V$ which is $\frac{H}{32} \times \frac{W}{32} + \frac{H}{64} \times \frac{W}{64} = 500$ ($H = W = 640$). Then the attention scores between $f_p$ and other tokens $\in \mathbb{R}^{1\times(1+40+500)}$ are distributed by the whole long sequence, it becomes quite small after the softmax activation function. We can find that the attention scores in QRNet are about two orders of magnitude smaller than ours. In this way, the textual information captured by the prediction token is insufficient. Therefore, we decouple the two modalities, taking into account the characteristics of REC and mathematically aligning with our language-guided motivation.

*4) TCA and TCF Analysis:* To further reveal the effectiveness of our TCA and TCF, we compare the attention map on visual tokens between our LGR-NET and LGR-NET (text-attn-img) as shown in Fig. 8. Specifically, in LGR-NET, the attention map comes from self-attn sublayer, in LGR-NET (text-attn-img) the attention map comes from cross-attn sublayer. In our LGR-NET, the TCA (visual tokens self-attention) will align the visual features with the textual feature by adjusting the attention weight on different visual tokens. We can find the attention map reveals the prediction token generally attends to referred object during layers. While in LGR-NET (text-attn-img), the attention map keeps relatively

fixed without adjustment, which cannot eliminate the impact of other objects' interference dynamically.

### E. Qualitative Analysis

In Fig. 9, we show some examples from RefCOCOg of our model, TransVG and QRNet. In case a, our model attends to the entire key expression "furtherest away from the wardrobe" and predicts the nearby cat correctly, rather than the single word "furtherest" which leads to the wrong result. In addition, our method performs better when complex reasoning is required. As case b shows, our LGR-NET conducts accurate reasoning and identifies the correct orange according to the "5 o'clock" expression. For case e, LGR-NET and QRNet predict wrong, while TransVG makes a correct prediction. This means that Swin Transformer does not always perform better than ResNet. For cases f to h, all models do not perform well. For cases f and g, several nested expressions and the referred "pasties" and "elephant" mixed in complex scenes make them challenging.

## VI. Conclusion and Future Work

In this paper, we propose LGR-NET for REC task. Our LGR-NET emphasizes the guidance of textual features for cross-modal reasoning from three aspects. The text-generated coordinate embedding helps the prediction token capture key visual features. The textual features are employed for alternated cross-modal reasoning. The novel cross-modal loss enhances the cross-modal alignment while localizing the referred object. Experimental results on the five popular benchmark datasets demonstrate the effectiveness of our LGR-NET.

In the future, implementing our LGR-NET in practical applications is a challenging and meaningful direction. Currently, our LGR-NET has only been evaluated on benchmark datasets. However, its performance with images from different domains, such as sketches and animations, remains unknown. In addition, when a query refers to nothing, a practical model should not predict a bounding box but output "no object". Therefore, developing a robust REC model capable of handling different image domains and arbitrary queries is crucial for practical applications.

## Acknowledgment

The authors would like to thank the editors and anonymous reviewers for their valuable comments on improving the final version of this article.

## References

[1] JATS Formatted O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3156–3164.

[2] P. Anderson et al., "Bottom-up and top-down attention for image captioning and visual question answering," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6077–6086.

[3] V.-Q. Nguyen, M. Suganuma, and T. Okatani, "GRIT: Faster and better image captioning transformer using dual visual features," in *Proc. 17th Eur. Conf. Comput. Vis. (ECCV)*, 2022, pp. 167–184.

[4] J. Gu, H. Wang, and R. Fan, "Coherent visual storytelling via parallel top-down visual and topic attention," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 1, pp. 257–268, Jan. 2023.

[5] S. Antol et al., "VQA: Visual question answering," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Santiago, Chile, Dec. 2015, pp. 2425–2433.

[6] D.-K. Nguyen and T. Okatani, "Improved fusion of visual and language representations by dense symmetric co-attention for visual question answering," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6087–6096.

[7] S. Karamcheti, R. Krishna, L. Fei-Fei, and C. Manning, "Mind your outliers! Investigating the negative impact of outliers on active learning for visual question answering," in *Proc. 59th Annu. Meeting Assoc. Comput. Linguistics, 11th Int. Joint Conf. Natural Lang. Process.*, 2021, pp. 7265–7281.

