Word2Pix: Word to Pixel Cross-Attention Transformer in Visual Grounding

Heng Zhao†, Joey Tianyi Zhou‡, Senior Member, IEEE, and Yew-Soon Ong§, Fellow, IEEE

Abstract—Current one-stage methods for visual grounding encode the language query as one holistic sentence embedding before fusion with visual features for target localization. Such a formulation provides insufficient ability to model query at the word level, and therefore is prone to neglect words that may not be the most important ones for a sentence but are critical for the referred object. In this article, we propose Word2Pix: a one-stage visual grounding network based on the encoder–decoder transformer architecture that enables learning for textual to visual feature correspondence via word to pixel attention. Each word from the query sentence is given an equal opportunity when attending to visual pixels through multiple stacks of transformer decoder layers. In this way, the decoder can learn to model the language query and fuse language with the visual features for target prediction simultaneously. We conduct the experiments on RefCOCO, RefCOCO+, and RefCOCOg datasets, and the proposed Word2Pix outperforms the existing one-stage methods by a notable margin. The results obtained also show that Word2Pix surpasses the two-stage visual grounding models, while at the same time keeping the merits of the one-stage paradigm, namely, end-to-end training and fast inference speed. Code is available at https://github.com/azurerain7/Word2Pix.

Index Terms—Cross-attention, deep learning, multimodal, referring expression comprehension, visual grounding.

I. INTRODUCTION

D eepl learning has been dominant in computer vision research and industry for a wide range of applications from action recognition [1], medical image processing [2], [3] to remote sensing [4]–[6] over the past decades. Vision is crucial for a machine to understand the real world; however, a single modality is not enough. Recent advances in multimodal learning [7]–[9] show that the addition of language modality enables AI access to an even wider range of interesting capabilities such as visual grounding.

Manuscript received 16 July 2021; revised 14 May 2022; accepted 9 June 2022. Date of publication 24 June 2022; date of current version 6 February 2024. This work was supported in part by the Joey Tianyi Zhou’s A*STAR Science and Engineering Research Council (SERC) Central Research Fund (Use-inspired Basic Research), in part by the Singapore Government’s Research, Innovation and Enterprise 2020 Plan (Advanced Manufacturing and Engineering domain) under Grant A18A1B0045, in part by the A*STAR Center for Frontier AI Research (CFAR), and in part by the Nanyang Technological University (NTU) Data Science and Artificial Intelligence Research Center. (Corresponding author: Joey Tianyi Zhou.)

Heng Zhao and Joey Tianyi Zhou are with the A*STAR Centre for Frontier AI Research (CFAR), Singapore 138632 (e-mail: zhaoh@ihpc.a-star.edu.sg; zhouyt@ihpc.a-star.edu.sg).

Yew-Soon Ong is with the School of Computer Science and Engineering, Nanyang Technological University, Singapore 639798, and also with the A*STAR Centre for Frontier AI Research (CFAR), Singapore 138632 (e-mail: assyong@ntu.edu.sg).

Digital Object Identifier 10.1109/TNNLS.2022.3183827

Visual grounding, or referring expression comprehension, is the task of localizing the object instance in an image with a bounding box given a natural language expression that describes the referred object. The existing methods for visual grounding can be split into two categories: two-stage and one-stage strategies. Two-stage grounders use pretrained off-the-shelf object detectors to generate all possible candidate object regions which are later ranked based on the similarity to the input language query. State-of-the-art performance [10]–[12] is achieved for their capability of modeling the relationship between the well-parsed language query and all off-the-shelf candidate regions. However, two-stage methods suffer from redundant computational and slow inference time (usually at a few hundreds of milliseconds) due to bounding box refinement and proposal generation process. Moreover, their performances are bound by the quality of region proposals from pretrained detectors whose model weights cannot be optimized for visual grounding objectives during training.

To improve inference efficiency and enable end-to-end training, many one-stage methods [13]–[16] are gaining interest recently. Unlike two-stage grounders, one-stage networks need to learn a joint feature representation for bounding box prediction, and therefore, the fusion between language and vision feature becomes the core of the visual grounding task for the one-stage methods. As shown in Fig. 1(a) [14], language query is encoded into one sentence-level vector embedding and this vector is duplicated and fused with visual feature at each spatial location for bounding box regression. In this formulation, the language modality can only provide sentence-level guidance for visual localization because the embedding for each word is implicitly merged as one single-sentence vector. This is a disadvantage as each individual word cannot directly interact with the visual feature so the importance of each word is diluted. To further illustrate this with Fig. 1(a), the cosine similarity between the sentence embedding (obtained from the pretrained BERT [17] model) of input query cat sitting under the chair and cat sitting on the chair is 0.986, and therefore, models with holistic sentence embedding fail to emphasize the critical word under or on because this single word shares the same sentence embedding in other words. As a result, individual words lose the chance to independently guide the network to focus on the corresponding visual feature.

However, it is challenging to learn attention between multiple visual pixel locations and multiple independent words from a query, while at the same time model the language query to exploit the context of these words. To address this...
challenge, we propose Word2Pix: a one-stage visual grounding network based on the transformer architecture to learn word to pixel attention and feature fusion for visual grounding simultaneously. Our work is inspired by the success of transformers [18] in NLP research and computer vision [19]. As illustrated in Fig. 1(b), the transformer decoder takes individual word token embedding as input, enabling each word to guide language to visual attention independently. To be specific, the first layer of the decoder takes the self-attended word tokens as query and flattened visual feature as key and value (see Fig. 2) to learn the cross-attention between words and pixels. This straightforward yet effective method enables word to pixel attention globally and at the same time takes the context of each word into consideration by self-attending the word tokens. Some visualization examples are shown in latter section (see Fig. 4). Instead of learning attention with holistic sentence embedding, learning word to pixel attention provides a chance for each word in a sentence to impact the grounding results individually so that the network can learn to focus on words that are critical for visual grounding rather than words that dominate the sentence embedding. Through multiple stacks of decoder layers, Word2Pix learns to fuse visual feature with grounding-critical language feature to generate a final joint representation for bounding box regression. Experiments on three referring expression datasets (RefCOCO [20], RefCOCO+ [20], and RefCOCOg [21]) show that our method greatly extends the performance of one-stage methods to surpass state-of-the-art two-stage ones, at the same time keeps the benefits of one-stage methods for fast inference speed and end-to-end training.

