GTMS: A Gradient-driven Tree-guided Mask-free Referring Image Segmentation Method

Haoxin Lyu, Tianxiong Zhong, and Sanyuan Zhao*

Beijing Institute of Technology
{lvhaoxin1165, inkosizhong}@gmail.com
zhaosanyuan@bit.edu.cn

Abstract. Referring image segmentation (RIS) aims to segment an object of interest by a given natural language expression. As fully-supervised methods require expensive pixel-wise labeling, mask-free solutions supervised by low-cost labels are largely desired. However, existing mask-free RIS methods suffer from complicated architectures or insufficient utilization of structural and semantic information resulting in unsatisfactory performance. In this paper, we propose a gradient-driven tree-guided mask-free RIS method, GTMS, which utilizes both structural and semantic information, while only using a bounding box as the supervised signal. Specifically, we first construct the structural information of the input image as a tree structure. Meanwhile, we utilize gradient information to explore semantically related regions from the text feature. Finally, the structural information and semantic information are used to refine the output of the segmentation model to generate pseudo labels, which in turn are used to optimize the model. To verify the effectiveness of our method, the experiments are conducted on three benchmarks, *i.e.*, RefCOCO/+/g. Our method achieves SOTA performance compared with other mask-free RIS methods and even outperforms many fully supervised RIS methods. Specifically, GTMS attains 66.54%, 69.98% and 63.41% IoU on RefCOCO Val-Test, TestA and TestB. Our code will be available at https://github.com/eternalld/GTMS

Keywords: Referring Image Segmentation · Weakly supervision

1 Introduction

Given a natural language expression, referring image segmentation (RIS) aims to localize the described object in an image. Despite various challenges, the task has been significantly pushed forward in recent years [20, 24, 36, 39] as it has large potential in human-robot interaction and autonomous driving. However, the fully supervised methods require training on mask annotations, where pixel-level labeling is time-consuming and laborious.

To tackle this problem, mask-free training methods for fundamental segmentation tasks are recently explored, which employ image-level label [40] or

* Corresponding author

Fig. 1: Our method mines structural and semantic information to construct high quality pseudo labels.

bounding box [8, 19, 33] as training annotations. The bounding box offers more information about the location of the target than other labels, which allows the model to focus more on semantic learning. For example, BBTP [8] integrates multiple instance learning (MIL) into instance segmentation and refines the instance masks by using DenseCRF [17]. BoxInst [33] proposes projection and pairwise affinity loss to replace the original pixel-level mask loss in mask-free instance segmentation. BBAM [19] proposes a bounding box attribution map that can utilize the rich semantics learned by the target detector to generate pseudo labels for training semantic segmentation and instance segmentation networks.

However, the above methods cannot be directly applied to RIS, because these methods only exploit the image information and unable to leverage the unique natural language information inherent to the RIS task. Consequently, these methods fail to achieve optimal results in mask-free RIS. For instance, [6] proposes a method that utilizes the bounding box as a supervised signal, designs adversarial boundary loss to capture the contour of the foreground and proposed a Co-T training strategy to purify pseudo labels. Although [6] effectively delineates object outlines, its training strategy cannot fully extract the semantic information of natural language but only uses pixel-level algorithms. Therefore, exploiting unique textual information is crucial for RIS tasks due to the abundant semantic information inherent in natural language. For example, [18] models word relationships via intra-chunk and inter-chunk consistency, leveraging semantic information to achieve superior results compared to [6]. However, [18] is primarily tailored for semi-supervised learning and does not fully exploit the supervisory signal provided by the bounding box.

Different from [6, 18], our method explicitly extracts semantic information and effectively utilizes the structural information provided by the bounding box to generate high-quality pseudo labels, thereby guiding the model to precisely focus on the target. In this paper, we propose a Gradient-driven Tree-guided Mask-free RIS (GTMS) method to generate high-quality pseudo labels utilizing structural information and semantic information. We extract the structural information of the image, which is achieved by modeling the input image into a tree with structure-preserving property [30].As shown in Figure 1, a structural forest

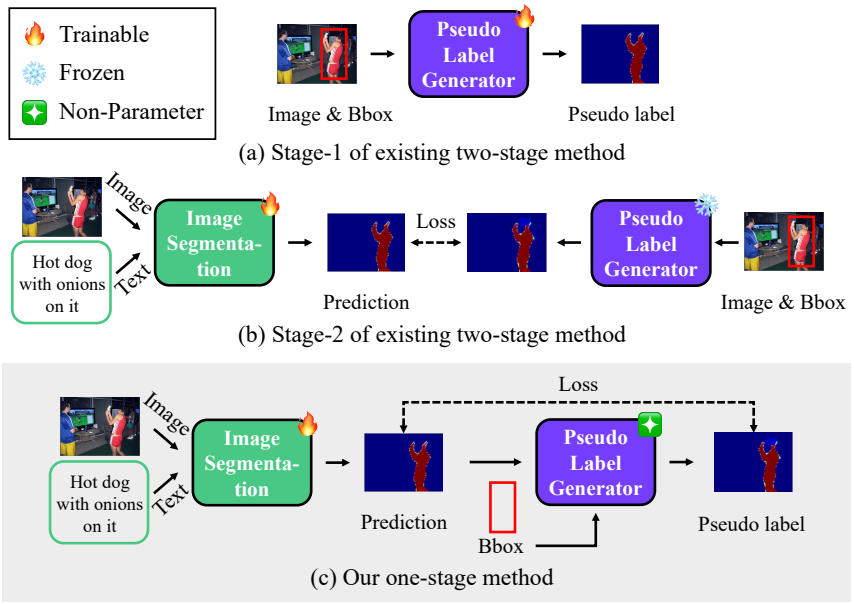

Fig. 2: Current mask-free approaches to RIS. (a) Stage-1 of the two-stage approach. (b) Stage-2 of the two-stage approach. (c) Our one-stage approach.

