

RSVG: Exploring Data and Models for Visual Grounding on Remote Sensing Data

Yang Zhan, Zhitong Xiong, *Member, IEEE*, and Yuan Yuan, *Senior Member, IEEE*

arXiv:2210.12634v1 [cs.CV] 23 Oct 2022

Abstract—In this paper, we introduce the task of visual grounding for remote sensing data (RSVG). RSVG aims to localize the referred objects in remote sensing (RS) images with the guidance of natural language. To retrieve rich information from RS imagery using natural language, many research tasks, like RS image visual question answering, RS image captioning, and RS image-text retrieval have been investigated a lot. However, the object-level visual grounding on RS images is still under-explored. Thus, in this work, we propose to construct the dataset and explore deep learning models for the RSVG task. Specifically, our contributions can be summarized as follows. 1) We build the new large-scale benchmark dataset of RSVG, termed RSVGD, to fully advance the research of RSVG. This new dataset includes image/expression/box triplets for training and evaluating visual grounding models. 2) We benchmark extensive state-of-the-art (SOTA) natural image visual grounding methods on the constructed RSVGD dataset, and some insightful analyses are provided based on the results. 3) A novel transformer-based Multi-Level Cross-Modal feature learning (MLCM) module is proposed. Remotely-sensed images are usually with large scale variations and cluttered backgrounds. To deal with the scale-variation problem, the MLCM module takes advantage of multi-scale visual features and multi-granularity textual embeddings to learn more discriminative representations. To cope with the cluttered background problem, MLCM adaptively filters irrelevant noise and enhances salient features. In this way, our proposed model can incorporate more effective multi-level and multi-modal features to boost performance. Furthermore, this work also provides useful insights for developing better RSVG models. The dataset and code will be publicly available at <https://github.com/ZhanYang-nwpu/RSVG-pytorch>.

Index Terms—Visual grounding for remote sensing data (RSVG), transformer, multi-level cross-modal feature learning (MLCM).

I. INTRODUCTION

WITH the rapid development of remote sensing (RS) technology, the quantity and resolution of RS images have been rapidly improved [1–3]. To efficiently process and retrieve RS imagery, tasks of integrating natural language and RS imagery have become a hot research topic. Although there are many studies combining natural language processing (NLP) with RS, like RS image captioning [4–6], RS image-text retrieval [7–9], and RS image visual question answering [10–12], the task of visual grounding for RS data (RSVG) is still under-explored.

RSVG aims to localize the object referred by the query expression in RS images, as shown in Fig. 1. Given an RS

Yang Zhan and Yuan Yuan are with the School of Artificial Intelligence, Optics, and Electronics (OPEN), Northwestern Polytechnical University, Xi'an 710072, China (e-mail:y.yuan@nwpu.edu.cn).

Zhitong Xiong is with the Chair of Data Science in Earth Observation, Technical University of Munich (TUM), 80333 Munich, Germany.

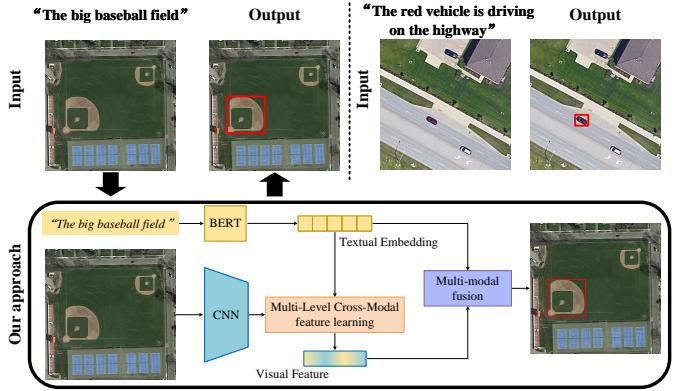


Fig. 1. Illustration of our task and approach. Top row: the input is an image-query pair and the output is a bounding box of the referred object. Each pair consists of an RS image and a query expression and the query can be a phrase or a sentence. Bottom row: our approach is an end-to-end transformer-based framework with four steps: 1) multi-modal encoding, 2) multi-level cross-modal feature learning, 3) multi-modal fusion, and 4) localizing.

image and a natural language expression, RSVG is asked to provide the referred object's bounding box. Query expressions include phrases and sentences. Multimodal machine learning (MML) [13, 14] enables computers to understand image-text pairs. Therefore, RSVG makes it possible for ordinary users, not limited to professionals or researchers, to retrieve objects in RS images, realizing human-computer interaction. It has a wide application prospect in scenarios such as military target detection, military intelligence generation, natural disaster monitoring, agriculture production, search and rescue activities, and urban planning [3, 4, 10].

Since RSVG has high potential in real-world applications, this paper explores the novel task and constructs a new large-scale dataset. We build a benchmark dataset, named RSVGD, using an automatic generation method with manual assistance. The construction procedure is shown in Fig. 2, including four steps: 1) box sampling, 2) attribute extraction, 3) expression generation, and 4) worker verification. The RSVGD dataset is sampled from the target detection dataset DIOR [15]. DIOR is large-scale on the number of object categories, object instances, and images, and has significant object size variations, image quality variations, inter-class similarity, and intra-class diversity. Thus, this new dataset provides researchers with a good data source to foster the research of RSVG. Specifically, RSVGD contains 38,320 RS image-expression pairs and 17,402 RS images, and the average length of expressions is 7.47. Nowadays, natural image visual grounding has been developed significantly. To fully advance the task of RSVG,

we benchmark extensive SOTA visual grounding methods on the RSVGD dataset. The existing methods can be divided into two-stage methods [16–31], one-stage methods [32–39], and transformer-based methods [40–45]. The experimental results show that transferring the visual grounding methods for natural image to RS image can obtain only acceptable results. Even if the above methods have achieved success in the natural domain, they still have some challenges that need to be tackled for the RSVG task.

Based on the characteristics of RS imagery and visual grounding, we propose a Multi-Level Cross-Modal feature learning (MLCM) module, which effectively improves the performance of RSVG. Firstly, unlike natural scene images, RS images are gathered from an overhead view by satellites, which results in large scale variations and cluttered backgrounds. Due to the characteristics, the model for solving RS tasks has to consider multi-scale inputs. The methods on natural images fail to fully take account of multi-scale features, which leads to suboptimal results on RS imagery. In addition, the background content of RS images contains numerous objects unrelated to the query, but natural images generally have salient objects. Due to the lack of filtering redundant features, the previous models are difficult to understand RS image-expression pairs. Therefore, we attempt to design a network that includes multi-scale fusion and adaptive filtering functions to refine visual features. Second, the previous frameworks that extract visual and textual features isolatedly do not conform to human perceptual habits, and such visual features lack the effective information needed for multi-modal reasoning. Inspired by the above discussion, we address the problem of how to learn fine-grained semantically salient image representations under multi-scale visual feature inputs. Based on cross-attention mechanism, MLCM module first utilizes multi-scale visual features and multi-granularity textual embeddings to guide the visual feature refining and achieve multi-level cross-modal feature learning. Considering that objects in an RS image are usually correlated, *e.g.*, stadiums usually co-occur with ground track fields, MLCM discovers the relations between object regions based on self-attention mechanism. Specifically, our MLCM includes multi-level cross-modal learning and self-attention learning. To sum up, our contributions can be summarized in the following aspects:

- 1) To foster the research of RSVG, we design an automatic RS image-query generation method with manual assistance, and the new large-scale dataset is constructed. Specifically, the new dataset contains 38,320 image-query pairs and 17,402 RS images.
- 2) We benchmark extensive SOTA natural image visual grounding methods on our RSVGD dataset. Based on experimental results, some analyses about the effects of different methods are given, which provide useful insights on the RSVG task.
- 3) To address the problems of scale-variation and cluttered background of RS images and capture the rich contextual dependencies between semantically salient regions, a novel transformer-based MLCM module is devised to learn more transcendent visual representations. MLCM

can incorporate effective information from multi-level and multi-modal features, which enables our method to achieve competitive performance.

This paper is organized as follows. We review the related work of natural image visual grounding in Section II. In Section III, the construction procedure of the new dataset is described and the characteristics are analyzed. In Section IV, we present our transformer-based RSVG method. Evaluation methods and extensive experiment results are shown in Section V. Finally, we conclude this work in Section VI.

II. RELATED WORK

In this section, we comprehensively review the related works about natural image visual grounding methods. To be more specific, two-stage, one-stage, and transformer-based methods are summarized in detail as follows.