We summarize our contributions in the following points.

1) **We Propose Word2Pix:** A one-stage visual grounding framework with word to pixel attention, enabling the network to consistently adjust the attention of each word on referent object and allowing a chance for individual word to guide the language–visual attention independently.

2) By modeling word to pixel attention via the encoder–decoder transformer architecture in a straightforward way, we improve the robustness and interpretability of the existing one-stage methods. The simplicity also leads to strong transferability, allowing the model to benefit from relevant training objectives without using extra data.

3) Our method extends the performance of the existing one-stage by a margin to surpass two-stage methods, setting new state-of-the-art on three datasets and at the same time keeping fast inference speed.

II. RELATED WORK

A. Visual Grounding

The visual grounding algorithms can be categorized as two-stage and one-stage methods. The former localizes referent objects in two steps: 1) extract visual features for candidate objects from off-the-shelf detectors and 2) rank language query–object candidate pairs and select the candidate with the highest ranking score. Although in the second step different modeling techniques can be exploited such as context modeling [21]–[26], modular attention [10]–[12], and graph and tree modeling [27]–[29], the two-stage methods still have to rely on off-the-shelf object detectors [30], [31] to extract visual features for candidate objects. As a result, two-stage methods suffer from performance bottleneck from proposal quality and slow inference speed.

To address these disadvantages, many one-stage methods [13]–[15], [32] are proposed recently. The one-stage networks enjoy the benefits of end-to-end training and fast inference speed. However, how to effectively learn to fuse features from two modalities becomes a core problem for them. Sadhu et al. [13] attend the query sentence embedding to different spatial locations of the visual feature and then aggregate all attended feature for feature fusion. Yang et al. [14] simply concatenate the holistic query embedding to all spatial locations as fusion. Liao et al. [16] use the query embedding as a correlation filter to convolve with the visual feature to locate the center of the referred object. Luo et al. [32] take the hidden state vector from GRU [33] as the query embedding when fusing with visual features. Almost all one-stage methods encode the query sentence into a single vector representation for ease of feature fusion. As a result, they are prone to neglect important information from a query sentence. To improve this situation, [15] proposes to use the weighted sum of all word token embedding as the final query embedding for fusion where the weight of each word is iteratively adjusted through multiple rounds of learning. However, the method is still subject to neglecting important words since the query embedding for each round remains a single vector.

B. Vision Transformer

The transformer architecture is originally proposed in [18] for the neural machine translation task in natural language
processing. Inspired by the success of the self-attention mechanism, researchers have explored to use the transformer framework for fundamental vision tasks such as image classification [34], [35], object detection [19], [36], [37], segmentation [38], and other low-level vision tasks such as image generation, super-resolution, and denoising [39], [40]. Dosovitskiy et al. [35] split an image into patches and feed them into a stack of transformer encoder to learn a representation for classification. Chen et al. [34] leverage generative pr-training to learn visual representation in a self-supervised way with stacks of transformer decoder block. The encouraging work DETR proposed in [19] formulates object detection as a set prediction problem, eliminating the need for proposal generation, anchor box design, and non-maximal suppression (NMS) all at once. This work sets a new paradigm for object detection and inspired many following works such as deformable DETR [36] and transformer-based set prediction (TSP) [37]. Zhu et al. [36] proposed deformable attention operation to solve the convergence issue in [19], while in [37] an encoder-only transformer-based detector is proposed.

C. Multimodal Transformer

The encoder–decoder architecture of the transformer can be conveniently adapted to multimodal tasks such as captioning, question-answering, reasoning, and visual grounding. VideoBERT proposed in [41] learns joint video-text representation with transformers in a self-supervised way for downstream tasks. A multimodal transformer is proposed in [42] for image captioning where object proposals generated from detectors are fed into the encoder, while the decoder learns a language model conditioned by the encoder outputs. Language and vision pretraining is becoming a trend in this field [43]–[45].

In general, the existing multimodal transformers focus on generic learning for vision-language feature representations with various pretraining techniques on large-scale datasets such as Visual Genome [46], and then the model can be fine-tuned on downstream tasks such as referring expression comprehension. However, the existing transformer-based models take region proposals from off-the-shelf object detectors as input for visual features and follow the two-stage paradigm on the visual grounding task, where they suffer from the same shortcoming of slow inference speed and bottleneck from object detector proposals. In contrast, our method focuses on building direct word to pixel attention while at the same time keeping the merits of one-stage methods such as end-to-end training and efficient computation.

III. Method

In this work, we describe the proposed architecture in detail. Inspired by the success of self-attention and cross-attention mechanism used in transformers [18] for NLP tasks, we model our word to pixel attention with the transformer framework. To be specific, an encoder–decoder transformer is proposed where the encoder is directly connected to the CNN backbone for visual feature extraction and the decoder enables guidance from language to visual feature at the word level and pixel level, respectively. We will review the attention mechanism and its usage in transformers as preliminaries in Section III-A and introduce our proposed architecture in Section III-B.
A. Preliminaries: Multi-Head Attention in Transformers

The attention mechanism is generally used to generate a new feature representation from original feature vectors which is specifically attended by another query feature. For instance, with given vector \( Q, K, \text{ and } V \) as query, key, and value vectors, respectively, the attended vector is a weighted sum of value vectors where the attention weights are calculated with dot-product of query and key vectors [18]

\[
\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_{\text{dim}}}}\right)V
\]  

where \( d_{\text{dim}} \) is the dimension of vectors. The above attended vector is considered as a single-head attention, and multiple single-head attention vectors are concatenated and scaled to form a multi-head attention. Note that the length of the attended output vector is aligned with the length of the query vector. Query can be from any source, but key and value vectors are usually from the same origin. Specifically for self-attention in language representation learning, query, key, and value are all from the same sentence. This enables the model to learn word vector embedding in a context-aware manner. Additionally in [18], a stackable transformer layer is formed by adding layer normalization (LN) [47] and a multilayer perceptron (MLP) after multi-head self-attention (MHSA) operation together with residual connections.