[8] F. Zhang, R. Wang, F. Zhou, and Y. Luo, "ERM: Energy-based refined-attention mechanism for video question answering," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 3, pp. 1454–1467, Mar. 2023.

[9] P. Anderson et al., "Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments," in *Proc. CVPR*, Salt Lake City, UT, USA, Jun. 2018, pp. 3674–3683.

[10] Y. Hong, C. Rodriguez, Q. Wu, and S. Gould, "Sub-instruction aware vision-and-language navigation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2020, pp. 3360–3376.

[11] W. Zhu et al., "Diagnosing vision-and-language navigation: What really matters," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Human Lang. Technol.*, 2022, pp. 5981–5993.

[12] W. Zhang, C. Ma, Q. Wu, and X. Yang, "Language-guided navigation via cross-modal grounding and alternate adversarial learning," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 9, pp. 3469–3481, Sep. 2021.

[13] R. Hu, M. Rohrbach, and T. Darrell, "Segmentation from natural language expressions," in *Proc. Comput. Vis.–ECCV 14th Eur. Conf.*, Amsterdam, The Netherlands. Cham, Switzerland: Springer, Oct. 2016, pp. 108–124.

[14] H. Li, M. Sun, J. Xiao, E. G. Lim, and Y. Zhao, "Fully and weakly supervised referring expression segmentation with end-to-end learning," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 10, pp. 5999–6012, Jun. 2023.

[15] C. Shang et al., "Cross-modal recurrent semantic comprehension for referring image segmentation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 7, pp. 3229–3242, Dec. 2023.

[16] V. K. Nagaraja, V. I. Morariu, and L. S. Davis, "Modeling context between objects for referring expression understanding," in *Proc. Comput. Vis. ECCV 14th Eur. Conf.*, Amsterdam, The Netherlands, vol. 9908, Oct. 2016, pp. 792–807.

[17] L. Yu et al., "MAttNet: Modular attention network for referring expression comprehension," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1307–1315.

[18] H. Zhang, Y. Niu, and S.-F. Chang, "Grounding referring expressions in images by variational context," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4158–4166.

[19] S. Yang, G. Li, and Y. Yu, "Dynamic graph attention for referring expression comprehension," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Seoul, South Korea, Oct. 2019, pp. 4643–4652.

[20] D. Liu, H. Zhang, Z. Zha, and F. Wu, "Learning to assemble neural module tree networks for visual grounding," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Seoul, South Korea, Oct. 2019, pp. 4672–4681.

[21] R. Hong, D. Liu, X. Mo, X. He, and H. Zhang, "Learning to compose and reason with language tree structures for visual grounding," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 2, pp. 684–696, Feb. 2022.

[22] Z. Yang, B. Gong, L. Wang, W. Huang, D. Yu, and J. Luo, "A fast and accurate one-stage approach to visual grounding," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Seoul, South Korea, Oct. 2019, pp. 4682–4692.

[23] Z. Yang, T. Chen, L. Wang, and J. Luo, "Improving one-stage visual grounding by recursive sub-query construction," in *Proc. Comput. Vis. (ECCV) 16th Eur. Conf.*, in Lecture Notes in Computer Science, Glasgow, U.K., vol. 12359, 2020, pp. 387–404.

[24] M. Sun, W. Suo, P. Wang, Y. Zhang, and Q. Wu, "A proposal-free one-stage framework for referring expression comprehension and generation via dense cross-attention," *IEEE Trans. Multimedia*, vol. 25, pp. 2446–2458, 2023.

[25] S. Ren, K. He, R. B. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.

[26] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.

[27] J. Deng, Z. Yang, T. Chen, W. Zhou, and H. Li, "TransVG: End-to-end visual grounding with transformers," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 1749–1759.

[28] M. Li and L. Sigal, "Referring transformer: A one-step approach to multi-task visual grounding," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 19652–19664.