A. Two-stage Visual Grounding Methods

With the development of visual grounding, various two-stage methods have been proposed. Yu et al. [17] introduced better visual context feature extraction methods and found that visual comparison with other objects in the image helps to improve the performance. In [18], a Spatial Context Recurrent ConvNet (SCRC) is presented, which contains two CNNs to extract local image features and global scene-level contextual features. Zhang et al. [19] proposed a variational Bayesian method for complex visual context modeling. Besides, a localization score function was also proposed, which is a variational lower bound consisting of multimodal modules of three specific cues and can be trained end-to-end using supervised or unsupervised losses. Hu et al. [20] attempted to parse the natural language into three modules: subject, relationship, and object, and align these components to candidate regions. The three modules are used to predict the scores of each candidate region. Attention mechanisms have been further introduced [21, 22] in each module to better model the interaction between language expressions and candidate regions. In addition, the attention mechanism [23] is utilized to reconstruct the input phrase and a parallel attention network (ParalAttn) [24], including image-level and proposal-level attention, is proposed. Yu et al. [25] found that existing two-stage methods pay more attention to multi-modal representation and region proposals ranking. Therefore, they proposed DDPN to improve region proposal generation, considering both the diversity and discrimination. Chen et al. [26] designed a reinforcement learning mechanism to guide the network to select more discriminative candidate boxes. In addition to the above methods, NMTTree [27] and RvG-Tree [28] utilized tree networks by parsing the language. To capture object relation information, several researchers [29–31] construct graphs. Yang et al. [29] and Wang et al. [30] proposed graph attention network to accomplish visual grounding. CMRIN [31] utilized Gated Graph Convolutional Network to fuse multimodal information.

B. One-stage Visual Grounding Methods

One-stage methods are more computation-efficient and can avoid error accumulation in multi-stage frameworks. Thus, many one-stage methods have been investigated. Some works use CNN and LSTM or Bi-LSTM to extract visual features and textual features [32–34]. Multimodal Compact Bilinear pooling (MCB) is first proposed in [32] to fuse the multimodal features. Chen et al. [33] designed a multimodal interactor to summarize the complex relationship between visual features and textual features. Besides, a new guided attention mechanism was designed to focus visual attention on the central area of the referred object. In [34], multi-scale features are extracted and multi-modal features are fed to the fully convolutional network to regress box coordinates. Significant improvement is observed as Yang et al. [35] fused textual embeddings with YOLOv3 detector results and augmented the visual features with spatial features. Liao et al. [36] defined the visual grounding problem as a correlation filtering process. They mapped textual features into three filtering kernels and performed correlation filtering on the image feature map. To address the limitations of FAOA [35] in complex queries for visual grounding, Yang et al. [37] proposed a recursive sub-query construction (ReSC) network. The latest one-stage methods [38, 39] focus on visual branching and use language expression to guide the visual feature extraction. A landmark feature convolution module [38] is designed to transmit visual features under the guidance of language and encode spatial relations between the object and its context. Liao et al. [39] proposed a language-guided visual feature learning mechanism to customize visual features in each stage and transfer them to the next stage.

C. Transformer-based Visual Grounding Methods

Recently, transformer-based methods have attracted more and more research attention due to the high efficiency and visual grounding performance. Du et al. [40] and Deng et al. [41] proposed the earliest end-to-end transformer-based visual grounding network, *i.e.*, VGTR and TransVG. VGTR [40] was a transformer structure that can learn visual features under the guidance of expression. TransVG [41] was a network stacked with multiple transformers, including BERT, visual transformer, and multimodal fusion transformer. Some studies [42, 43] propose a multi-task framework. Li and Sigal [42] utilized transformer encoder to refine visual and textual features and designed a query encoder and decoder for referring expression comprehension (REC) and segmentation (RES) at the same time. Sun et al. [43] proposed the transformer model for REC and referring expression generation (REG), which uses the same cross-attention module and fusion module to perform multi-modal interaction. Similar to the latest one-stage methods, the latest transformer-based methods [44, 45] also focus on the improvement of visual branches and adjusting visual features by combining multi-modal features. VLTVG [44] aims to adjust visual features with a visual-linguistic verification module and aggregate visual context with a language-guided context encoder. The core of these modules is multi-head attention. QRNet [45] contains a language query aware

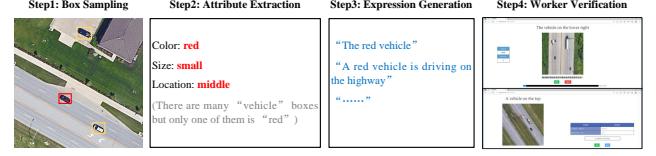


Fig. 2. Illustration of the dataset construction processes. Step 1: the red box is the sampling result; yellow boxes are ignored. Step 2: attribute extraction examples in the previous RS image. Step 3: expression generation examples in the previous RS image. Step 4: the dataset is manually validated using a data correction system.

dynamic attention mechanism and a language query aware multi-scale fusion to adjust visual features.

III. DATASET CONSTRUCTION

In this section, we will introduce the construction procedure of the new dataset in Section III-A. The statistical analysis of our RSVGD is shown in Section III-B.

A. RSVGD: a new dataset for RSVG

The dataset for RSVG requires lots of RS images with the annotation and description of different objects. Therefore, we utilize the existing target detection dataset DIOR [15] as the basic data to construct a new benchmark dataset. Over the years, various visual grounding datasets [17, 46–60] based on real-world and computer-generated images have been proposed to study visual grounding. The construction methods of each dataset are divided into manual annotation [46, 47, 49, 50, 55, 56, 60], game collection [17, 48, 51], and automatic generation [52, 54, 57–59]. We design an automatic image-query generation method with manual assistance to collect image/expression/box triplets, as shown in Fig. 2. A detailed description of the generation of different query expressions is given in what follows.

Step 1: Box sampling. DIOR dataset includes 23,463 RS images, 192,472 object instances, and 20 object categories. The image size is 800×800 pixels and the spatial resolution range is from 0.5m to 30m. First, the data containing annotation errors in the DIOR dataset are removed, *e.g.* axis-aligned bounding box coordinates $x_{min} \geq x_{max}$ or $y_{min} \geq y_{max}$. $(x_{min}, y_{min}, x_{max}, y_{max})$ is the coordinate of the ground-truth bounding box. Then, bounding boxes that are less than 0.02% or greater than 99% of the image size are also removed. Finally, we sample no more than 5 objects of the same category in each RS image to avoid unclear references of expression caused by many of the same category of objects in the image.

Step 2: Attribute extraction. By analyzing visual grounding datasets from the real world, such as RefItGame [48], RefCOCO [17], and RefCOCO+ [17], a set of attributes widely contained in referring expressions is summarized. we extract the attribute set and define it as a 7-tuple $A = \{a_1, a_2, a_3, a_4, a_5, a_6, a_7\}$. The symbol, type, and example of each attribute are shown in Table I. The object category can be obtained directly from the DIOR dataset. The HSV color recognition method is used to obtain the object's color. Object size is measured by the ratio of bounding box area to image size. The geometry attribute is set in advance for some objects

TABLE I
SPECIFIC INFORMATION FOR EACH ATTRIBUTE.

Attribute	Example
a_1 category	(e.g. “plane, ship”)
a_2 color	(e.g. “blue, red”)
a_3 size	(e.g. “tiny, big”)
a_4 geometry	(e.g. “square, round”)
a_5 absolute location	(e.g. “top of the image”)
a_6 relative location relation	(e.g. “the car is on the left of the tree”)
a_7 relative size relation	(e.g. “the car is smaller than the tree”)

with a fixed shape, such as rectangular basketball courts, circular storage tanks, etc. The geometric attribute of some objects that do not have a describable geometry is empty, such as airports, golf fields, etc. For other objects, we use relevant functions in the OpenCV library to extract object contours for common geometry recognition. Besides, the length and width of the bounding box are also combined to judge whether the object is slender or square. The absolute location refers to the location of the object in the image, which can be judged by the coordinates of bounding boxes. The above attributes $\{a_1, a_2, a_3, a_4, a_5\}$ belong to the object’s own attributes and the relationship attributes are $\{a_6, a_7\}$. The relative location and relative size relation allow expressions to be associated with another object. The relative location relation is obtained by comparing the coordinates of bounding boxes and center points. The relative size relation is determined by comparing two objects’ ratios of the bounding box area to the image size.

Step 3: Expression generation. To make the generated query expressions representative of the natural language used in the real world, we pre-set textual templates following the Cops-Ref [58] dataset. The filling of the textual template is the expression generation process. Textual templates include the phrase template and the sentence template. The phrase template uses the object’s own attributes $\{a_1, a_2, a_3, a_4, a_5\}$ in the following form:

$The/A (att_0) (obj_0) \Rightarrow The/A \langle a_2 \rangle \langle a_3 \rangle \langle a_4 \rangle a_1 \langle in/on the a_5 \rangle .$

The left of \Rightarrow is the phrase template and the right is the example filled in with specific attributes. The attribute that can be null is bounded with $\langle \rangle$ and attributes $\{a_2, a_3, a_4\}$ can be filled in any order. The sentence template uses the relationship attributes $\{a_6, a_7\}$ to relate two objects in the following form:

$The/A (att_0) (obj_0) is a_6/a_7 the (att_1) (obj_1) .$

We select textual templates and fill attributes to generate a query expression for each bounding box. The generation algorithm may be summarized as the following few steps:

- 1) We first check if the selected object category is unique in the RS image. If so, we fill the phrase template with the category name and randomly selected object attributes.
- 2) If the object category is not unique, we look for unique attributes of the object to distinguish it from other objects of the same category. If such an object attribute exists, we combine it with the category name to fill the phrase template.
- 3) If no such unique object attribute exists, we look for distinguishable relationship attributes. If such a relationship attribute exists, we combine it with the attributes of two objects to fill the sentence template.