B. Word2Pix: One-Stage Visual Grounding With Transformer

We build our model with three parts: 1) visual encoding branch; 2) word to pixel attention branch; and 3) targets’ prediction head.

1) Visual Encoding Branch: The vision branch includes a CNN backbone from ResNet [48] for general visual feature extraction and a stack of transformer encoder layers.

Concretely, given an RGB image input with shape \( H \times W \times 3 \), the image feature from the output of mid-layer CNN is of shape \((H/r) \times (W/r) \times C\) where \( C \) is the number of feature maps. The feature maps are then flattened along the spatial axis and transformed to a sequence of vectors \( \{x_k\}_{k=1}^{h\times w} \) where \( h = (H/r), w = (W/r) \), and \( x_k \in \mathbb{R}^D \). As a default, the dimension of feature maps from the ResNet backbone is set to \( C = 2048 \) and is reduced to \( D = 256 \) via a linear layer. The downsampling rate \( r \) is picked as 32 to keep the length of the sequence at a reasonable number for the following self-attention-based module. Since the self-attention mechanism is pair-wise and ignores the order of the input sequence, positional encoding has to be added for the flattened visual feature. Thus, the input visual features for the transformer encoder layer are denoted as

\[
F^0_v = \{x_k + \text{PosEmb}_b(k)\}_{k=1}^{h\times w}
\]  

where \( \text{PosEmb}_b(.) \) is the sine positional encoding in [19].

The feature after each transformer encoder layer can be denoted as

\[
F^{i+1}_v = F^i_v + \text{MHSA}(F^i_v) + \text{FFN}(F^i_v + \text{MHSA}(F^i_v))
\]  

where \( \text{FFN(·)} \) is a two-layer MLP and \( \text{MHSA} \) is the MHSA described in Section III-A. LN is omitted here for brevity.

2) Word to Pixel Attention Branch: In this branch, we model word to pixel attention: the correspondence between individual word tokens and pixel-level visual features with the transformer decoder. Specifically, given a natural language query sentence comprising many words \( \mathcal{S} = \{w_k\}_{k=1}^T \), where \( T \) is the length of the sentence and \( w_k \in \mathbb{R}^D \) is the embedding for each word. The embeddings of words can either be learned with an embedding matrix or be obtained with a pretrained language model. These word embeddings serve as the sequential query input \( F^0_{i} \) for the decoder layer

\[
F^0_i = \{w_k + \text{PosEmb}(k)\}_{k=1}^{T+1}
\]  

where \( \text{PosEmb}(·) \) is a learnable positional embedding matrix with random initialization. In contrast to the existing one-stage methods which consider holistic sentence embedding for feature fusion, our formulation keeps each word embedding as independent input for each decoder layer. As a comparison, [15] tries to solve this query modeling issue by encoding the sentence embedding as \( q = \sum_{k=1}^{T} a_k w_k \) where \( q \in \mathbb{R}^D \) is of the same dimension with each word embedding \( w_k \). However, this single vector is the weighted sum of all word embeddings, and thus the importance of individual words becomes diluted and the relationships between different words when given a visual feature cannot be modeled.

After \( M \) stacks of encoder layer, the output \( F^M_v = \{\hat{x}_k\}_{k=1}^{h\times w} \) can be considered as a spatial-context-aware pixel vector sequence. With this vector sequence as the visual memory for an input image, we build the word to pixel attention between self-attended word embedding sequence \( \hat{F}_v = \text{MHSA}(F_v^i) = \{\hat{w}_k\}_{k=1}^{T+1} \) and pixel feature sequence \( F_v^M = \{\hat{x}_k\}_{k=1}^{h\times w} \) in each decoder layer. Self-attention is applied to the word embedding sequence to generate context-aware feature, followed by word to pixel cross-attention:

\[
\hat{F}_i^{i+1} = \hat{F}_i^i + \text{MHSA}(\hat{F}_i^i) + \text{FFN}(\hat{F}_i^i + \text{MHSA}(\hat{F}_i^i, F_v^M)).
\]  

The multi-head cross-attention (MHCA) for each head is computed as

\[
\text{MHCA}_{\text{head}}(\hat{F}_i^i, F_v^M) = \text{softmax}\left(\frac{Q_i K_i^\top}{\sqrt{d_{\text{dim}}}}\right)V_i
\]

where \( Q_i, K_i, \text{ and } V_i \) are the learnable weight matrices for query, key, and value, respectively. Note the cross-attention weight is of shape \((w \times h) \times (T + 1)\) where each textual token is attending to all pixel locations independently.

3) Targets Prediction: Instead of only using bounding box regression as the optimization target, our grounding model also predicts the category and attributes of the referred object with the same high-level feature with rich semantic information. The source of value and key vectors in MHCA block for every layer of the decoder is the memory visual feature from the encoder, and the output is a weighted sum of this memory

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
feature which is also from visual space. Thus, the semantics of this visual feature at the last decoder layer can naturally be used for all three vision-related prediction tasks. We use the visual representation attended by the dummy token (CLS) for all prediction heads.

A two-layer MLP with ReLU activation is constructed for bounding box regression predicting the center coordinates, height, and width of the target object. The classification heads for category and attributes’ prediction are two simple linear layers followed by sigmoid function, respectively.