is constructed by decoupling the tree into a set of sub-trees to effectively use the bounding box supervision signal. Furthermore, we extract semantic information that utilizes GradCAM. Since GTMS is based on CRIS [36], the rich semantic information contained in such knowledge is extracted by GradCAM [28]. We use the GradCAM mechanism to capture the related regions of the image relative to the text, which constitutes an attention map. GTMS assigns a weight for each sub-tree according to the attention map and propagates the weight to all vertices within the sub-tree, producing the vertex weights. Finally, we fuse structural information and semantic information with the original model prediction to generate the high-quality pseudo label. Additionally, to make full use of the location prior information provided by bounding box, we adopt the projection of the bounding box to coarsely localize the target at the early stage of training.

The existing method [6] is a two-stage approach, which first trains a pseudo label generator with the image and Bbox, as shown in Figure 2(a). Then, it freezes the parameters of the pseudo label generator and uses it as a guidance of the RIS model training (Figure 2(b)). In contrast, as depicted in Figure 2(c), our method performs a one-stage manner, which refines the model prediction to generate the pseudo label online. This allows GTMS to utilize the semantic information from the input text. Meanwhile, the pseudo label generator in GTMS is parameter-free, which eliminates the need for a time-consuming training process. Notably, our method is very flexible and not limited to any specific backbone architecture as it does not modify the model architecture.

We evaluate the Gradient-driven Tree-guided Mask-free Referring Image Segmentation (GTMS) method extensively on three RIS benchmarks, known as RefCOCO, RefCOCO+ and RefCOCO-G. GTMS achieves competitive RIS performance without using any mask annotation on all datasets.

Our main contributions are as follows:

- We propose a one-stage mask-free RIS framework that integrates structural information and semantic information to generate high-quality pseudo labels.
- We introduce a structural forest to make full use of the box supervision signal and alleviate the excessive concentration problem of the attention map.
- Our method extracts the semantic information to model the foreground and background information of the subtrees. It attains SOTA performance in the mask-free RIS task on RefCOCO, RefCOCO+, and RefCOCO-G.

2 Related Work

Referring Image Segmentation. Referring Image Segmentation (RIS) aims to segment the target or object specified by the referring expression in the image by understanding the given linguistic expression, which was first introduced by [9]. Compared with traditional image segmentation tasks, RIS is more challenging because it needs to realize the mutual matching between various types of linguistic and visual information, such as entities, attributes, and relationships. [2, 4, 10, 11, 20, 22, 36–38] utilize attention mechanisms to achieve multimodal language-visual integration. Cross-Modal Self-Attention (CMSA) [37] captures long-range dependencies between linguistic and visual contexts to produce a robust multimodal feature representation. BRINet [10] proposes BCAM and GBFM which utilize both visual and linguistic guidances to capture the dependencies between multi-modal features and flexibly incorporate the multilevel cross-modal features. CMPC [11] utilizes different types of words in expressions to segment the referred objects in a graph-based structure, constructing a fully connected graph where each vertex corresponds to an image region and the features of each vertex contain multimodal information about the entity. Recently, VLT [4] builds a transformer-based [35] encoder-decoder architecture that utilizes attention mechanisms to enhance global contextual information. MaIL [22] proposes a unified Mask-Image-Language trimodal framework that utilizes mask information to enhance the interaction of multimodal features. CRIS [36] proposes a framework, which transfers the knowledge of the CLIP model. RefTR [20] proposes a novel transformer decoder that makes use of contextual phrase queries and decodes directly, based on contextual image embedding, into the corresponding image regions and segments. Our proposed method based on CRIS [36] only relies on bounding box annotations for training.

Box Level Mask-free Segmentation. Recently, [3, 8, 14, 29] have employed box level annotation to train semantic segmentation or instance segmentation tasks. For example, BoxSup [3] uses MCG to obtain a rough foreground region of the object, which is used as supervisory information to update the parameters of the network in combination with the fully convolutional network frame-

work in semantic segmentation. BBTP [8] proposes a model that integrates MIL into a fully supervised instance segmentation network, with the core ideas of "Unary term" and "Pairwise term", where the former evaluates each foreground and background regions, and the latter makes the evaluated target mask into a whole. BoxInst's [33] core idea is to replace the original pixel-level mask loss with the proposed projection and pairwise affinity loss, which is significantly improve performance on COCO and Pascal VOC [5] in instance segmentation. BBAM [19] can utilize the rich semantics learned by the target detector to generate pseudo labels for training semantic segmentation and instance segmentation networks. More recently, Box Levelset [21] proposes to learn a series of level sets in an end-to-end manner by iterating through functions based on continuous ChanVese energy in instance segmentation. By minimizing fully differentiable energy functions, the level set of each instance is iteratively optimized within its corresponding bounding box annotation. [18] employs RIS's distinctive textual input for computing semantic information, yet it incompletely leverages the information from bounding box. [6] proposes adversarial boundary loss to obtain foreground contours in the context of training based on bounding box annotations while introducing a Co-T strategy to cleanse pseudo labels in RIS. In contrast to the two-stage method [6], which solely depends on image information in its training pipeline, our proposed one-stage method effectively leverages both textual and image structural information inherent to the RIS.