- 4) If all the above fail, the object is discarded.

Step 4: Worker verification. Due to the complex backgrounds and numerous objects of RS images, attribute extraction may be wrong, especially for color and geometry. In addition, we use box regions instead of object pixel regions, which may cause errors in size attribute and relative size relation. Coupled with the unreliability of simple judgments of absolute location and relative location relation, the expression may be ambiguous. Therefore, RSVGD requires worker verification to help correct errors or ambiguous language expressions. The worker verification method consists of two main strategies, majority voting [61] and rapid judgments [62]. We only use rapid judgments to speed up the validation of datasets. To improve the efficiency, we develop a dataset manual correction system, and the system interface is shown in Fig. 2.

B. Data analysis

We construct a large-scale RSVGD, where each object instance in the RS image corresponds to a unique language expression. Our constructed RSVGD consists of 38,320 language expressions across 17,402 RS images and contains 20 object categories. The average length of expressions is 7.47 and the size of the vocabulary is 100. We now present a more detailed statistical analysis of the RSVGD dataset.

Fig. 3 (a) shows the proportion of the number of each object category. The vehicle and harbor are respectively the most and least in the dataset, and the remaining 18 categories account for a relatively uniform proportion, all between 2%-10%. Fig. 3 (b) provides the proportion of the number of attributes that appear in each expression. We find that most expressions used two attributes, followed by four attributes, with very few expressions containing five or six attributes. Fig. 3 (c) and (d) respectively show the proportion of object attributes and relationship attributes in query expressions of each object category. The percentage of each attribute is similar in different object categories, so the use of different attributes doesn’t depend on the object category. The bar chart shown in Fig. 3 (e) shows three kinds of information about query expressions from bottom to top: the proportion of expressions having object category information (cat) and the proportion of expressions that can distinguish objects by category information alone (cat+), and similarly for attributes and relationships. Specifically, 38.36% of objects can be distinguished by the object category alone (cat+), 56.62% of objects can be distinguished by the object’s own attribute(att+), and 15.74% by a relationship attribute (rel+). Fig. 3 (f) shows the distribution of the length of query expressions. The average length of expressions is 7.47 words, with a minimum of 3 words and a maximum of 22 words. The expressions need to be specific enough to describe individual objects in the RS image, such as the query “a dam”, but they also need to be general enough to describe high-level concepts in the RS image. Specifically, covering most of the areas in the RS image is often a general description of the image, while covering only a small part of the image is often more specific. The top row of Fig. 4 shows the distribution of the width, height, and area of the bounding box, with the area mainly within

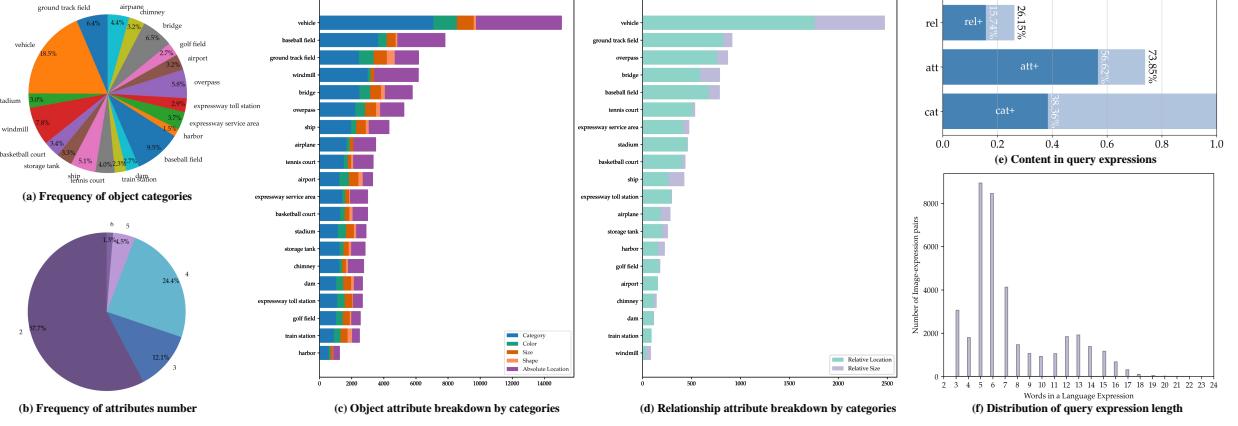


Fig. 3. Statistical analysis of the constructed RSVGD. (a) shows the frequency of each object category in RSVGD. (b) shows the frequency of the number of attributes contained in each query expression. (c) shows the frequency of the object attribute occurrence for each category. (d) shows the frequency of the relationship attribute occurrence for each category. (e) shows the distribution of expressions that have category (cat+) and expressions that can distinguish objects by category alone (cat+), and similarly for attributes and relationships. (f) shows the distribution of the length of query expressions.

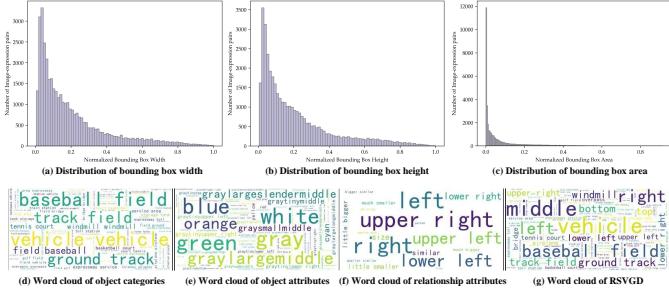


Fig. 4. Statistical analysis of the proposed RSVGD. Top row: the distribution of bounding box width (a), bounding box height (b), and bounding box area (c) in the RSVGD. Bottom row: word clouds of categories (d), attributes (e), relationships (f), and the RSVGD (g). The size of each word is proportional to its frequency in the dataset.

20% of the RS image. The bottom row of Fig. 4 shows the word clouds of object categories, object attributes, relationship attributes, and the RSVGD dataset. We can see that RSVGD covers a wide range of objects, with the vehicle, baseball field, and ground track field being the most common object names. The most common object attributes are color (*e.g.*, blue, white, green, and gray), size (*e.g.*, large), and absolute position (*e.g.*, middle), and the most common relationship attributes are relative location relation (*e.g.*, upper right and left).

IV. METHODS

This section introduces the transformer-based RSVG framework and our proposed MLCM module. We first overview the overall framework in Section IV-A. Then, we elaborate the architecture of the framework and our designs of MLCM module in Section IV-B. Finally, Section IV-C details the loss function of our framework for training.

A. Overview

RSVG is a task about localizing a target object described by a natural language expression in RS images. Our goal is to deal with the problems of scale-variation and the cluttered

background of RS images and capture the rich contextual dependencies between semantically salient regions. To this end, we propose an MLCM module to adaptively filter irrelevant noise and discover the relations between object regions, so that visual representations are focused on the valid regions referred by query expressions. To get full clues from multi-modal features at different semantic levels, MLCM takes advantage of multi-scale visual features and multi-granularity textual embeddings to refine visual features. We design CNN backbone to extract multi-scale visual features and use BERT to obtain word-level and sentence-level textual embeddings. Multi-scale visual features contain coarse-scale semantic information and fine-scale detailed information. Multi-granularity textual embeddings contain local and global information from different aspects. To mine the potential relationship between text semantics and visual perception, we use the transformer-based Multimodal Fusion Module that is analogous to TransVG [41] model. Since the visual grounding inference requires more detailed information, we take the refined visual features and word-level textual embeddings as the input of Multimodal Fusion Module. Two linear projection layers are applied to map visual features and textual embeddings into the same dimension. A learnable embedding (a learnable token) is pre-appended to visual embeddings and textual embeddings. It gathers the intra-modal and inter-modal information through Transformer's self-attention mechanism to facilitate visual grounding. Finally, the learnable token is sent to Localization Module for the regression of box coordinates. The overall framework is shown in Fig. 5 (a). We will introduce each module as follows in detail.

B. Multi-level Cross-modal Fusion

First, we denote the cross-modal RS image-query dataset as $\mathcal{O} = \{(i_m, s_m)\}_{m=1}^M$ that has M image-query pairs. RS images $\mathcal{I} = \{i_m\}_{m=1}^M$ and queries $\mathcal{S} = \{s_m\}_{m=1}^M$ have M instances in each modality. To simplify the notations, we denote I and S as single instances of image and text modality, respectively.