4) Optimization Objectives: We optimize the proposed grounding model with four losses

\[ L = L_{\text{ce}} + \lambda_{\text{bce}} L_{\text{bce}} + \lambda_{i} L_{i} + \lambda_{\text{giou}} L_{\text{giou}}. \] (7)

Specifically, given three ground-truth labels: class label \( Y = \{Y_{i}\}_{i=1}^{N_{c}} \), attribute labels \( y = \{y_{i}\}_{i=1}^{N_{a}} \), and location bounding box coordinates \( b \) of the referent object, the four losses are noted as follows:

\[ L_{\text{ce}} = -\frac{1}{N_{c}} \sum_{i=1}^{N_{c}} y_{i} \log(\hat{Y}_{i}) \]

\[ L_{i} = \|b - \hat{b}\|_{1} \sum_{[i : y_{i} = 1]} \log(\hat{Y}_{i}) \]

\[ L_{\text{bce}} = -\frac{1}{|\{i : y_{i} = 1\}|} \sum_{[i : y_{i} = 1]} \log(\hat{Y}_{i}) \]

\[ -\frac{1}{|\{i : y_{i} = 0\}|} \sum_{[i : y_{i} = 0]} \log(1 - \hat{Y}_{i}) \] (8)

where \( N_{c} \) and \( N_{a} \) are the number of categories and attributes, respectively. \( \hat{Y}_{i} \) is the softmax of predicted logits for class label \( i \). \( \hat{b} \) denotes the predicted bounding box coordinates. The generalized IoU loss \( L_{\text{giou}} \) follows [49]. For multi-label prediction, we use a dynamically weighted binary cross-entropy loss where \( |\{i : y_{i} = 1\}| \) denotes the number of positive attribute labels that one object has.

IV. EXPERIMENTS

A. Datasets

We conduct our experiments on three datasets: RefCOCO [23], RefCOCO+ [23], and RefCOCOg [21] where the images are mostly from MSCOCO [50]. There are 19994/19992/25799 images with 50000/49856/49822 referred entities and 142210/141564/95010 expressions in RefCOCO, RefCOCO+, and RefCOCOg respectively. The average number of words for expressions in RefCOCO and RefCOCO+ is 3.5, while the sentence in RefCOCOg is longer with 8.4 words on average. Compared with RefCOCO, expressions in RefCOCO+ are not allowed to contain words indicating absolute location of the referred object.

In terms of evaluation, the test set is split into testA and testB for RefCOCO and RefCOCO+ where testA focuses on person, while testB focuses on object. There are two splits for RefCOCOg, and we follow the split RefCOCOg-umd used in [23] where images in the training, validation, and testing sets do not overlap.

B. Implementation and Training Details

The images are resized with the shorter side as 640 with a maximum longer side as 1333. The original aspect ratio is maintained, and zero-valued pixels are padded for batch training. No other data augmentation techniques are used during training. We choose ResNet-101 as our feature extractor, and high-level layers are discarded. Six layers of encoders are stacked on top of the backbone CNN with LN applied after the attention operation. For the language branch, the maximal input length is set to 10 and 15 for the decoder. Dummy token (CLS) is added at the beginning of the input sentence, and (PAD) is appended at the end to support batch training. Words are tokenized as indices in vocabulary and embedded with a learnable matrix where the vocabulary size of all expressions in RefCOCO is around 2000. In case embedding from BERT is used, a linear layer is added to project the dimension from 768 to 256, and the embedding from the dummy token (CLS) is used as the feature for the whole sentence. Note that the BERT tokenizer is a sub-word tokenizer, and thus the tokenized sentence length will be longer than the length of textual expression. We trim the token embedding sequence if the length exceeds the maximal input length of the decoder.

For training objectives, there are 80 categories for classification loss and 50 attributes for binary cross-entropy loss. The ground-truth attribute labels are very sparse where the value of most labels is zero for one object, and thus the attribute loss is dominated by negative samples. We implement a mean averaged attribute loss to balance the positive and negative labels.

In the current study, for the sake of reproducibility we considered a transformer encoder that is built on top of the DETR [19] and initialize the corresponding weights from models pretrained on MSCOCO [50]. Note that all the validation and testing images in RefCOCO, RefCOCO+, and RefCOCOg are also included in the MSCOCO training dataset. For fair comparison, we retrain a model from scratch excluding all the validation and testing images from the grounding dataset for 100 epochs. The model is optimized with AdamW [57] with a batch size of 6, and the initial learning rate is set to \( 10^{-4} \) and \( 10^{-5} \) for the transformer and CNN backbone, respectively. The learning rate is decayed by 10 at an epoch of 160 until training stops at epoch of 180. Due to the fact that the encoder is initialized from pretrained weights while the decoder is randomly initialized. We freeze the CNN backbone and encoder weights during the first 80 epochs. The weight coefficient for \( L_{\text{ce}} \), \( L_{\text{bce}} \), \( L_{i} \), and \( L_{\text{giou}} \) is set to 1, 10, 5, and 2, respectively.

C. Comparison With State-of-the-Art Methods

We report our grounding results in comparison to the existing one-stage and two-stage methods, as shown in Table I. Following the one-stage paradigm, we observed significant improvements (>5% on average) over the state-of-the-art one-stage methods on three datasets. As mentioned previously, all the existing one-stage methods use one or several sentence-level embedding for language query, and hence the
attention interaction between word tokens and visual feature has been neglected. In contrast, our Word2Pix model emphasizes attention from individual word to pixel-level visual features, bringing a large performance leap of $\sim 6.5\%$ on testA set of RefCOCO+.

Built on top of [14], [15] has been reported to offer better performance (67.20\%) over the former on the RefCOCOg test set where the average query sentence length is longer. Nonetheless, the Word2Pix model is shown to extend the performance of those reported in [15] by a good margin of $\sim 4\%$ on the same set, thus illustrating the importance of word to pixel attention, hence the benefits of our proposed approach.