Tree Filter and GradCAM. Consider an image as an undirected graph, where the vertices are all the pixels in the image and the relationships between pixels form the edges of the undirected graph. Since the pixels of the same object have some similarity, removing the edges with high weight in the graph is equivalent to removing the dissimilar edges and capturing the object boundary information. Due to the properties of tree to maintain structure, it is widely used in vision tasks such as abstraction [16], image smoothing [1], denoising [32], and salient object detection [34]. Besides, LTF [31] proposed a learnable tree filter to capture long-range dependencies for semantic segmentation.

GradCAM (Gradient-weighted Class Activation Mapping) is an important tool for explaining deep learning models, especially in image classification tasks. It can help us understand the decision-making process of the model. The core idea of GradCAM is to determine which areas are most important for the neural network's classification decision by calculating the gradient of the neural network output to the input image. The visualization results generated by GradCAM can help us understand the decision-making path of the network in the classification task, thereby better locating the problems of the model and guiding further optimization and improvement. In the detection task, GradCAM can help us determine the location of the detection frame and improve the accuracy of detection. In the segmentation task, GradCAM can help us locate which areas have the greatest impact on the results predicted by the model, thereby helping us perform better segmentation of these areas, thus improving the overall accuracy and robustness of the model.

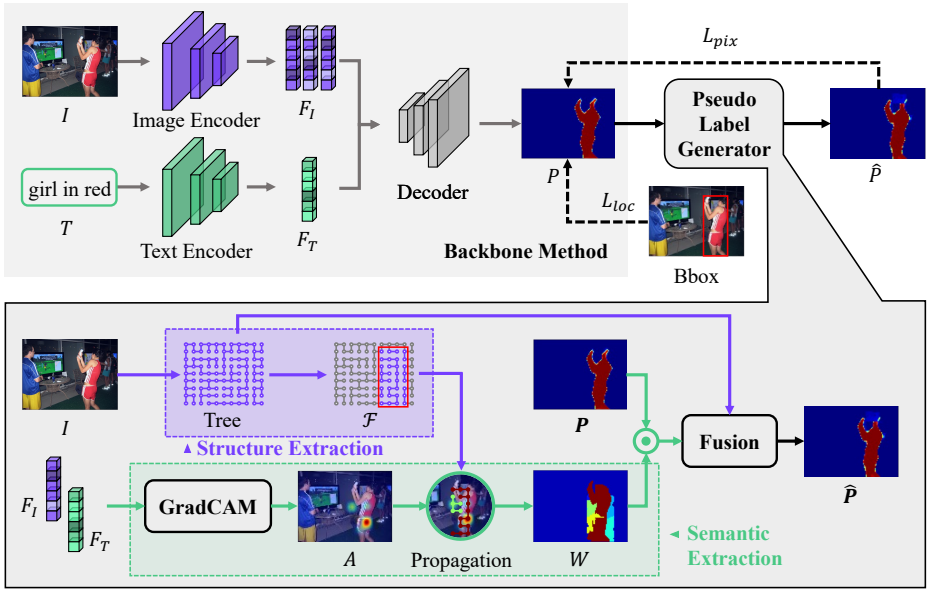

Fig. 3: Overview of GTMS. The I and T are the input image and text respectively, while F_I and F_T are the corresponding high-level feature. The A, $\mathcal{F}$ and W denote the attention map, structural forest and vertex weights, respectively. The P and $\hat{P}$ are the model prediction and pseudo label, respectively.

3 Method

The overall architecture of GTMS is shown in Figure 3, where GTMS enhances the prediction of the backbone method [36] to construct the pseudo label for model training. In the pseudo label generator, an image I and its corresponding high-level features F_I and F_T extracted by the image and text encoders of the backbone method are served as input. The structural property [30] of image I is preserved by the tree structure. Then, the tree is converted into a structural forest by the bounding box to distinguish individual objects. (Section 3.1) Meanwhile, the high-level features F_I and F_T are used to get the attention map through GradCAM, which is denoted as A. With the help of the structural forest, we propagate the attention score to all vertices belonging to the same object. (Section 3.2) Finally, we use the edge weights and vertex weights to enhance the model output P to generate pseudo labels $\hat{P}$. (Section 3.3) To allow the model prediction to fill the bounding box while avoiding introducing irrelevant pixels, we propose two losses L_{loc} and L_{pix} for adversarial training. (Section 3.4)

3.1 Structural Information Extraction

Tree Structure Construction. We adopt tree filter [23, 30] as a structural information extractor. Compared to other filters, the tree filter brings detailed

object structure as well as long-range dependencies. Now, we introduce the tree structure generation. Suppose that an image is represented as an undirected graph $G = (V, E)$, where the vertex set V consists of all pixels and the edges between two vertices form the edge set E. As shown in Figure 3, the structure of a graph with 4-neighbor pixels is employed. Let vertices i and j be adjacent to each other on the graph, and the distance between them can be represented by Euclidean distance:

$$d_{i,j} = d_{j,i} = |I(i) - I(j)|^2, \tag{1}$$

where $I(i) \in \mathbb{R}^{\frac{H}{4} \times \frac{W}{4} \times 3}$, H and W are the height and width of the input image. A minimum spanning tree is constructed by sequentially removing the maximum edge weights in the graph, which means removing dissimilarity between pixels.