Multimodal Encoder. Given an RS image $I \in \mathbb{R}^{H_0 \times W_0 \times 3}$ and a query expression $S = \{w_n\}_{n=1}^N$ (N is the sentence

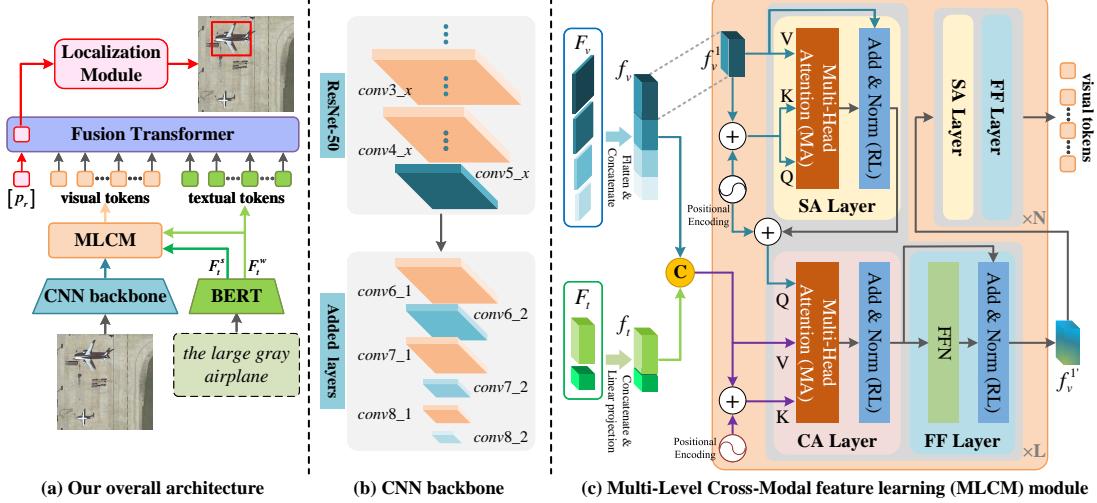


Fig. 5. (a) The architecture of the transformer-based RSVG framework. The framework consists of four components: a multimodal encoder, an MLCM module, a multimodal fusion module, and a localization module. (b) Illustration of our CNN backbone. It contains a truncated ResNet-50 and 6 additional convolution layers. (c) Illustration of our proposed MLCM module. It includes multi-level cross-modal learning and self-attention learning. MLCM first utilizes multi-scale visual features and multi-granularity textual embeddings to adaptively filter irrelevant noise and learn more discriminative visual representations. Then, MLCM captures the rich contextual dependencies between semantically salient regions based on self-attention mechanism.

length) as input of multimodal encoder, where $H_0 \times W_0 \times 3$ denotes the size of the RS image and w_n represents the n -th word. An overview of our CNN backbone is shown in Fig. 5 (b). We forward the RS image into the ResNet-50 that removed the average pool and FC layer to generate a 2D visual feature map $F_v^1 \in \mathbb{R}^{H \times W \times C}$. Multi-scale visual features are extracted by adding six additional convolution layers ($conv6_1$, $conv6_2$, $conv7_1$, $conv7_2$, $conv8_1$, and $conv8_2$) to the end of the truncated ResNet-50. $conv6_1$, $conv7_1$, and $conv8_1$ are all designed with 128 filters with a 1×1 filter size and a stride of 1, whereas $conv6_2$, $conv7_2$, and $conv8_2$ are all designed with 256 filters with a 3×3 filter size and a stride of 2. $conv6_2$, $conv7_2$, and $conv8_2$ output feature maps of size $9 \times 9 \times 256$, $4 \times 4 \times 256$, $1 \times 1 \times 256$ respectively, which are denoted by $[F_v^2, F_v^3, F_v^4]$. F_v^1 is also transformed into the same channel dimension $c = 256$, and F_v represents the multi-scale visual features:

$$F_v = [F_v^1, F_v^2, F_v^3, F_v^4]. \quad (1)$$

For the expression, we first embed each word w_n into a one-hot embedding vector. Then, we convert each one-hot vector into a language token and append $[CLS]$ token and $[SEP]$ token following the common approach in [41, 63–65]. To capture local semantic information and global sentence contextual information, we use the pre-trained BERT model [65] to extract word-level textual embeddings and sentence-level textual embeddings. BERT contains 12 Transformer encoders and the output channel dimension of BERT is b . Specifically, the average of the last four layers' hidden states is taken as the word-level textual embedding $F_t^w \in \mathbb{R}^{N_t \times b}$ of this query. Here, N_t represents the length of language tokens and the tokens ensure a fixed length N_t by padding or cutting. The embeddings output by the BERT are pooled as the sentence-level textual embedding $F_t^s \in \mathbb{R}^{1 \times b}$. F_t represents the multi-

granularity textual embeddings:

$$F_t = [F_t^w, F_t^s]. \quad (2)$$

MLCM Module. Unlike the traditional Transformer encoder-decoder structure, our MLCM first has a separate decoder, which is connected to a separate encoder. As shown in Fig. 5 (c), our MLCM consists of two parts: a multi-level cross-modal layer and a self-attention layer. MLCM requires two inputs, x and y . It can refine x by selecting and aggregating valid information from y based on the global relationship between x and y at all positions. In order to refine x under the guidance of multi-level and multi-modal features, the input y should contain information for all levels and modalities. We flatten multi-scale visual features F_v into $[f_v^1, f_v^2, f_v^3, f_v^4]$, where $f_v^i \in \mathbb{R}^{N_i \times c}$, $N_i = H_i \times W_i$, and $i \in \{1, 2, 3, 4\}$. Although N_i is different, the channel dimension is the same. Instead of sampling to the same size for feature fusion, we directly concatenate f_v^1 , f_v^2 , f_v^3 , and f_v^4 on the channel dimension to obtain $f_v \in \mathbb{R}^{N_{1234} \times c}$, where $N_{1234} = \sum_{i=1}^4 N_i$. The object scale of RS imagery varies greatly. The method maintains the resolution of original features and can preserve useful information about objects of different scales. Word-level features play an important role in visual grounding, especially the word-level feature directly related to the target object in the text. In addition, contextual semantic information at the sentence level can also provide useful clues for visual grounding. Similarly, we concatenate F_t^w and F_t^s , and use a linear layer to get $f_t \in \mathbb{R}^{(N_t+1) \times c}$. After intra-modal concatenation, we concatenate inter-modal features f_v and f_t to obtain $f_{vt} \in \mathbb{R}^{(N_{1234}+N_t+1) \times c}$. As the input y , f_{vt} provides rich information on all levels and modalities. We express the L-layer decoder's process of l -th layer as

$$f_v^{1l} = DE^l \left(f_v^{1(l-1)}, f^{vt} \right), \quad (3)$$

where $l \in [1, \dots, L]$, \mathbf{f}_v^{1l} is the output of l -th layer. $\mathbf{f}_v^1 \in \mathbb{R}^{N_1 \times c}$ is the initial visual feature as input x of the 1-th layer, where $N_1 = H \times W$. The decoder includes self-attention (SA) layer which refines itself, cross-attention (CA) layer which aggregates complementary information in \mathbf{f}_{vt} , and feed forward (FF) layer. The specific formula of $DE^l(\cdot)$ is as follows:

$$\mathbf{F}_{sa} = SA(\mathbf{f}_v^{1(l-1)}), \quad (4)$$

$$\mathbf{F}_{ca} = CA(\mathbf{F}_{sa}, \mathbf{f}_{vt}), \quad (5)$$

$$\mathbf{f}_v^{1l} = FF(\mathbf{F}_{ca}). \quad (6)$$

The SA layer and CA layer both contain a multi-head attention (MA) module and a residual connection and layer normalization (RL) block. In the MA module, attention is calculated h times. The single attention takes Query \mathbf{Q} , Key \mathbf{K} , and Value \mathbf{V} as input and is calculated by:

$$Att(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = softmax\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_K}}\right)\mathbf{V}, \quad (7)$$

where $\mathbf{Q} \in \mathbb{R}^{N_Q \times d}$, $\mathbf{K} \in \mathbb{R}^{N_M \times d}$, $\mathbf{V} \in \mathbb{R}^{N_M \times d}$. N_Q is the length of \mathbf{Q} and N_M is the length of \mathbf{K} and \mathbf{V} . d_K is the dimension of \mathbf{K} . The output of $Att(\cdot)$ is the same size $\mathbb{R}^{N_Q \times d}$ as \mathbf{Q} . For each attention, \mathbf{Q} and \mathbf{K} are appended with their corresponding positional encodings.

$$\begin{aligned} SA(\mathbf{x}) &= RL(Att(\tilde{\mathbf{x}}, \tilde{\mathbf{x}}, \mathbf{x})) \\ &= norm(\mathbf{x} + Att(\tilde{\mathbf{x}}, \tilde{\mathbf{x}}, \mathbf{x})) = \mathbf{x}_{sa}, \end{aligned} \quad (8)$$

$$\begin{aligned} CA(\mathbf{x}_{sa}, \mathbf{y}) &= RL(Att(\tilde{\mathbf{x}}_{sa}, \tilde{\mathbf{y}}, \mathbf{y})) \\ &= norm(\mathbf{x}_{sa} + Att(\tilde{\mathbf{x}}_{sa}, \tilde{\mathbf{y}}, \mathbf{y})) = \mathbf{x}_{ca}, \end{aligned} \quad (9)$$

where $\tilde{\mathbf{x}}$, \mathbf{x}_{sa} , and $\tilde{\mathbf{y}}$ are \mathbf{x} , \mathbf{x}_{sa} , and \mathbf{y} with positional encoding, respectively.

$$\tilde{\mathbf{x}} = \mathbf{x} + PosEncoding(\mathbf{x}), \quad (10)$$

$$\tilde{\mathbf{x}}_{sa} = \mathbf{x}_{sa} + PosEncoding(\mathbf{x}), \quad (11)$$

$$\tilde{\mathbf{y}} = \mathbf{y} + PosEncoding(\mathbf{y}), \quad (12)$$

where $PosEncoding(\cdot)$ denotes the function to get positional encoding. The positional encoding of \mathbf{f}_{vt} is obtained by concatenating positional encoding of multi-level multi-modal features in sequence.