We also compare Word2Pix with the current state-of-the-art two-stage method [12]. We observe a performance boost of $\sim 2\%$ and $\sim 3\%$ on the testB set of RefCOCO and testA of RefCOCO+ dataset, respectively. Small gains on validation and test set of other datasets are also observed. When compared with the transformer-based two-stage multimodal methods [44], [45], we reference their results obtained without additional training data for a fair comparison. Aligned with other one-stage methods, our model maintains a real-time inference speed while two-stage methods must make a tradeoff between accuracy and inference speed because they need massive region proposals to guarantee a high recall rate of the ground-truth candidate. As a matter of fact, recent state-of-the-art two-stage grounding models [11], [12] share the same framework with MAttNet [10] where the visual extractor is the pretrained mask-RCNN [31] and the grounding module is the modular attention from MAttNet [10]. In all the three methods, the visual extractor weights are fixed and not optimized for grounding tasks. In contrast, our proposed one-stage model is able to tune the visual extractor with grounding supervision signals in an end-to-end way. We show that being able to train the visual feature extractor is important for us to beat all two-stage methods in ablation experiments.

We note that the maximum number of input tokens from the query sentence and computational efficiency remains a tradeoff to be considered. In the present study, the number of input word tokens is set at 15 for the experiments since the majority of the sentences do not exceed this number. On RefCOCOg despite some of the long sentences being trimmed, Word2Pix is shown to outperform the state-of-the-art one-stage methods that use holistic sentence embedding by $\sim 4\%$.

### D. Complexity Analysis and Inference Speed

As shown in (6), the cross-attention between words and pixel is calculated by matrix multiplication. Given sentence length $n$, visual feature map size $hw$, and common feature space dimension $d$, the computational complexity of one decoder layer is $O(nd^2 + n^2d) + O((hw+n)d^2 + hwnd)$ where the first term is for self-attention between word tokens and the second is for word to pixel cross-attention.$^1$ Likewise, the computational complexity for one encoder layer is $O(hwfd^2 + (hw)^2d)$ which is much less than that of a decoder layer for $n \ll hw$. In our implementation, the query length $n$ is capped to 10 and 15, and the visual feature size has been downsampled from $640 \times 640$ (simplified for illustration) by a factor of $r = 32$ to $20 \times 20$ by the CNN backbone. The FLOPS required by one encoder layer is roughly 213 million.

As a comparison, the computational complexity of a single convolutional layer is $O(k^2(HW)^2C_{in}C_{out})$ where $k$ is the kernel size and $C_{in}$, $C_{out}$ is the input and output channel number. If we look at the very first layer of ResNet-101 when

---

**TABLE I**

| Methods       | Venue | Backbone | RefCOCO | RefCOCO+ | RefCOCOg |
|---------------|-------|----------|---------|----------|----------|
|               |       |          | val     | testA(%) | testB(%) | val     | testA(%) | testB(%) | val     | testA(%) | testB(%) |
| ParalAttn [51] | CVPR’18 | vgg16    | –       | 75.31    | 65.52    | –       | 61.34    | 50.86    | –       | –        | –        |
| NMTTree [29]  | ICCV’19 | vgg16    | 71.65   | 74.81    | 67.34    | 58.00   | 61.09    | 53.45    | 61.01   | 61.46    |
| LGRAN [52]    | CVPR’19 | vgg16    | –       | 76.60    | 66.40    | –       | 64.00    | 53.40    | –       | –        | –        |
| DGA [28]      | ICCV’19 | vgg16    | –       | 78.42    | 65.53    | –       | 69.07    | 51.99    | –       | 63.28    |
| RvGTTree [53] | TPAMI’19 | ResNet-101 | 75.06  | 78.61    | 69.85    | 63.51   | 67.45    | 56.66    | 66.95   | 66.51    |
| MAttNet [10]  | CVPR’18 | ResNet-101 | 76.65  | 81.14    | 69.99    | 65.33   | 71.62    | 56.02    | 66.58   | 67.27    |
| VL-BERT [44]  | ICLR’20 | ResNet-101 | –      | –       | –       | 66.03   | 71.87    | 56.13    | –       | –        | –        |
| CMAE [11]     | CVPR’19 | ResNet-101 | 78.35  | 83.14    | 71.32    | 68.09   | 73.65    | 58.03    | 67.99   | 68.67    |
| Ref-NMS [12]  | AAAI’21 | ResNet-101 | 80.70  | 84.00    | 76.04    | 68.25   | 73.68    | 59.42    | 70.55   | 70.62    |
| ViLBERT [45]  | NIPS’19 | ResNet-101 | –      | –       | –       | 68.61   | 75.97    | 58.44    | –       | –        | –        |
| Word2Pix(ours) | –     | ResNet-101 | 81.20  | 84.39    | 78.12    | 69.74   | 76.11    | 61.24    | 70.81   | 71.34    |
feeding with a 640 × 640 RGB image, the FLOPS will be 963 million,\(^2\) whereas the FLOPS of the whole ResNet-101 backbone is 7.6 billion [48]. This illustrates that our Word2Pix model only adds negligible computational cost when compared with the current one-stage grounders whose cost is mainly from the CNN backbone. We tested the inference speed of FAOA [14], ReSC-Large [15], and Word2Pix on a Tesla V100 GPU card, and the average latency is 28, 29, and 38 ms, respectively. As for the two-stage methods [10], the object detector alone costs more than 250 ms.

E. Ablation Studies

In this ablation study, we show evidence that our proposed end-to-end network is able to overcome the shortcomings of the traditional one-stage and two-stage methods through several designed ablation experiments on the RefCOCO [23] dataset.

1) Word to Pixel Versus Sentence to Pixel: The current state-of-the-art one-stage methods encode the holistic sentence query into one [14], [16], [54] or several [15] single vector embedding for textual and visual feature fusion. In this ablation experiment, we demonstrate that the proposed word to pixel attention is superior to sentence to pixel attention which uses multiple holistic sentence-level representations. Specifically, three experiment settings are designed.

1) Replace decoder query inputs where each query token is a single vector sentence embedding from the pretrained BERT model [17]. We take vector embedding of (CLS) token from the last four layers of the pretrained BERT model as sentence-level embedding, resulting in four different sentence-level embeddings for given language query input. At the last layer of the decoder, output feature vectors of the four sentence-level embeddings are pooled and forwarded to the prediction head.

2) Instead of using the pretrained BERT feature for word token embedding, a word embedding matrix is learned with the dataset’s vocabulary during training.