[29] A. Kamath, M. Singh, Y. LeCun, G. Synnaeve, I. Misra, and N. Carion, "MDETR–modulated detection for end-to-end multi-modal understanding," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 1760–1770.

[30] J. Ye et al., "Shifting more attention to visual backbone: Query-modulated refinement networks for end-to-end visual grounding," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 15481–15491.

[31] L. Yang, Y. Xu, C. Yuan, W. Liu, B. Li, and W. Hu, "Improving visual grounding with visual-linguistic verification and iterative reasoning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 9489–9498.

[32] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders, "Selective search for object recognition," *Int. J. Comput. Vis.*, vol. 104, no. 2, pp. 154–171, Sep. 2013.

[33] J. Mao, J. Huang, A. Toshev, O. Camburu, A. Yuille, and K. Murphy, "Generation and comprehension of unambiguous object descriptions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 11–20.

[34] L. Chen, W. Ma, J. Xiao, H. Zhang, and S. Chang, "Ref-NMS: Breaking proposal bottlenecks in two-stage referring expression grounding," in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 1036–1044.

[35] A. Rohrbach, M. Rohrbach, R. Hu, T. Darrell, and B. Schiele, "Grounding of textual phrases in images by reconstruction," in *Proc. Comput. Vis. (ECCV)*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. 2016, pp. 817–834.

[36] M. Sun, J. Xiao, E. G. Lim, S. Liu, and J. Y. Goulermas, "Discriminative triad matching and reconstruction for weakly referring expression grounding," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 11, pp. 4189–4195, Nov. 2021.

[37] M. Sun, J. Xiao, E. G. Lim, and Y. Zhao, "Cycle-free weakly referring expression grounding with self-paced learning," *IEEE Trans. Multimedia*, vol. 25, pp. 1611–1621, 2023.

[38] M. Sun, J. Xiao, and E. G. Lim, "Iterative shrinking for referring expression grounding using deep reinforcement learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 14055–14064.

[39] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inform. Process. Syst.*, 2017, pp. 5998–6008.

[40] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.* Minneapolis, MI, USA: Association for Computational Linguistics, vol. 1, Jun. 2019, pp. 4171–4186.

[41] A. Dosovitskiy et al., "An image is worth 16×16 words: Transformers for image recognition at scale," in *Proc. 9th Int. Conf. Learn. Represent. (ICLR)*, Virtual Event, Austria, May 2021.

[42] L. He et al., "End-to-end video object detection with spatial–temporal transformers," 2021, *arXiv:2105.10920*.

[43] Z. Liu et al., "Swin Transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Montreal, QC, Canada, Oct. 2021, pp. 9992–10002.

[44] L. Dai, H. Liu, H. Tang, Z. Wu, and P. Song, "AO2-DETR: Arbitrary-oriented object detection transformer," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 5, pp. 2342–2356, May 2023.

[45] W. Su et al., "Language adaptive weight generation for multi-task visual grounding," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 10857–10866.

[46] P. Wang et al., "OFA: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework," in *Proc. Int. Conf. Mach. Learn. (ICML)*, vol. 162, Baltimore, MD, USA, 2022, pp. 23318–23340.

[47] W. Su et al., "VL-BERT: Pre-training of generic visual-linguistic representations," in *Proc. 8th Int. Conf. Learn. Represent. (ICLR)*, Addis Ababa, Ethiopia, Apr. 2020, pp. 1–16.

[48] L. H. Li, M. Yatskar, D. Yin, C. Hsieh, and K. Chang, "VisualBERT: A simple and performant baseline for vision and language," 2019, *arXiv:1908.03557*.

[49] A. Radford et al., "Learning transferable visual models from natural language supervision," in *Proc. 38th Int. Conf. Mach. Learn.*, vol. 139, 2021, pp. 8748–8763.

[50] J. Li, D. Li, C. Xiong, and S. Hoi, "BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation," in *Proc. Int. Conf. Mach. Learn.*, Baltimore, MD, USA, 2022, pp. 12888–12900.

[51] J. Li, D. Li, S. Savarese, and S. C. H. Hoi, "BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," 2023, *arXiv:2301.12597*.