The FF layer contains a feed forward network (FFN), which consists of two linear layers and a ReLU activation function in the middle, and an RL block. FF is defined as below:

$$\begin{aligned} FF(\mathbf{x}_{ca}) &= RL(FFN(\mathbf{x}_{ca})) \\ &= norm(\mathbf{x}_{ca} + FFN(\mathbf{x}_{ca})). \end{aligned} \quad (13)$$

Considering that objects in an RS image are usually correlated, MLCM discovers the relations between object regions based on self-attention mechanism. Specifically, we build an N-layer encoder. The encoder consists of 6 transformer encoder layers, including 8 MA layers. The output channel

sizes of the two fully connected layers in FFN are 2048 and 256 respectively. Through self-attention layers, the output \mathbf{f}_v^{1l} of the L-layer decoder can see other information in the same feature map. Meanwhile, self-attention layers generate visual embedding for the multimodal fusion module.

Multimodal Fusion Module. The visual tokens generated by MLCM and the word-level textual embeddings $\mathbf{f}_t^w \in \mathbb{R}^{N_t \times c}$ serve as the input of the fusion module. After the projection, the visual tokens and textual tokens are denoted as $\mathbf{p}_v \in \mathbb{R}^{N_v \times D}$ and $\mathbf{p}_t \in \mathbb{R}^{N_t \times D}$, respectively. Then a learnable token (a learnable embedding) is attached, concatenated together with \mathbf{p}_v and \mathbf{p}_t . The joint sequence is denoted as:

$$\mathbf{P} = [\underbrace{\mathbf{p}_v^1, \mathbf{p}_v^2, \dots, \mathbf{p}_v^{N_v}}_{visual\ tokens\ \mathbf{p}_v}, \underbrace{\mathbf{p}_t^1, \mathbf{p}_t^2, \dots, \mathbf{p}_t^{N_t}}_{textual\ tokens\ \mathbf{p}_t}, \mathbf{p}_l], \quad (14)$$

where $\mathbf{p}_l \in \mathbb{R}^{1 \times D}$ is the learnable token, which is randomly initialized before the training.

Next, a fusion transformer is used to embed $\mathbf{P} \in \mathbb{R}^{(N_v+N_t+1) \times D}$, which specifically includes 6 transformer encoder layers.

Localization Module. We use the representation of the learnable token from the multimodal fusion module as the input. The localization module is composed of a multi-layer perceptron (MLP), which is specifically composed of a 256-dim hidden layer, a ReLU activation function, and a linear layer, and outputs a 4-dimensional bounding box coordinate.

C. Loss

Following the previous method [41], we apply the commonly used smooth L1 loss [66] $\mathcal{L}_{smooth-L1}(\cdot)$ and the generalized IoU (GIoU) loss [67] $\mathcal{L}_{GIoU}(\cdot)$ on the 4-dim bounding box coordinate. Adding GIoU loss is important for RSVG. The target size in RS images varies greatly, so the smooth L1 loss will be a large value when predicting a large box. Smooth L1 loss will be a small number when predicting a small box, even if the predicted box has a large error. Therefore, we normalize the coordinates of the ground-truth boxes according to the image size and use GIoU loss that is not affected by scale. Then, the whole loss function for training our proposed network can be written:

$$\mathcal{L} = \mathcal{L}_{smooth-L1}(\mathbf{b}, \hat{\mathbf{b}}) + \lambda \cdot \mathcal{L}_{GIoU}(\mathbf{b}, \hat{\mathbf{b}}), \quad (15)$$

where $\mathbf{b} = (x_{min}, y_{min}, x_{max}, y_{max})$ denotes the coordinates of the ground-truth bounding box and $\hat{\mathbf{b}} = (\hat{x}_{min}, \hat{y}_{min}, \hat{x}_{max}, \hat{y}_{max})$ denotes the coordinates of the prediction. λ is the hyper-parameter to balance two losses.

V. EXPERIMENTS

In this section, we present extensive experiments to validate the merits of our proposed MLCM. In Section V-A and Section V-B, we introduce the evaluation metrics for RSVG and experimental setup details. We provide the main results of our method and compare the results with other state-of-the-art approaches for visual grounding in Section V-C. In Section V-D, we perform sufficient ablation experiments to verify the effectiveness of our MLCM. Finally, we show some qualitative results to fully analyze our model in Section V-E.

A. Evaluation metrics for RSVG

Given an RS image-query pair, the predicted bounding box is considered right if the intersection-over-union (IoU) with the ground-truth bounding box is above a threshold. In previous visual grounding works, a threshold of 0.5 is used as an accuracy metric. We report the metrics with IoU thresholds at 0.5, 0.6, 0.7, 0.8, and 0.9, termed as Pr@0.5, Pr@0.6, Pr@0.7, Pr@0.8, and Pr@0.9, respectively. In addition, we follow the evaluation metrics of [59], including mean IoU and cumulative IoU (cumIoU), with the following equations:

$$\text{meanIoU} = \frac{1}{M} \sum_t I_t / U_t, \quad (16)$$

and

$$\text{cumIoU} = (\sum_t I_t) / (\sum_t U_t). \quad (17)$$

Here t is the index of image-query pairs and M represents the size of the dataset. I_t and U_t are the intersection and union area between predicted and ground-truth bounding boxes.

B. Implementation Details

We split the dataset by randomly assigning 40%, 10%, and 50% of the expressions and their corresponding images to the training, validation, and test set. We resize the image size to a fixed size of 640×640 for training. We set the maximum length of language tokens $N_t = 40$ and the dimension $D = 256$. The ResNet-50 and MLCM use the pre-training weights of the DETR model [68]. We use the pre-trained weights of BERT [65] to initialize $BERT_{base}$ for textual feature extraction. The hidden size b of BERT is 768. We follow the TransVG [41] to process the input images and expressions. During training, we adopt AdamW [69] with weight decay 10^{-4} as our optimizer. We train our network with a batch size of 8 for 150 epochs on one GTX 1080Ti 11GiB GPU. The dropout ratio is set to 0.1 for FFN in Transformer. We set the initial learning rate of our network to 10^{-5} for pre-trained parameters and 10^{-4} for other parameters. We use Xavier [70] to randomly initialize the parameters without pre-training in our network. For the loss function in Eq. 15, we set $\lambda = 1$.

C. Remote Sensing Image Visual Grounding Results

In order to assess the merits of our proposed method, we report our performance and compare it with the SOTA methods for natural image on our constructed RSVGD. As the results are shown in Table II, we observe that our method outperforms other works. The two-stage method relies on a pre-trained object detector to generate object proposals and extract features, such as Mask R-CNN [71]. Since the existing object detectors are pre-trained on natural images, the visual features of these detectors may not be compatible with the RSVG task. The quality of pre-generated proposals can be a performance bottleneck for the two-stage methods. The top parts of the table show results of current one-stage methods. The one-stage methods require pre-set anchors or manually designed complex multi-modal fusion mechanisms to yield bounding boxes. In fact, these works may lead to insufficient use of multi-modal information or over-fitting of

datasets for specific scenes. Our approach uses the transformer structure for feature encoding and feature fusion, which is more flexible and can realize more full interaction between visual information and textual information. Except for the Pr@0.9, our method is much higher than one-stage methods in other metrics. FAOA [35] fuses textual embeddings into YOLOv3 and fuses visual, textual, and spatial features at three different spatial resolutions. The model achieves the best accuracy at the threshold of 0.9 due to feature fusion at different resolutions, but the performance is still deficient at smaller thresholds. In the middle parts of Table II, we also compare our method to other transformer-based methods, *i.e.*, TransVG [41] and VLTVG [44]. In contrast to our method, VLTVG designs a language-guided visual feature aggregation method and a multi-stage cross-modal decoder. The distinctiveness of visual features can be improved because visual features are concentrated in areas related to text descriptions while irrelevant areas are ignored in the training process. However, the performance is still insufficient because it ignores multi-level modality information. Our method follows the visual-linguistic transformer structure in TransVG to fuse multi-modal features. Besides, our MLCM uses multi-scale visual features and multi-granularity textual embeddings to learn more discriminative visual representations, which can aggregate effective information from multi-level multi-modal features and filter the redundant features of RS images.

D. Ablation Study

In this section, we conduct detailed experiments to systematically analyze the proposed MLCM. As shown in Table III, we study the effectiveness of the multi-level cross-modal feature learning mechanism. The first row shows the RSVG results without multi-level cross-modal feature learning, which achieves 72.41% Pr@0.5 on the testset of RSVGD. The second row shows the results of unimodal feature learning containing only multi-scale visual features, and the performance is dropped by 6.63%. Then, we add sentence-level and word-level textual embeddings, respectively. The results, as shown in the third and fourth row, drop by 4.95% and 1.51%, respectively. In the fifth row, we adopt unimodal feature learning containing only multi-granularity textual embeddings, and the result is improved by 0.37%. The last row indicates results of the complete multi-level cross-modal feature learning, showing a 4.37% performance improvement over the absence of MLCM. The performance is greatly improved, which demonstrates that the representations for RSVG can be modeled more effectively with the design of MLCM module.