3) Word and (CLS) token embeddings are obtained from the last layer of the pretrained BERT model.

The results are shown in Table II. We observe that only using sentence-level embeddings for the grounding task yields inferior performance, even compared with models with word embedding learned directly from the dataset’s vocabulary. We note that the vocabulary size for all language queries in RefCOCO dataset is only 2000, and therefore, the language embedding learned from the dataset is weak compared with that from a pretrained BERT model. This demonstrates the significance of word-level modeling and the superiority of our proposed framework.

2) Off-the-Shelf Visual Feature Versus End-to-End Training: One key disadvantage of the traditional two-stage methods is that the visual feature extractors or detectors cannot be fine-tuned with visual grounding supervision signals. Instead, their weights are fixed during the training of grounding networks. However, visual features suitable for object classification and detection tasks may need to be refined for language-guided visual grounding tasks. For example, when an object detector is to decide whether a region of interest is a person or not, features for action, color, and pose would not contribute much to this category classification task. However, these features are critical for visual grounding and they are usually encoded in the lower layer of the CNN backbone. Unable to fine-tune the low-level visual extractor leads to inferior performance when compared with those models that are trained in an end-to-end way. To validate this, we initialize the CNN backbone and encoder of our model with weights pretrained on the MSCOCO object detection dataset (excluding RefCOCO/+/g validation/testing set images). Experiments on RefCOCO dataset with the following settings are conducted: 1) freeze CNN backbone and encoder weights and train word to pixel attention decoder only; 2) freeze CNN backbone only; and 3) end-to-end training with all model weights trainable. The results are shown in Table III. It can be observed that uniform improvements are obtained on all test and validation splits when more visual feature extractor weights are trainable. We observe a consistent improvement of ~5% on testA set where the referent objects all belong to person category. This demonstrates that training visual feature extractors in an end-to-end way is especially critical for referring to objects with class-agnostic features. However, when CNN+encoder weights are no longer trainable, the performance (75.58%) did not drop as much on testB set when compared with end-to-end training (78.12%). We argue that this is due to the fact that referents in testB are common objects instead of person; therefore, the number of same type of referent objects in one image is fewer. In this case, an off-the-shelf object detector has already contributed a good part to the accuracy.

3) Impact of Transferability and Pretraining: We investigate the impact of pretraining Word2Pix with detection objectives on the performance and also the transferability of the proposed method. To be specific, we initialize the visual encoder part with three settings: 1) initialize the CNN backbone with weights pretrained on MSCOCO (excluding RefCOCO/+/g validation/testing) and train the rest of the network from scratch; 2) initialize the CNN backbone and transformer

\(^2\)For the first convolutional layer of ResNet-101, \(C_{in} = 3, C_{out} = 64\), kernel size \(k = 7\), and the stride is 2.
encoder with weights pretrained on MSCOCO (only using RefCOCO/+/g training images); and 3) initialize the CNN backbone and transformer encoder with weights pretrained on MSCOCO (excluding RefCOCO/+/g validation/testing). Note that in this ablation, no weights are frozen during training. The results in Table IV show that the performance drops when the encoder is trained from scratch. We argue that this is due to the late fusion strategy of the proposed method. Word2Pix emphasizes the ability to exploit query structure and model individual word to pixel attention. However, the CNN backbone and encoder together form the visual extractor. Initializing only a part of the visual extractor is not ideal, yet the performance still rivals that of the current one-stage methods. To validate our assumption, we pretrain the CNN and encoder with RefCOCO training images, but with object detection objectives to provide a complete visual extractor base without using extra data. Note that this is different from the current two-stage works as the whole model weights will be trained in an end-to-end fashion with grounding objectives after initialization. We also observe that the accuracy on testB improves more than testA in this setting, which also validates the assumptions in the previous ablation study that a good pretrained helps improve the performance on testB.

4) Ablations on Network Architecture and Losses: We further conduct ablation experiments with different numbers of decoder layers and different supervision signal combinations, as shown in Table V. For fast verification purposes, we freeze the CNN backbone and encoder weights and only train the decoder from random initialization. Thus the baseline model here with all four losses and the number of decoder layers ($N = 3$) suffers a minor performance drop ($<1\%$) compared with the baseline model trained in an end-to-end way (main results in Table I). We note that the impact of attribute and classification loss is not significant compared with the model trained only with bounding box regression. We argue it is because in the RefCOCO dataset less than one-third of all referent objects have attribute labels, while the rest have nothing to be trained with.

F. Qualitative Analysis

We illustrate that our proposed method is able to overcome the disadvantage of one-stage methods by qualitative examples in Fig. 3. We feed free-text input (query sentence that is free
Fig. 4. Cross-attention weights between word tokens and pixel-level visual feature. For each image, two similar queries are tested with a different keyword shown in bold. For the three sub-rows of each row, the first and last sub-rows correspond to the first and final decoder layer attention weights, respectively. Brighter pixels represent higher attention values.

TABLE V

| ABLATION ON LOSS SETTINGS AND NUMBER OF DECODER LAYERS |
|-------------------------------------------------------|
| **Losses** | **RefCOCO** | **testA(%)** | **testB(%)** |
| $\mathcal{L}_{l_1}$ | $\mathcal{L}_{giou}$ | $\mathcal{L}_{ce}$ | $\mathcal{L}_{bce}$ | val | |
| $\checkmark$ | | | | 79.09 | 82.14 | 75.29 |
| $\checkmark$ | | | | 79.89 | 82.93 | 76.22 |
| $\checkmark$ | | | | 80.05 | 83.33 | 76.78 |
| $\checkmark$ | | | | 79.63 | 83.07 | 76.98 |
| $\checkmark$ | | | | 79.95 | 83.45 | 77.11 |
| $\checkmark$ | $\checkmark$ | | | 80.20 | 83.60 | 77.68 |
| Number of decoder layers | | | | |
| $N = 1$ | | | | 77.58 | 81.16 | 74.80 |
| $N = 2$ | | | | 79.06 | 82.52 | 75.72 |
| $N = 3$ | | | | 80.20 | 83.60 | 77.68 |
| $N = 4$ | | | | 80.07 | 83.15 | 77.29 |