Similarly, let vertices m and n be any two vertices on the undirected graph, the distance between m and n is determined by the sum of the distances between two adjacent vertices on the minimum spanning tree, which is defined as D:

$$D_{m,n} = D_{n,m} = \sum_{(i,j) \in \mathbb{E}_{m,n}} d_{i,j}, \tag{2}$$

where $\mathbb{E}$ contains a set of vertices in G traced from vertex i to j, called hyper-edge. Let the distance D between any two vertices map to $(0, 1) \in \mathbb{R}$, which is the edge weight between vertices, defined as $w^{edge}_{m,n}$:

$$w^{edge}_{m,n} = \exp(\frac{-D_{m,n}}{\sigma}), \tag{3}$$

where σ is a hyper-parameter that regulates the color information of the data.

Structural Forest Planting. As shown in Figure 3, the bounding box can decouple the tree into several subtrees. We observe that the different objects are more likely to be distributed in different subtrees. Furthermore, the pixels belonging to the same object often have continuous texture changes and thus the vertices are located in the same sub-tree. The subtrees form a structural forest, which well describes the structural information of different objects within the bounding box. The structural forest guides the propagation of the semantic information, which will be detailed in Section 3.2. As shown in Line 1-6 (Algorithm 1), we cluster the vertices within the same sub-tree with the Union-find algorithm. Specifically, for each edge (u, v) in the tree, if both vertices are within the bounding box, then u and v belong to the same sub-tree.

3.2 Semantic Information Extraction

Since the CLIP model is pre-trained with contrastive learning, the text features and image features are naturally semantically related. In our framework, we utilize GradCAM [28] to get the attention map from text features and image features. However, we found that using the attention map directly would cause excessive concentration on a small region of the entire object. The structural

Algorithm 1: Structural forest planting and attention propagation

Input: $Tree$, $Bbox$, attention map A
Output: Vertex weights W

```
// 1.Cluster vertices belonging to the same sub-tree in the forest
L ← InitParentMatrix() ;          // 2D matrix to record parent vertices
foreach (u, v) ∈ Tree do      // u and v are vertices, (u, v) is an edge
    if u ⊆ Bbox and v ⊆ Bbox then
        l_u ← FindParent(u, L);
        l_v ← FindParent(j, L);
        L[l_u] ← l_v ;                          // Set the parent of l_u as l_v
// 2.Set the highest attention score as the weight of the sub-tree
D ← Dict() ;                        // Record the weight of each sub-tree
foreach u ⊆ Bbox do
    root ← FindParent(u, L) ;                // Root node of the sub-tree
    w ← A[u] ;                    // Get the score from the attention map
    if root ∈ D.keys() then
        if w > D[root] then
            D[root] ← w;
    else
        D[root] ← w
// 3.Propagate the weight to all vertices within the sub-tree
W ← ZerosLike(P) ;                  // Record the weight of each vertex
foreach u ⊆ Bbox do
    root ← FindParent(u, L);
    W[j] ← D[root];
Return W
```

forest is utilized to alleviate the problem. Since the vertices belonging to the same sub-tree are more likely to belong to the same object, we assign the same weight value, *i.e.*, the largest attention score contained within the sub-tree, to all vertices within the sub-tree. Specifically, as shown in Algorithm 1 Line 7-15, the largest attention score for each sub-tree is acquired. In Line 16-19, the weight of each sub-tree is propagated to all its vertices. Additionally, the weight value of the vertices beyond the bounding box is set as 0.

3.3 Pseudo Label Generation

The pseudo label generation directly affects the final model output, it is one of the most important aspects of weakly supervised learning, and our goal is to generate high-quality pseudo labels. As shown in the Figure 3, the model prediction P is first weighted by the vertex weights, which contain the semantic information corresponding to the referring sentence. Then, the weighted prediction is further enhanced by the tree structure to produce the pseudo label $\hat{P}$.

The whole generation process is shown as Equation (4):

$$\begin{aligned} P^{'} &= P \odot W, \quad \theta = \sum_{\forall j \in V} w_{i,j}^{edge}, \\ \hat{P}(i) &= \frac{1}{\theta} \sum_{\forall j \in V} w_{i,j}^{edge} P^{'}(j), \end{aligned} \tag{4}$$

where θ represents the sum of edge weights $w_{i,j}^{edge}$ alone with j which used for normalization, V is the set of all vertices, i,j are vertices indexes.

3.4 Loss Function

We finally introduce the loss functions adopted in our framework, including location loss which guides the model to fill bounding box and pixel loss which exploits pseudo labels for fine-grained segmentation to avoid introducing irrelevant pixels.

Followed boxinst [33], the bounding box is an external rectangle of mask annotation, thus the enclosing rectangle of the mask should be wrapped inside the bounding box. The height and width projection of the mask's enclosing rectangle should be approximately the same as the projection of the bounding box.

Let the one-dimensional projection of the bounding box on the height and width be recorded as (B_w, B_h), and the one-dimensional projection of the mask on the height and width as (P_w, P_h). By minimizing the dissimilarity between the model output projection (P_w, P_h) and bounding box projection (B_w, B_h), the model can focus on mining the foreground region and reduce the negative effect of foreground and background imbalance. The projection operation can be expressed as Equation (5):

$$\begin{aligned} B_* &= \text{Proj}(b_*) = \max(b_*) \\ P_* &= \text{Proj}(p_*) = \max(p_*) \end{aligned} \tag{5}$$

where $*$ denotes height h and width w, $\text{Proj}(\cdot)$ represents for the projection operation, b_* and p_* stand for slice along height or width direction. The location loss is defined as follows:

$$L_{loc} = \text{Dice}(B_w, P_w) + \text{Dice}(B_h, P_h), \tag{6}$$

where Dice represents Dice Loss [25].