To deeply analyze the results containing only multi-scale visual features, we visually compare the attention map of MLCM and ablation model(b), as shown in Fig. 6. The darker background color in the attention map indicates the higher attention to this region. According to the attention maps, there are also many regions of dark color in the background or non-target areas when performing unimodal feature learning containing only multi-scale visual features. Therefore, due to the cluttered background of RS images, a large amount of noise is introduced when containing only

TABLE II
COMPARISON WITH THE STATE-OF-THE-ART METHODS FOR RSVG ON THE TEST SET OF RSGVD. THE BEST PERFORMANCE IS WITH BOLD AND THE SECOND PERFORMANCE IS WITH UNDERLINE.

Methods	Venue	Visual Encoder	Language Encoder	Pr@0.5	Pr@0.6	Pr@0.7	Pr@0.8	Pr@0.9	meanIoU	cumIoU
<i>One-stage:</i>										
ZSGNet [34]	ICCV'19	VGG	BiLSTM	48.12	43.79	36.82	25.04	6.62	40.23	46.11
ZSGNet [34]	ICCV'19	ResNet-50	BiLSTM	51.67	48.13	42.30	32.41	10.15	44.12	51.65
FAOA-no Spatial [35]	ICCV'19	DarkNet-53	BERT	63.63	61.20	56.92	50.15	38.83	57.53	62.66
FAOA [35]	ICCV'19	DarkNet-53	BERT	67.21	64.18	59.23	50.87	34.44	59.76	63.14
FAOA [35]	ICCV'19	DarkNet-53	LSTM	70.86	67.37	62.04	53.19	<u>36.44</u>	62.86	67.28
ReSC [37]	ECCV'20	DarkNet-53	BERT	72.71	68.92	63.01	53.70	<u>33.37</u>	64.24	68.10
LBYL-Net [38]	CVPR'21	DarkNet-53	LSTM	73.29	69.92	63.97	48.07	16.60	65.86	75.45
LBYL-Net [38]	CVPR'21	DarkNet-53	BERT	73.78	69.22	65.56	47.89	15.69	65.92	76.37
<i>Transformer-based:</i>										
TransVG [41]	ICCV'21	ResNet-50	BERT	72.41	67.38	60.05	49.10	27.84	63.56	76.27
VLTVG [44]	CVPR'22	ResNet-50	BERT	69.41	65.16	58.44	46.56	24.37	59.96	71.97
VLTVG [44]	CVPR'22	ResNet-101	BERT	<u>75.79</u>	<u>72.22</u>	<u>66.33</u>	<u>55.17</u>	33.11	<u>66.32</u>	<u>77.85</u>
Ours	-	ResNet-50	BERT	76.78	72.68	66.74	56.42	35.07	68.04	78.41

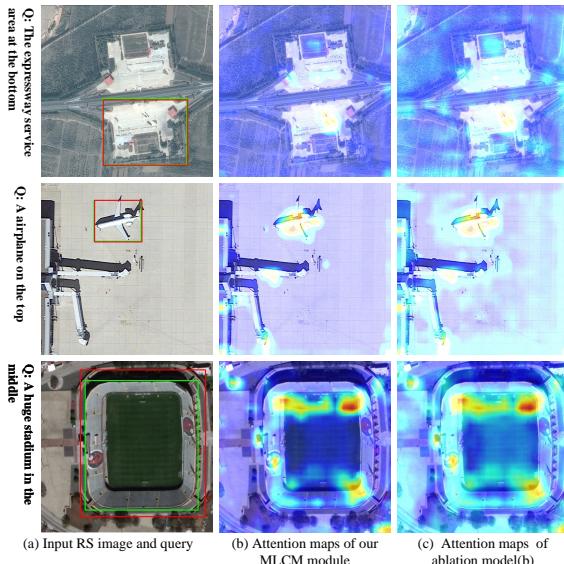


Fig. 6. Visualization of the final grounding results (green/red boxes are ground-truths/predicted regions), the attention maps of our MLCM, and the attention maps of ablation model(b) for various input expressions and RS images on the RSGVD test set.

multi-scale visual features. So the performance is greatly dropped. However, when multi-granularity textual embeddings are added, the noise in the attention maps is greatly filtered, so the performance is significantly improved. When only single-granularity textual embeddings (*i.e.*, word-level or sentence-level) are added, some noise can be filtered, but the performance is still slightly lower than without MLCM. When unimodal feature learning containing only multi-granularity textual embeddings is performed, the performance is slightly improved than without MLCM. The above analyses prove that RS visual features are complex. But the MLCM module has multi-level multi-modal feature input and cross-modal learning capability to adaptively filter irrelevant noise, enhance salient features, and learn more discriminative visual representations.

E. Qualitative Results

In Figs. 6 and 7, we show some qualitative results on the test set. We visualize the final grounding results and the attention

TABLE III
THE ABLATION STUDIES OF THE MLCM MODULE IN OUR NETWORK.

Models	Visual multi-scale	Textual word-level	Textual sentence-level	Pr@0.5 (%)
(a)	✗	✗	✗	72.41
(b)	✓	✗	✗	$65.78 \downarrow 6.63$
(c)	✓	✗	✓	$67.46 \downarrow 4.95$
(d)	✓	✓	✗	$70.90 \downarrow 1.51$
(e)	✗	✓	✓	$72.78 \uparrow 0.37$
ours	✓	✓	✓	76.78 $\uparrow 4.37$

maps for various inputs. It is observed that our method can accurately localize objects described in query expressions with specific attributes. In addition, MLCM can focus on the visual features of the region where the target object is localized under the guidance of multi-level and multi-modal features. For example, the first three image-query pairs in Fig. 7 refer to a bridge with a vehicle, an overpass, and a vehicle driving on the overpass. MLCM accurately enhances the visual features of the corresponding areas of the bridge, overpass, and vehicle. MLCM can effectively generate interpretable attention of natural language corresponding to the shape and location of the entire target object.

Due to the sufficient interaction of fine-granularity textual embeddings and multi-scale visual features, our method can accurately localize small-scale objects. The MLCM proposed in this paper has a better visual representation learning effect for small-scale targets. As shown in Fig. 8, the dark backgrounds in the attention map are the regions where various small-scale objects are located. MLCM can combine multi-scale visual features and multi-granularity textual embeddings to precisely enhance the visual features of small-scale objects and improve the grounding accuracy.

According to the visualization, many hot regions that are not the target regions are observed in attention maps. The regions of wrong attention are mainly divided into two types. The first type is the region where objects belonging to the same category as the target object are located, as shown in the first two data of Fig. 9. The other is the region where objects that have a relationship with the target object are located. The

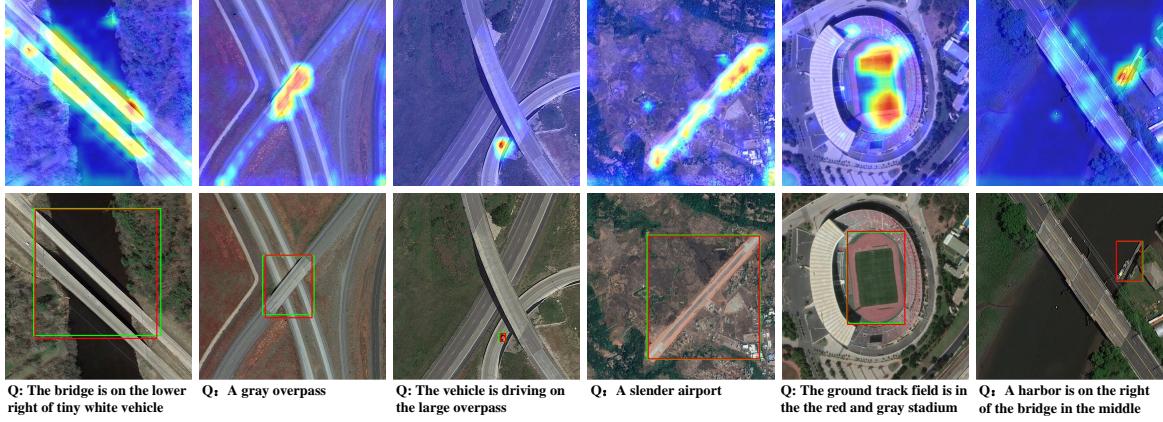


Fig. 7. Visualization of the final grounding results and the attention maps of our proposed MLCM.