input from user instead of directly quoting from the dataset) to FAQA [14], ReSC [15], and Word2Pix. Specifically, three pairs of similar queries are used to illustrate the weakness of using holistic sentence embedding when interacting with visual feature for visual grounding. We calculate the cosine similarities between the sentence embeddings of these three query pairs to show that the vector distance between holistic sentence embeddings is small (i.e., 0.986, 0.983, and 0.977, respectively). It can be seen that FAQA fails to distinguish the differences between input sentences, and ReSC is able to improve but is still not robust against subtle but critical changes in query sentences. As a comparison, Word2Pix is able to detect minor changes in queries because each word in a query is able to affect the final grounding results individually. For example, in the first two columns in Fig. 4, the embedding for word under and on contributes only a small part in the whole sentence embedding, but they are critical for referring to correct referent. It can also be observed that both FAQA and ReSC fail to note the critical word with sentence-level embedding, but our proposed method correctly localizes the referent.

We demonstrate the interpretability of Word2Pix by further visualizing the cross-modal attention weights between individual word tokens and pixel-level visual feature in Fig. 4. It can be observed that the same word token from two sentences attends to pixels in similar locations (1st, 2nd, 3rd, 5th, and 6th columns of the first sub-row), but the changed keywords from two sentences (4th column of the first sub-row) have different focuses in attention maps in the first decoder layer (first sub-row). As the inference goes deeper via decoder layers...
For both the comparisons, the first three images show our successful cases over other methods, while the last two show our failure cases. (IoU > 0.5), red for incorrect, and blue for ground-truth boxes. Results comparison on REFCOCO+ dataset: (a) versus MAttNet [10], (b) versus Ref-NMS [12]. For both the comparisons, the first three images show our successful cases over other methods, while the last two show our failure cases.

(2nd and 3rd sub-rows), this different focus will guide other word tokens to focus on the correct referent (cat under chair and cat on chair) via the language–vision feature fusion.

Qualitative comparison examples between our method and state-of-the-art two-stage methods are shown in Fig. 5. It can be seen that some of the incorrect predictions by the two-stage methods for referents with color or action attributes can be corrected by our method. Nonetheless, long query sentence with complex structure is still a challenging case for our model.

V. CONCLUSION AND FUTURE WORK

In this article, we propose Word2Pix: a one-stage visual grounding network that enables the word to pixel attention learning based on the encoder–decoder transformer architecture. The proposed method outperforms the state-of-the-art one-stage methods with a large margin and also surpasses the existing best two-stage models while at the same time maintaining the benefit of fast inference speed via end-to-end training. Ablation experiments and qualitative examples illustrate the importance and effectiveness of word to pixel attention over holistic sentence embedding for one-stage methods in visual grounding tasks. However, slow convergence in training and how to further explore the structure of the query sentence remain open challenges which will also be the possible direction for future work.

REFERENCES

[1] L. Wang et al., “Temporal segment networks: Towards good practices for deep action recognition,” in Computer Vision—(ECCV), B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham, Switzerland: Springer, 2016, pp. 20–36.
[2] O. Ronneberger, P. Fischer, and T. Brox, “U-Net: Convolutional networks for biomedical image segmentation,” 2015, arXiv:1505.04597.
[3] G. Litjens et al., “A survey on deep learning in medical image analysis,” Med. Image Anal., vol. 42, pp. 60–88, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1361841517301135
[4] X. Yang et al., “SCRDet: Towards more robust detection for small, cluttered and rotated objects,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2019, pp. 8231–8240.
[5] X. Sun, B. Wang, Z. Wang, H. Li, H. Li, and K. Fu, “Research progress on few-shot learning for remote sensing image interpretation,” IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens., vol. 14, pp. 2387–2402, 2021.
[6] Q. He, X. Sun, Z. Yan, and K. Fu, “DABNet: Deformable contextual and boundary-weighted network for cloud detection in remote sensing images,” IEEE Trans. Geosci. Remote Sens., vol. 60, pp. 1–16, 2022.
[7] A. Mogadala, M. Kalimuthu, and D. Klakow, “Trends in integration of vision and language research: A survey of tasks, datasets, and methods,” J. Artif. Intell. Res., vol. 71, pp. 1183–1317, Aug. 2021.
[8] J. Gao, P. Li, Z. Chen, and J. Zhang, “A survey on deep learning for multimodal data fusion,” Neural Comput., vol. 32, no. 5, pp. 829–864, May 2020.
[9] Y. Wei, X. Wang, W. Guan, and L. Nie, “Neural multimodal cooperative learning toward micro-video understanding,” IEEE Trans. Image. Process., vol. 29, pp. 1–14, 2020.
[10] 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.
[11] X. Liu, Z. Wang, J. Shao, X. Wang, and H. Li, “Improving referring expression grounding with cross-modal attention-guided erase,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 1950–1959.
[12] L. Chen, W. Ma, J. Xiao, H. Zhang, and S.-F. Chang, “Ref-NMS: Breaking proposal bottlenecks in two-stage referring expression grounding,” in Proc. AAAI, 2021, pp. 1036–1044.
[13] A. Sadhu, K. Chen, and R. Nevatia, “Zero-shot grounding of objects from natural language queries,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2019, pp. 4694–4703.
[14] 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), Oct. 2019, pp. 4683–4693.
[15] Z. Yang, T. Chen, L. Wang, and J. Luo, “Improving one-stage visual grounding by recursive sub-query construction,” in Proc. ECCV, 2020, pp. 387–404.
[16] 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), Jun. 2020, pp. 10877–10886.
[17] 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, MN, USA, vol. 1, Jun. 2019, pp. 4171–4186.

[18] A. Vaswani et al., “Attention is all you need,” in Proc. Adv. Neural Inf. Process. Syst., vol. 30, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds. Red Hook, NY, USA: Curran Associates, 2017, pp. 5998–6008.