After the pseudo label $\hat{P}$ is obtained, the pixel loss L_{pix} is designed as follows:

$$L_{pix} = \frac{1}{\Omega_B} \sum_{\forall i \in \Omega_B} |P_i - \hat{P}_i|. \tag{7}$$

where Ω_B denotes the area of the bounding box. The final loss function is a linear combination of the two loss functions, which is shown below:

$$L = L_{loc} + \gamma L_{pix}. \tag{8}$$

where γ is the control factor to balance the two constraints. Considering that the goal in the early stage of training is to constrain the prediction to be inside the bounding box, and in the later stage, it will focus more on the precision of target segmentation, the value of γ will be adjusted during the training process.

4 Experiment

4.1 Setting

Datasets and Metrics. RefCOCO [13] dataset contains 142,210 language expressions with 50,000 objects in 19,994 images. Every image contains two or more objects and each expression has an average length of 3.6 words. The dataset is split into train, validation, testA, and testB with 120,624, 10,834, 5,657, and 5,095 samples.

The RefCOCO+ [13] dataset contains 141,564 language expressions and 49,856 objects in 19,992 images. Compared to the RefCOCO dataset, the RefCOCO+ dataset excludes some types of absolute positional words, which could be more challenging than the RefCOCO dataset. The dataset is split into 120,624 training samples, 10,758 validation samples, 5,726 test A samples and 4,889 test B samples respectively.

RefCOCO-Google [26] dataset contains 104,560 language expressions and 54,822 objects in 26,711 images. The average length of the sentences is 8.4 words, with more words about location and appearance. Compared to the RefCOCO and RefCOCO+, natural expressions in G-Ref are collected from Amazon Mechanical Turk instead of a two-player game. We adopt UNC partition.

Following the setup of previous work [4, 20, 36], we evaluate the proposed method using Intersection-over-Union (IoU) and Precision (Prec@X). The IoU metric represents the ratio between the total intersection region and the total joint region of the predicted segmentation mask and ground truth for all test images. Prec@X measures the percentage of the predicted mask's IoU score above a threshold X, where X $\in \{0.5, 0.6, 0.7, 0.8, 0.9\}$.

Implementation. We adopt ResNet [7] as the image encoder and Transformer [35] as the text encoder, which is initialized with the CLIP [27] pretraining model. The input resolution is 416×416 for RefCOCO [13], RefCOCO+ [13] and RefCOCO-Google [26]. The Adam [15] optimizer and MultiStepLR are employed. The initialized learning rate is 0.00001 and the learning rate decay factor is 0.1. The network is trained for 50 epochs. We preheat the pix loss in a linear growth manner, with the factor γ reaching a maximum at 10 epochs. As for σ in Equation (3), we set $\sigma = 0.02$ in all experiments. Experiments are conducted on Pytorch with 2 NVIDIA RTX 3090 GPUs.

For an input of image $I \in \mathbb{R}^{H \times W \times 3}$, visual features $F_I \in \mathbb{R}^{\frac{H}{32} \times \frac{W}{32} \times C}$ are generated from the ResNet layer 4. For an input of referring expression $T \in \mathbb{R}^L$, the text features $F_T \in \mathbb{R}^L$ are generated from the Transformer. Note that C represents the number of channels and L represents the sentence length in the experiment. Text features F_T and visual features F_I are fused by GradCAM to

Table 1: Experimental results of the fully and weakly supervised RIS methods. "RN", "DL" and "DN" represent ResNet, DeepLab and DarkNet, respectively; "Fully" means the supervised signal is mask, "Box" means the supervised signal is bounding box; "*"represents the CRF utilized as a post-processing method.

Method	Backbone	Sup.	RefCOCO			RefCOCO+			RefCOCO-G	
			Val	TestA	TestB	Val	TestA	TestB	Val	Test
MattNet [38]	RN101	Fully	56.51	62.37	51.70	46.67	52.39	40.08	47.64	48.61
STEP [2]	RN101	Fully	60.04	63.46	57.97	48.19	52.33	40.41	46.40	-
CMSA* [37]	DL	Fully	58.32	60.61	55.09	43.76	47.60	37.89	39.98	-
BRINet [10]	RN101	Fully	61.35	63.37	59.57	48.57	52.87	42.13	48.04	-
CMPC* [11]	RN101	Fully	61.36	64.53	59.64	49.56	53.44	43.23	-	-
SHNet [12]	DLv3+	Fully	65.32	68.56	62.04	52.75	58.46	44.12	49.90	-
VLT [4]	DN53	Fully	65.65	68.29	62.73	55.50	59.20	49.36	52.99	56.65
MaIL [22]	RN50	Fully	70.13	71.71	66.92	62.23	65.92	56.06	62.45	62.87
CRIS [36]	RN101	Fully	70.47	73.18	66.10	62.27	68.08	53.68	59.87	60.36
RefTR [20]	RN101	Fully	70.56	73.49	66.57	61.08	64.69	52.73	58.73	58.51
Two-stage* [6]	RN101	Bbox	58.83	61.31	56.01	47.85	51.54	40.73	46.42	-
Chunk [18]	RN101	Bbox	58.12	61.23	55.47	48.19	53.01	42.83	49.64	-
Ours	RN101	Bbox	**66.54**	**69.98**	**63.41**	**57.59**	**63.46**	**50.32**	**54.52**	**54.75**

generate attention map $A \in \mathbb{R}$. After the Sigmoid and upsampling operation, P becomes the final network prediction $P \in [0,1]^{H \times W}$, where 0 and 1 represent the background and foreground. During the training process, we tend to make the weight of location loss larger in the early stage because it needs to constrain the predicted location of the model.