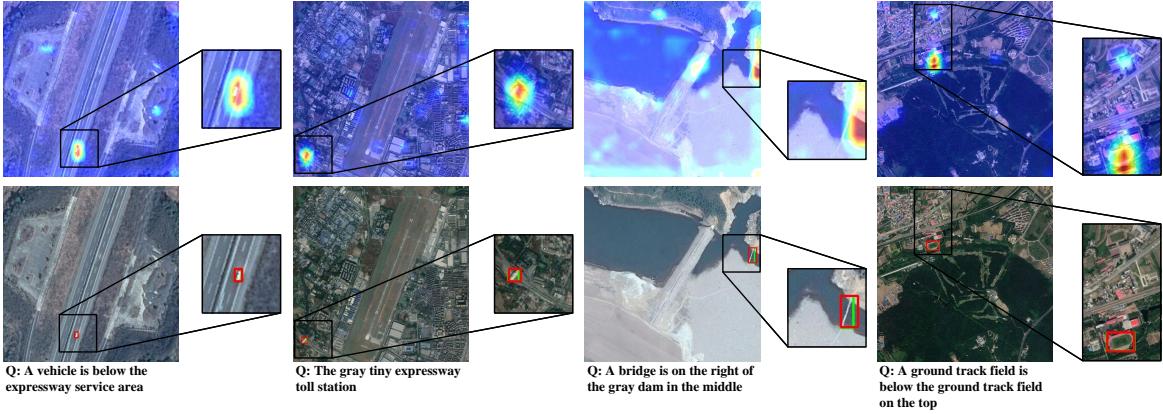


Fig. 8. Some cases of small-scale target grounding on the RSVGD test set.

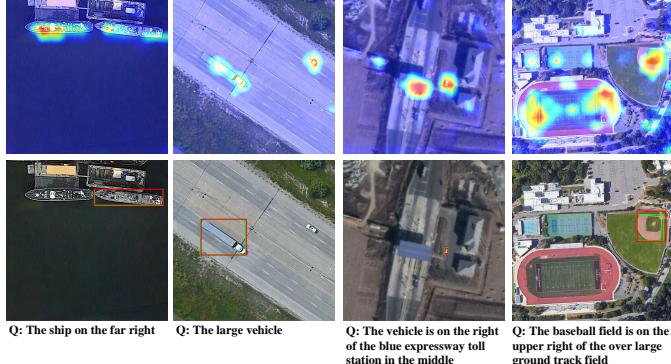


Fig. 9. Some cases with wrong attention regions on the RSVGD test set.

second type exists in the last two data, which are the vehicle that has a relationship with an expressway toll station and the baseball field that has a relationship with a ground track field. However, the impact of wrong attention will be avoided in the transformer-based multimodal fusion module. The module makes full use of word-level textual embeddings with local semantic information to further align visual modality and textual modality for accurate visual grounding.

Our model is in line with human perception habits, which can learn more discriminative visual representations of RS images and effectively fuse and align visual features and textual

embeddings. It can model and reason under the guidance of query expressions with complex relationships. However, our approach also has some failure results. As shown in Fig. 10, there are three main types of grounding failure. The first type is shown in the first column. Due to the cluttered backgrounds of the RS image, there are areas or different objects with similar visual features to the target object and the scope of the object is difficult to define precisely. The second is shown in the second column. Due to the incompleteness, complexity, and ambiguity of query expressions, the objects between the same category cannot be distinguished or the target objects cannot be clearly referred and localized. The above problems in the dataset make it difficult for the model to accurately ground target objects, and errors will inevitably occur. The last column of Fig. 10 is caused by the lack of performance of the model. When the object in the RS image is salient and the attributes described by the expression are clear and unambiguous, the model cannot complete the RSVG correctly. The result indicates that our proposed method still has some shortcomings and needs further research and improvement.

VI. CONCLUSION

In this paper, we introduce a new task to ground natural language expressions on RS imagery. To the best of our knowledge, we build the new large-scale dataset for RSVG. RSVG is obtained from the DIOR dataset by an automatic generation

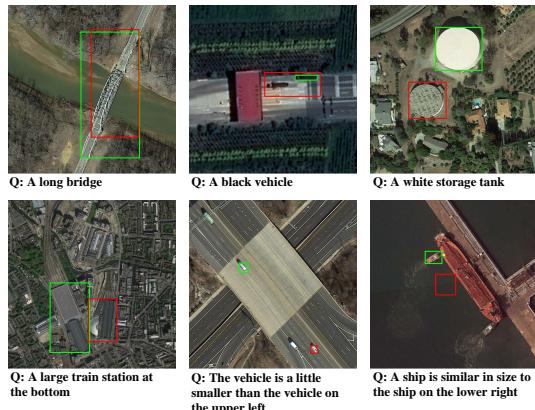


Fig. 10. Failure cases of our method on the RSVGD test set.

algorithm with manual verification, which greatly reduces the collection cost of the dataset and ensures the correctness of the dataset. RSVGD is large-scale on the number of image-query pairs and has high inter-class similarity and intra-class diversity. In addition, we benchmark extensive SOTA natural image methods on our constructed RSVGD and analyze the results. We obtain only acceptable results using natural image methods, suggesting the potential for future research. Finally, a novel transformer-based MLCM module is devised to solve problems of the cluttered background and scale-variation of RS images. The main innovation is that MLCM adapts to multi-scale inputs and incorporates effective information from multi-level and multi-modal features to learn the attention of visual representations relevant to the query. Compared with existing natural image visual grounding methods, our approach achieves better performance and shows its superiority. In future work, more works need to be done on RSVG considering the characteristics of RS images.

REFERENCES

- [1] Z. Xiong, F. Zhang, Y. Wang, Y. Shi, and X. X. Zhu, “EarthNets: Empowering AI in Earth Observation,” *arXiv:2210.04936*, 2022.
- [2] S.-B. Chen, Q.-S. Wei, W.-Z. Wang, J. Tang, B. Luo, and Z.-Y. Wang, “Remote sensing scene classification via multi-branch local attention network,” *IEEE Transactions on Image Processing*, vol. 31, pp. 99–109, 2021.
- [3] H. Ning, B. Zhao, and Y. Yuan, “Semantics-consistent representation learning for remote sensing image–voice retrieval,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–14, 2022.
- [4] X. Lu, B. Wang, X. Zheng, and X. Li, “Exploring models and data for remote sensing image caption generation,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 4, pp. 2183–2195, 2018.
- [5] Y. Li, X. Zhang, J. Gu, C. Li, X. Wang, X. Tang, and L. Jiao, “Recurrent attention and semantic gate for remote sensing image captioning,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–16, 2021.
- [6] Z. Zhang, W. Zhang, M. Yan, X. Gao, K. Fu, and X. Sun, “Global visual feature and linguistic state guided attention for remote sensing image captioning,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–16, 2021.
- [7] G. Mikriukov, M. Ravanbakhsh, and B. Demir, “Deep unsupervised contrastive hashing for large-scale cross-modal text-image retrieval in remote sensing,” *arXiv preprint arXiv:2201.08125*, 2022.
- [8] Z. Yuan, W. Zhang, X. Rong, X. Li, J. Chen, H. Wang, K. Fu, and X. Sun, “A lightweight multi-scale crossmodal text-image retrieval method in remote sensing,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–19, 2021.
- [9] Z. Yuan, W. Zhang, K. Fu, X. Li, C. Deng, H. Wang, and X. Sun, “Exploring a fine-grained multiscale method for cross-modal remote sensing image retrieval,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–19, 2021.
- [10] S. Lobry, D. Marcos, J. Murray, and D. Tuia, “Rsvqa: Visual question answering for remote sensing data,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 12, pp. 8555–8566, 2020.
- [11] Z. Yuan, L. Mou, Q. Wang, and X. X. Zhu, “From easy to hard: Learning language-guided curriculum for visual question answering on remote sensing data,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–11, 2022.
- [12] Z. Yuan, L. Mou, Z. Xiong, and X. X. Zhu, “Change detection meets visual question answering,” *IEEE Transactions on Geoscience and Remote Sensing*, 2022.
- [13] Z. Xiong, Y. Yuan, N. Guo, and Q. Wang, “Variational context-deformable ConvNets for indoor scene parsing,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 3992–4002.
- [14] Z. Xiong, Y. Yuan, and Q. Wang, “Ask: Adaptively selecting key local features for rgb-d scene recognition,” *IEEE Transactions on Image Processing*, vol. 30, pp. 2722–2733, 2021.
- [15] K. Li, G. Wan, G. Cheng, L. Meng, and J. Han, “Object detection in optical remote sensing images: A survey and a new benchmark,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 159, pp. 296–307, 2020.
- [16] C. L. Zitnick and P. Dollár, “Edge boxes: Locating object proposals from edges,” in *European conference on computer vision (ECCV)*, 2014, pp. 391–405.
- [17] L. Yu, P. Poirson, S. Yang, A. C. Berg, and T. L. Berg, “Modeling context in referring expressions,” in *European Conference on Computer Vision (ECCV)*, 2016, pp. 69–85.
- [18] R. Hu, H. Xu, M. Rohrbach, J. Feng, K. Saenko, and T. Darrell, “Natural language object retrieval,” in *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2016, pp. 4555–4564.
- [19] H. Zhang, Y. Niu, and S.-F. Chang, “Grounding referring expressions in images by variational context,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 4158–4166.
- [20] R. Hu, M. Rohrbach, J. Andreas, T. Darrell, and K. Saenko, “Modeling relationships in referential expressions with compositional modular networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2017, pp. 1115–1124.
- [21] L. Yu, Z. Lin, X. Shen, J. Yang, X. Lu, M. Bansal, and T. L. Berg, “Mattnet: Modular attention network for referring expression comprehension,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 1307–1315.
- [22] X. Liu, Z. Wang, J. Shao, X. Wang, and H. Li, “Improving referring expression grounding with cross-modal attention-guided erasing,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, 2019, pp. 1950–1959.
- [23] A. Rohrbach, M. Rohrbach, R. Hu, T. Darrell, and B. Schiele, “Grounding of textual phrases in images by reconstruction,” in *European Conference on Computer Vision (ECCV)*, 2016, pp. 817–834.
- [24] B. Zhuang, Q. Wu, C. Shen, I. Reid, and A. Van Den Hengel, “Parallel attention: A unified framework for visual object discovery through dialogs and queries,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 4252–4261.
- [25] Z. Yu, J. Yu, C. Xiang, Z. Zhao, Q. Tian, and D. Tao, “Rethinking diversified and discriminative proposal generation for visual grounding,” in *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI)*, 2018, pp. 1114–1120.
- [26] K. Chen, R. Kovvuri, and R. Nevatia, “Query-guided regression network with context policy for phrase grounding,” in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 824–832.
- [27] D. Liu, H. Zhang, F. Wu, and Z.-J. Zha, “Learning to assemble neural module tree networks for visual grounding,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 4673–4682.
- [28] R. Hong, D. Liu, X. Mo, X. He, and H. Zhang, “Learning to compose and reason with language tree structures for visual grounding,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 2, pp. 684–696, 2022.
- [29] S. Yang, G. Li, and Y. Yu, “Dynamic graph attention for referring expression comprehension,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 4644–4653.
- [30] 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 *Proceedings of the IEEE/CVF*

- Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 1960–1968.
- [31] S. Yang, G. Li, and Y. Yu, “Relationship-embedded representation learning for grounding referring expressions,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 8, pp. 2765–2779, 2020.
- [32] A. Fukui, D. H. Park, D. Yang, A. Rohrbach, T. Darrell, and M. Rohrbach, “Multimodal compact bilinear pooling for visual question answering and visual grounding,” in *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2016, pp. 457–468.
- [33] X. Chen, L. Ma, J. Chen, Z. Jie, W. Liu, and J. Luo, “Real-time referring expression comprehension by single-stage grounding network,” *arXiv preprint arXiv:1812.03426*, 2018.
- [34] A. Sadhu, K. Chen, and R. Nevatia, “Zero-shot grounding of objects from natural language queries,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 4694–4703.
- [35] Z. Yang, B. Gong, L. Wang, W. Huang, D. Yu, and J. Luo, “A fast and accurate one-stage approach to visual grounding,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 4683–4693.
- [36] Y. Liao, S. Liu, G. Li, F. Wang, Y. Chen, C. Qian, and B. Li, “A real-time cross-modality correlation filtering method for referring expression comprehension,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 10880–10889.
- [37] Z. Yang, T. Chen, L. Wang, and J. Luo, “Improving one-stage visual grounding by recursive sub-query construction,” in *European Conference on Computer Vision (ECCV)*, 2020, pp. 387–404.
- [38] B. Huang, D. Lian, W. Luo, and S. Gao, “Look before you leap: Learning landmark features for one-stage visual grounding,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 16888–16897.
- [39] Y. Liao, A. Zhang, Z. Chen, T. Hui, and S. Liu, “Progressive language-customized visual feature learning for one-stage visual grounding,” *IEEE Transactions on Image Processing*, vol. 31, pp. 4266–4277, 2022.
- [40] Y. Du, Z. Fu, Q. Liu, and Y. Wang, “Visual grounding with transformers,” *arXiv preprint arXiv:2105.04281*, 2021.
- [41] J. Deng, Z. Yang, T. Chen, W. Zhou, and H. Li, “Transvg: End-to-end visual grounding with transformers,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 1769–1779.
- [42] M. Li and L. Sigal, “Referring transformer: A one-step approach to multi-task visual grounding,” *Advances in Neural Information Processing Systems (NIPS)*, vol. 34, pp. 19652–19664, 2021.
- [43] 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 Transactions on Multimedia*, 2022.
- [44] L. Yang, Y. Xu, C. Yuan, W. Liu, B. Li, and W. Hu, “Improving visual grounding with visual-linguistic verification and iterative reasoning,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 9499–9508.
- [45] J. Ye, J. Tian, M. Yan, X. Yang, X. Wang, J. Zhang, L. He, and X. Lin, “Shifting more attention to visual backbone: Query-modulated refinement networks for end-to-end visual grounding,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 15502–15512.
- [46] P. Young, A. Lai, M. Hodosh, and J. Hockenmaier, “From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions,” *Transactions of the Association for Computational Linguistics*, vol. 2, pp. 67–78, 2014.
- [47] S. Guadarrama, E. Rodner, K. Saenko, N. Zhang, R. Farrell, J. Donahue, and T. Darrell, “Open-vocabulary object retrieval,” in *Robotics: science and systems*, vol. 2, no. 5, 2014, p. 6.
- [48] S. Kazemzadeh, V. Ordonez, M. Matten, and T. Berg, “Referitgame: Referring to objects in photographs of natural scenes,” in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 787–798.
- [49] 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 *Proceedings of the IEEE international conference on computer vision (ICCV)*, 2015, pp. 2641–2649.
- [50] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma *et al.*, “Visual genome: Connecting language and vision using crowdsourced dense image annotations,” *International journal of computer vision (IJCV)*, vol. 123, no. 1, pp. 32–73, 2017.
- [51] H. De Vries, F. Strub, S. Chandar, O. Pietquin, H. Larochelle, and A. Courville, “Guesswhat?! visual object discovery through multi-modal dialogue,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 5503–5512.
- [52] J. Johnson, B. Hariharan, L. Van Der Maaten, L. Fei-Fei, C. Lawrence Zitnick, and R. Girshick, “Clevr: A diagnostic dataset for compositional language and elementary visual reasoning,” in *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2017, pp. 2901–2910.
- [53] A. Nguyen, Q. D. Tran, T.-T. Do, I. Reid, D. G. Caldwell, and N. G. Tsagarakis, “Object captioning and retrieval with natural language,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 2019, pp. 0–0.
- [54] R. Liu, C. Liu, Y. Bai, and A. L. Yuille, “Clevr-ref+: Diagnosing visual reasoning with referring expressions,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 4185–4194.
- [55] T. Deruyttere, S. Vandenhende, D. Grrijicic, L. Van Gool, and M. F. Moens, “Talk2car: Taking control of your self-driving car,” 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, pp. 2088–2098.
- [56] P. Wang, D. Liu, H. Li, and Q. Wu, “Give me something to eat: referring expression comprehension with commonsense knowledge,” in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 28–36.
- [57] S. Yang, G. Li, and Y. Yu, “Graph-structured referring expression reasoning in the wild,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 9952–9961.
- [58] Z. Chen, P. Wang, L. Ma, K.-Y. K. Wong, and Q. Wu, “Cops-ref: A new dataset and task on compositional referring expression comprehension,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 10086–10095.
- [59] C. Wu, Z. Lin, S. Cohen, T. Bui, and S. Maji, “Phrasectut: Language-based image segmentation in the wild,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 10216–10225.
- [60] H. Liu, A. Lin, X. Han, L. Yang, Y. Yu, and S. Cui, “Refer-it-in-r3db: A bottom-up approach for 3d visual grounding in rgbd images,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 6032–6041.
- [61] R. Snow, B. O’Connor, D. Jurafsky, and A. Y. Ng, “Cheap and fast—but is it good? evaluating non-expert annotations for natural language tasks,” in *Proceedings of the 2008 conference on empirical methods in natural language processing (EMNLP)*, 2008, pp. 254–263.
- [62] R. A. Krishna, K. Hata, S. Chen, J. Kravitz, D. A. Shamma, L. Fei-Fei, and M. S. Bernstein, “Embracing error to enable rapid crowdsourcing,” in *Proceedings of the 2016 CHI conference on human factors in computing systems*, 2016, pp. 3167–3179.
- [63] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” *Advances in neural information processing systems*, vol. 30, 2017.
- [64] M. Dehghani, S. Gouws, O. Vinyals, J. Uszkoreit, and L. Kaiser, “Universal transformers,” in *International Conference on Learning Representations (ICLR)*, 2018.
- [65] J. D. M.-W. C. Kenton and L. K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” in *Proceedings of NAACL-HLT*, 2019, pp. 4171–4186.
- [66] R. Girshick, “Fast r-cnn,” in *Proceedings of the IEEE international conference on computer vision (ICCV)*, 2015, pp. 1440–1448.
- [67] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, “Generalized intersection over union: A metric and a loss for bounding box regression,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, 2019, pp. 658–666.
- [68] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, “End-to-end object detection with transformers,” in *European conference on computer vision (ECCV)*, 2020, pp. 213–229.
- [69] I. Loshchilov and F. Hutter, “Decoupled weight decay regularization,” in *International Conference on Learning Representations (ICLR)*, 2018.
- [70] X. Glorot and Y. Bengio, “Understanding the difficulty of training deep feedforward neural networks,” in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, 2010, pp. 249–256.
- [71] K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask r-cnn,” in *Proceedings of the IEEE international conference on computer vision (ICCV)*, 2017, pp. 2961–2969.