[19] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, “End-to-end object detection with transformers,” in Proc. 16th Eur. Conf. ( Lecture Notes in Computer Science), vol. 12346, A. Vedaldi, H. Bischof, T. Brox, and J. Frahm, Eds. Glasgow, U.K.: Springer, 2020, pp. 213–229.

[20] S. Kazemzadeh, V. Ordonez, M. Matten, and T. Berg, “ReferItGame: Referring to objects in photographs of natural scenes,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 4418–4427.

[21] 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. (CVPR), Jun. 2019, pp. 4140–4149.

[22] Y. Liu, B. Wan, X. Zhu, and X. He, “Learning cross-modal context graph for visual grounding,” in Proc. AAAI Conf. Artif. Intell., 2020, pp. 11645–11652.

[23] S. Yang, G. Li, and Y. Yu, “Dynamic graph attention for referring expression grounding,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Los Alamitos, CA, USA, Jun. 2019, pp. 4140–4149.

[24] Y. Liu, B. Wan, X. Zhu, and X. He, “Learning cross-modal context graph for visual grounding,” in Proc. AAAI Conf. Artif. Intell., 2020, pp. 11645–11652.

[25] S. Yang, G. Li, and Y. Yu, “Cross-modal relationship inference for grounding referring expressions,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Los Alamitos, CA, USA, Jun. 2019, pp. 4140–4149.

[26] B. Zhuang, Q. Wu, C. Shen, I. Reid, and A. V. D. Hengel, “Parallel attention: A unified framework for visual object discovery through dialogues and queries,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 658–666.

[27] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” 2015, arXiv:1512.03385.

[28] 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 Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 13–23.

[29] 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, 2017.

[30] L. Liu, B. Wan, X. Zhu, and X. He, “Learning cross-modal context graph for visual grounding,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 4418–4427.

[31] 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. (CVPR), Salt Lake City, UT, USA, Jun. 2018, pp. 4158–4166.

[32] S. Yang, G. Li, and Y. Yu, “Cross-modal relationship inference for grounding referring expressions,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Los Alamitos, CA, USA, Jun. 2019, pp. 4140–4149.

[33] J. Lei Ba, J. Ryan Kiros, and G. E. Hinton, “Layer normalization,” 2016, arXiv:1607.06450.

[34] W. Su et al., “VI-BERT: Pre-training of generic visual-linguistic representations,” in 8th Int. Conf. Learn. Represent. (ICLR) Addis Ababa, Ethiopia, Apr. 2020. [Online]. Available: https://openreview.net/forum?id=SygX2EyVH

[35] J. Lu, D. Batra, D. Parikh, and S. Lee, “VIBERT: Pretraining task-agnostic viisualsiolinguistic representations for vision-and-language tasks,” in Proc. Adv. Neural Inf. Process. Syst., 32nd Annu. Conf. Neural Inf. Process. Syst. (NeurIPS), H. Wallach, H. Larochelle, A. Beygelzimer, F. Alché-Buc, E. Fox, and R. Garnett, Eds. Vancouver, BC, Canada, Dec. 2019, pp. 12–23.

[36] B. Zhuang, Q. Wu, C. Shen, I. Reid, and A. V. D. Hengel, “Parallel attention: A unified framework for visual object discovery through dialogues and queries,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 658–666.

[37] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” 2015, arXiv:1512.03385.

[38] Z. Yang, B. Gong, L. Wang, W. Huang, D. A. Shamma, and J. Luo, “Deep residual learning for image recognition,” 2015, arXiv:1512.03385.

[39] C. Sun, A. Myers, C. Vondrick, K. Murphy, and C. Schmid, “VideoBERT: A joint model for video and language representation learning,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2019, pp. 7464–7473.

[40] J. Yu, J. Li, Z. Yu, and Q. Huang, “Multimodal transformer with multi-view visual representation for image captioning,” IEEE Trans. Circuits Syst. Video Technol., vol. 30, no. 12, pp. 4467–4480, Dec. 2020.

[41] J. Lei Ba, J. Ryan Kiros, and G. E. Hinton, “Layer normalization,” 2016, arXiv:1607.06450.

[42] W. Su et al., “VI-BERT: Pre-training of generic visual-linguistic representations,” in 8th Int. Conf. Learn. Represent. (ICLR) Addis Ababa, Ethiopia, Apr. 2020. [Online]. Available: https://openreview.net/forum?id=SygX2EyVH

[43] J. Lu, D. Batra, D. Parikh, and S. Lee, “VIBERT: Pre-training task-agnostic viisualsiolinguistic representations for vision-and-language tasks,” in Proc. Adv. Neural Inf. Process. Syst., 32nd Annu. Conf. Neural Inf. Process. Syst. (NeurIPS), H. Wallach, H. Larochelle, A. Beygelzimer, F. Alché-Buc, E. Fox, and R. Garnett, Eds. Vancouver, BC, Canada, Dec. 2019, pp. 12–23.

[44] B. Zhuang, Q. Wu, C. Shen, I. Reid, and A. V. D. Hengel, “Parallel attention: A unified framework for visual object discovery through dialogues and queries,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2018, pp. 4522–4621.

[45] 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.

[46] 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.

[47] X. Chen, L. Ma, J. Chen, Z. Jie, W. Liu, and J. Luo, “Real-time referring expression comprehension by single-stage grounding network,” 2018, arXiv:1802.03426.

[48] 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), Oct. 2019, pp. 4682–4692.

[49] F. Yu, D. Wang, E. Shelhamer, and T. Darrell, “Deep layer aggregation,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Salt Lake City, UT, USA, Jun. 2018, pp. 2403–2412.

[50] I. Loshchilov and F. Hutter, “Decoupled weight decay regularization,” in Proc. 3rd Int. Conf. Learn. Represent. (ICLR), New Orleans, LA, USA, May 2019. [Online]. Available: https://openreview.net/forum?id=Bkg6RiGy7T