4.2 Main Results

Our proposed Gradient-driven Tree-guided Mask-free Referring Image Segmentation(RIS) method achieves competitive results, as shown in Table 1, we compare it with the popular fully supervised RIS approaches [11, 20, 36, 37] which utiliz attention mechanism and weakly supervised method [6], which proposes a two-stage mask-free RIS method based on BRINet [10]. We conduct experiments on three datasets, including RefCOCO, RefCOCO+, RefCOCO-G. Compared with [6] which uses CRF [17] as post-processing during testing and introducing additional trainable parameters during training, our method achieves SOTA results without using any post-processing and introducing trainable parameters. Specifically, we achieve 66.54%, 69.98%, and 63.41% on Val, TestA, and TestB. Compared with fully supervised methods, we also achieve remarkable results.

As shown in Figure 4, utilizing structural information and semantic information at the same time, our method can find the target for the input text that does not contain position hint words. For example, text "Bed next to the desk", the model first needs to understand which objects in the image are "desk" and which objects are "bed", and then find the target referred to by the text.

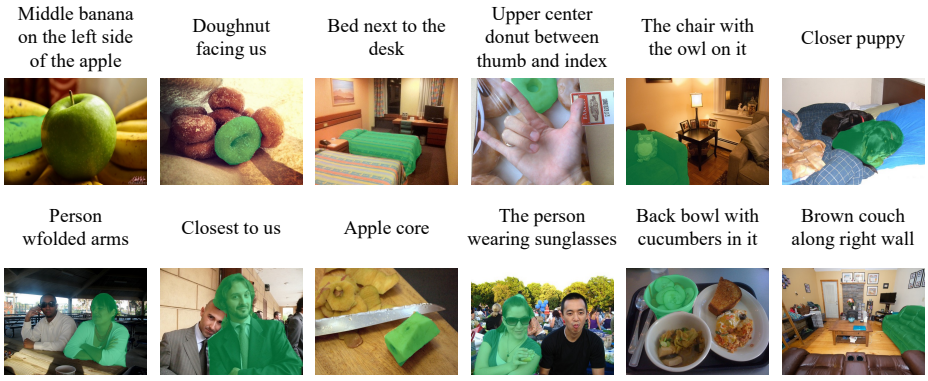

Fig. 4: Visual quality for the GTMS.

Table 2: Ablation study. BCE denotes binary cross-entropy loss. Experiments are implemented on the RefCOCO dataset, using ResNet50 as backbone.

Signal	Architecture	Loss	Val	TestA	TestB
Mask	-	BCE	69.52	72.72	64.70
Bbox	-	BCE	46.11	45.70	47.09
Bbox	-	L_{loc}	58.62	61.17	55.53
Bbox	Tree	$L_{loc} + \lambda L_{pix}$	61.72	65.14	57.94
Bbox	Forest+GradCAM	$L_{loc} + \lambda L_{pix}$	**64.33**	**67.59**	**60.39**

We have also achieved pleasing segment results for images where the foreground and background are similar and no location information is provided, such as "Doughnut facing us". For complex long sentences, such as "Middle banana on the left side of the apple", and "Upper center donut between thumb and index", we also achieved remarkable results. For sentences that require understanding image depth, such as "Closest to us" and "Closer puppy", this proves that our method can also understand image depth information.

4.3 Ablation Study

To verify the effectiveness of our proposed method, we conduct ablation studies on the RefCOCO [13] dataset. We analyze the impact of the proposed components, then investigate the effect of different hyper-parameters.

As shown in Table 2, binary cross entropy (BCE) loss is utilized as the loss function, with 'mask' and 'Bbox' employed as supervision signals respectively. As expected, directly using BCE loss makes the model output tend to be consistent with the given supervision signal. Therefore, when bounding box is used as the supervision signal, the effect is very poor. Then, we use location loss instead of BCE loss, and the effect is improved by 12.51%, which takes advantage of the prior information that the one-dimensional projections of Bbox and target in the height and width directions are similar. Furthermore, we add the Tree structure

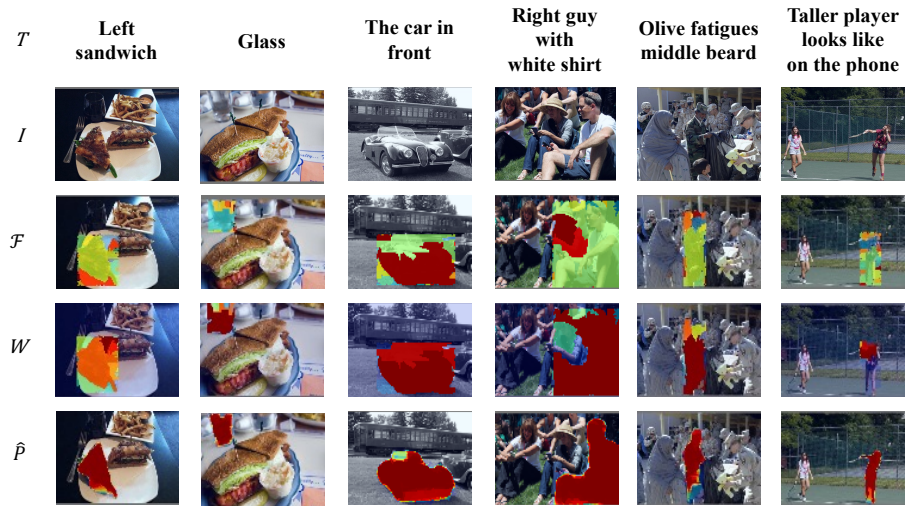

Fig. 5: Visualization of intermediate results. T and I represent text and image. $\mathcal{F}$ denotes the structural forest, where the colors are randomly selected to distinguish different sub-trees. W represents the vertex weights. $\hat{P}$ indicates the pseudo label.