[52] M. U. Khattak, H. Rasheed, M. Maaz, S. Khan, and F. S. Khan, "MaPLe: Multi-modal prompt learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 19113–19122.

[53] D. Zhu, J. Chen, X. Shen, X. Li, and M. Elhoseiny, "MiniGPT-4: Enhancing vision-language understanding with advanced large language models," 2023, *arXiv:2304.10592*.

[54] S. Zhang et al., "OPT: Open pre-trained transformer language models," 2022, *arXiv:2205.01068*.

[55] L. Zheng, "Judging LLM-as-a-judge with MT-bench and chatbot arena," 2023, *arXiv:2306.05685*.

[56] H. Rezatofighi et al., "Generalized intersection over union: A metric and a loss for bounding box regression," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, Jun. 2019, pp. 658–666.

[57] S. Kazemzadeh, V. Ordonez, M. Matten, and T. Berg, "ReferItGame: Referring to objects in photographs of natural scenes," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, Doha, Qatar: Association for Computational Linguistics, 2014, pp. 787–798.

[58] B. A. Plummer, L. Wang, C. M. Cervantes, J. C. Caicedo, J. Hockenmaier, and S. Lazebnik, "Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Santiago, CL, USA, Dec. 2015, pp. 2641–2649.

[59] T. Lin et al., "Microsoft COCO: Common objects in context," in *Proc. Comput. Vis. ECCV 13th Eur. Conf.*, Zurich, Switzerland, vol. 8693, Sep. 2014, pp. 740–755.

[60] H. J. Escalante et al., "The segmented and annotated IAPR TC-12 benchmark," *Comput. Vis. Image Understand.*, vol. 114, no. 4, pp. 419–428, Apr. 2010.

[61] P. Young, A. Lai, M. Hodosh, and J. Hockenmaier, "From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions," *Trans. Assoc. Comput. Linguistics*, vol. 2, pp. 67–78, Dec. 2014.

[62] P. Wang, Q. Wu, J. Cao, C. Shen, L. Gao, and A. V. D. Hengel, "Neighbourhood watch: Referring expression comprehension via language-guided graph attention networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, Jun. 2019, pp. 1960–1968.

[63] Y. Liao et al., "A real-time cross-modality correlation filtering method for referring expression comprehension," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, Jun. 2020, pp. 10877–10886.

[64] J. Ye, X. Lin, L. He, D. Li, and Q. Chen, "One-stage visual grounding via semantic-aware feature filter," in *Proc. 29th ACM Int. Conf. Multimedia*, Oct. 2021, pp. 1702–1711.

[65] B. Huang, D. Lian, W. Luo, and S. Gao, "Look before you leap: Learning landmark features for one-stage visual grounding," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 16883–16892.

[66] H. Zhao, J. T. Zhou, and Y.-S. Ong, "Word2Pix: Word to pixel cross-attention transformer in visual grounding," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 2, pp. 1523–1533, Feb. 2022.

[67] C. Ho, S. Appalaraju, B. Jasani, R. Manmatha, and N. Vasconcelos, "YORO—Lightweight end to end visual grounding," 2022, *arXiv:2211.07912*.

[68] C. Zhu et al., "SeqTR: A simple yet universal network for visual grounding," in *Proc. Eur. Conf. Comput. Vis.*, Cham, Switzerland: Springer, 2022, pp. 598–615.

[69] J. Deng et al., "TransVG++: End-to-end visual grounding with language conditioned vision transformer," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 11, pp. 13636–13652, Nov. 2023.

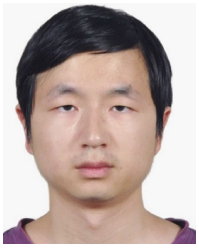
[70] F. Shi, R. Gao, W. Huang, and L. Wang, "Dynamic MDETR: A dynamic multimodal transformer decoder for visual grounding," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 2, pp. 1181–1198, Feb. 2024.

[71] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *Proc. 7th Int. Conf. Learn. Represent.*, New Orleans, LA, USA, 2019, pp. 1–12.