containing low-level structural information of the image on the basis of location loss, and the effect was improved by 3.1%, which shows that our tree structural information promotes pseudo label generation. Finally, we use GradCAM to extract high-level semantic information and jointly use high-level information and low-level information to guide high-quality pseudo label generation, improving the effect by 2.61%.

Specifically, we visualize the various inputs of the fusion module as shown in Figure 5, including the structural forest, vertex weight and pseudo label. We selected cases that cover a variety of categories, including position words such as “left, right”, color words such as “blue, white, olive”, actions such as “sitting”, and depth words such as “front”, adjectives such as "taller, fatigues", and sentences of varying lengths.

As shown in the second row of Figure 5 which indicates the structural forest $\mathcal{F}$. The foreground and background areas are divided into different sub-trees, which shows that our method uses the tree structure to effectively capture the structural information in the image, and uses the bounding box to achieve the distinction of different object structures. For example, we can see multiple colors of sandwich, which is because the main body and background of the sandwich inside bounding box are divided into different sub-tree areas. As shown in the third row of Figure 5, which indicates the vertex weights W. For the target of interest, the vertex has a higher weight, which shows that our method can well capture the key information in the picture. Our method effectively utilizes the structural information provided by low-level features and the rich semantic information provided by high-level features. For example, for the input text

"Left Sandwich", the sandwich on the left is considered the target of interest. The relevant area of its Vertex Weight is obviously higher, while the weight value of the background area such as the plate is very low, which shows that the model successfully captures the semantics of "Sandwich".

CLIP is not the dominant factor in performance improvement. To prove that the effectiveness of GTMS does not only come from the use of the CLIP large model, we implemented the two-stage method [6] and modified the backbone model to CRIS [36], which is the same as GTMS. As shown in Table 3, although Two-stage (CRIS) has an average 7.64% improvement compared to Two-stage [6], there is still a 10.8% gap compared with ours, which illustrates the effectiveness of the proposed method.

Table 3: Performance comparison between Two-stage and GTMS using the same backbone. Two-stage(CRIS): two-stage method using CRIS as the backbone.

Method	RefCOCO			RefCOCO+			RefCOCO-G	
	Val	TestA	TestB	Val	TestA	TestB	Val	Test
Two-stage [6]	58.01	60.52	55.48	47.12	50.86	40.26	46.03	-
Two-stage (CRIS)	62.38	64.07	58.98	50.76	55.83	43.86	49.42	-
Ours	**66.54**	**69.98**	**63.41**	**57.59**	**63.46**	**50.32**	**54.52**	**54.75**

Efficiency of GTMS. Since GTMS does not introduce additional modules into the backbone method, it shares the same inference time as the backbone. In our default implementation and hardware, GTMS can process 17 images per second. It is worth noting that the efficiency of the two-stage method [6] also fully depends on its backbone method. When replacing its backbone method with CRIS [36], it shares the same inference speed with GTMS. However, the speed dropped to 9 fps when it introduces DenseCRF post-processing to achieve its optimal segmentation result.

5 Conclusion

In this work, we propose a gradient-driven tree-guided mask-free method, GTMS, for box level mask-free referring image segmentation. GTMS leverages both the location information from the bounding box and the structural information derived from the structural forest. Additionally, it employs semantic attention utilizing gradient information. Both the structural and semantic information are simultaneously fused with model output to obtain high-quality pseudo labels, which in turn are used to optimize the performance of the segmentation model. We conduct experiments to verify the effectiveness of the method on three different datasets, RefCOCO, RefCOCO+, and RefCOCO-G. Our method achieves SOTA performance on mask-free RIS methods and even outperforms many fully supervised RIS methods.

References

1. Bao, L., Song, Y., Yang, Q., Yuan, H., Wang, G.: Tree filtering: Efficient structure-preserving smoothing with a minimum spanning tree. IEEE Transactions on Image Processing **23**(2), 555–569 (2013)
2. Chen, D.J., Jia, S., Lo, Y.C., Chen, H.T., Liu, T.L.: See-through-text grouping for referring image segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 7454–7463 (2019)
3. Dai, J., He, K., Sun, J.: Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In: Proceedings of the IEEE international conference on computer vision. pp. 1635–1643 (2015)
4. Ding, H., Liu, C., Wang, S., Jiang, X.: Vision-language transformer and query generation for referring segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 16321–16330 (2021)
5. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. International journal of computer vision **88**, 303–338 (2010)
6. Feng, G., Zhang, L., Hu, Z., Lu, H.: Learning from box annotations for referring image segmentation. IEEE Transactions on Neural Networks and Learning Systems (2022)
7. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
8. Hsu, C.C., Hsu, K.J., Tsai, C.C., Lin, Y.Y., Chuang, Y.Y.: Weakly supervised instance segmentation using the bounding box tightness prior. Advances in Neural Information Processing Systems **32** (2019)
9. Hu, R., Rohrbach, M., Darrell, T.: Segmentation from natural language expressions. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14. pp. 108–124. Springer (2016)
10. Hu, Z., Feng, G., Sun, J., Zhang, L., Lu, H.: Bi-directional relationship inferring network for referring image segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4424–4433 (2020)
11. Huang, S., Hui, T., Liu, S., Li, G., Wei, Y., Han, J., Liu, L., Li, B.: Referring image segmentation via cross-modal progressive comprehension. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10488–10497 (2020)
12. Jain, K., Gandhi, V.: Comprehensive multi-modal interactions for referring image segmentation. arXiv preprint arXiv:2104.10412 (2021)
13. Kazemzadeh, S., Ordonez, V., Matten, M., Berg, T.: Referitgame: Referring to objects in photographs of natural scenes. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). pp. 787–798 (2014)
14. Kervadec, H., Dolz, J., Wang, S., Granger, E., Ayed, I.B.: Bounding boxes for weakly supervised segmentation: Global constraints get close to full supervision. In: Medical imaging with deep learning. pp. 365–381. PMLR (2020)
15. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
16. Koga, T., Suetake, N.: Structural-context-preserving image abstraction by using space-filling curve based on minimum spanning tree. In: 2011 18th IEEE International Conference on Image Processing. pp. 1465–1468. IEEE (2011)

17. Krähenbühl, P., Koltun, V.: Efficient inference in fully connected crfs with gaussian edge potentials. Neural Information Processing Systems (2011)
18. Lee, J., Lee, S., Nam, J., Yu, S., Do, J., Taghavi, T.: Weakly supervised referring image segmentation with intra-chunk and inter-chunk consistency. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 21870–21881 (October 2023)
19. Lee, J., Yi, J., Shin, C., Yoon, S.: Bbam: Bounding box attribution map for weakly supervised semantic and instance segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2643–2652 (2021)
20. Li, M., Sigal, L.: Referring transformer: A one-step approach to multi-task visual grounding. Advances in neural information processing systems **34**, 19652–19664 (2021)
21. Li, W., Liu, W., Zhu, J., Cui, M., Hua, X.S., Zhang, L.: Box-supervised instance segmentation with level set evolution. In: Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXIX. pp. 1–18. Springer (2022)
22. Li, Z., Wang, M., Mei, J., Liu, Y.: Mail: A unified mask-image-language trimodal network for referring image segmentation. arXiv preprint arXiv:2111.10747 (2021)
23. Liang, Z., Wang, T., Zhang, X., Sun, J., Shen, J.: Tree energy loss: Towards sparsely annotated semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16907–16916 (2022)
24. Liu, J., Ding, H., Cai, Z., Zhang, Y., Satzoda, R.K., Mahadevan, V., Manmatha, R.: Polyformer: Referring image segmentation as sequential polygon generation. arXiv e-prints pp. arXiv–2302 (2023)
25. Milletari, F., Navab, N., Ahmadi, S.A.: V-net: Fully convolutional neural networks for volumetric medical image segmentation. In: 2016 fourth international conference on 3D vision (3DV). pp. 565–571. Ieee (2016)
26. Nagaraja, V.K., Morariu, V.I., Davis, L.S.: Modeling context between objects for referring expression understanding. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14. pp. 792–807. Springer (2016)
27. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PMLR (2021)
28. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: Visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE international conference on computer vision. pp. 618–626 (2017)
29. Song, C., Huang, Y., Ouyang, W., Wang, L.: Box-driven class-wise region masking and filling rate guided loss for weakly supervised semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3136–3145 (2019)
30. Song, L., Li, Y., Li, Z., Yu, G., Sun, H., Sun, J., Zheng, N.: Learnable tree filter for structure-preserving feature transform. Advances in neural information processing systems **32** (2019)
31. Song, L., Li, Y., Li, Z., Yu, G., Sun, H., Sun, J., Zheng, N.: Learnable tree filter for structure-preserving feature transform. Advances in neural information processing systems **32** (2019)

32. Stawiaski, J., Meyer, F.: Minimum spanning tree adaptive image filtering. In: 2009 16th IEEE International Conference on Image Processing (ICIP). pp. 2245–2248. IEEE (2009)
33. Tian, Z., Shen, C., Wang, X., Chen, H.: Boxinst: High-performance instance segmentation with box annotations. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5443–5452 (2021)
34. Tu, W.C., He, S., Yang, Q., Chien, S.Y.: Real-time salient object detection with a minimum spanning tree. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2334–2342 (2016)
35. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)
36. Wang, Z., Lu, Y., Li, Q., Tao, X., Guo, Y., Gong, M., Liu, T.: Cris: Clip-driven referring image segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11686–11695 (2022)
37. Ye, L., Rochan, M., Liu, Z., Wang, Y.: Cross-modal self-attention network for referring image segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10502–10511 (2019)
38. Yu, L., Lin, Z., Shen, X., Yang, J., Lu, X., Bansal, M., Berg, T.L.: Mattnet: Modular attention network for referring expression comprehension. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1307–1315 (2018)
39. Zhao, W., Rao, Y., Liu, Z., Liu, B., Zhou, J., Lu, J.: Unleashing text-to-image diffusion models for visual perception. arXiv preprint arXiv:2303.02153 (2023)
40. Zhou, Y., Zhu, Y., Ye, Q., Qiu, Q., Jiao, J.: Weakly supervised instance segmentation using class peak response. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3791–3800 (2018)