[72] R. Krishna et al., "Visual genome: Connecting language and vision using crowdsourced dense image annotations," *Int. J. Comput. Vis.*, vol. 123, no. 1, pp. 32–73, May 2017.
[73] L. Wang, Y. Li, J. Huang, and S. Lazebnik, "Learning two-branch neural networks for image-text matching tasks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 2, pp. 394–407, Feb. 2019.
[74] B. A. Plummer et al., "Conditional image-text embedding networks," in *Proc. Comput. Vis. ECCV 15th Eur. Conf.*, Munich, Germany, vol. 11216, Sep. 2018, pp. 258–274.
[75] Z. Yu, J. Yu, C. Xiang, Z. Zhao, Q. Tian, and D. Tao, "Rethinking diversified and discriminative proposal generation for visual grounding," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Stockholm, Sweden, Jul. 2018, pp. 1114–1120.
[76] Z. Mu, S. Tang, J. Tan, Q. Yu, and Y. Zhuang, "Disentangled motif-aware graph learning for phrase grounding," in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 13587–13594.
[77] J. Lu, D. Batra, D. Parikh, and S. Lee, "ViLBERT: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 13–23.
[78] Y. Chen et al., "UNITER: Universal image-text representation learning," in *Proc. Comput. Vis. (ECCV) 16th Eur. Conf.*, Glasgow, U.K., vol. 12375, Aug. 2020, pp. 104–120.
[79] K. Chen, Z. Zhang, W. Zeng, R. Zhang, F. Zhu, and R. Zhao, "Shikra: Unleashing multimodal LLM's referential dialogue magic," 2023, *arXiv:2306.15195*.
[80] C. Li et al., "mPLUG: Effective and efficient vision-language learning by cross-modal skip-connections," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2022, pp. 7241–7259.
[81] P. Wang et al., "ONE-PEACE: Exploring one general representation model toward unlimited modalities," 2023, *arXiv:2305.11172*.
[82] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

**Mingcong Lu** received the B.E. degree in intelligent science and technology from Beijing University of Posts and Telecommunications (BUPT), China, in 2021, where he is currently pursuing the master's degree with the School of Artificial Intelligence. His research interests include multimedia information processing and machine learning.

**Ruifan Li** (Member, IEEE) received the B.S. and M.S. degrees in control systems, and in circuits and systems from the Huazhong University of Science and Technology, Wuhan, China, in 1998 and 2001, respectively, and the Ph.D. degree in signal and information processing from Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 2006. He is currently an Associate Professor with the School of Artificial Intelligence, BUPT. In February 2011, he spent one year as a Visiting Scholar with the Information Sciences Institute, University of Southern California, CA, USA. His research interests include multimedia information processing and natural language processing. He is a member of IEEE Signal Processing Society and Computer Society.



**Fangxiang Feng** received the B.S. and Ph.D. degrees from Beijing University of Posts and Telecommunications (BUPT) in 2010 and 2015, respectively. He is currently an Assistant Professor with the School of Artificial Intelligence, BUPT. His research interests include multimedia information retrieval, multimodal deep learning, and computer vision.

**Zhanyu Ma** (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from the KTH Royal Institute of Technology, Stockholm, Sweden, in 2011. From 2012 to 2013, he was a Post-Doctoral Research Fellow with the School of Electrical Engineering, KTH Royal Institute of Technology. He was an Associate Professor with Beijing University of Posts and Telecommunications (BUPT), Beijing, China, from 2014 to 2019. He has been an Adjunct Associate Professor with Aalborg University, Aalborg, Denmark, since 2015. He is currently a Full Professor with BUPT. His research interests include pattern recognition and machine learning fundamentals with a focus on applications in computer vision, multimedia signal processing, and data mining.

**Xiaojie Wang** received the Ph.D. degree from Beihang University in 1996. He is currently a Full Professor with Beijing University of Posts and Telecommunications. His research interests include natural language processing and multi-modal cognitive computing. He is an Executive Member of the Council of Chinese Association of Artificial Intelligence and the Director of the Natural Language Processing Committee. He is a member of the Council of Chinese Information Processing Society and the Chinese Processing Committee of China Computer Federation.