Extending CLIP's Image-Text Alignment to Referring Image Segmentation

Seoyeon Kim[1] **Minguk Kang**[1] **Dongwon Kim**[1] **Jaesik Park**[2] **Suha Kwak**[1]
[1]POSTECH, [2]Seoul National University
[1]{syeonkim07,mgkang,kdwon,suha.kwak}@postech.ac.kr
[2]jaesik.park@snu.ac.kr

Abstract

Referring Image Segmentation (RIS) is a cross-modal task that aims to segment an instance described by a natural language expression. Recent methods leverage large-scale pretrained unimodal models as backbones along with fusion techniques for joint reasoning across modalities. However, the inherent cross-modal nature of RIS raises questions about the effectiveness of unimodal backbones. We propose RISCLIP, a novel framework that effectively leverages the cross-modal nature of CLIP for RIS. Observing CLIP's inherent alignment between image and text features, we capitalize on this starting point and introduce simple but strong modules that enhance unimodal feature extraction and leverage rich alignment knowledge in CLIP's image-text shared-embedding space. RISCLIP exhibits outstanding results on all three major RIS benchmarks and also outperforms previous CLIP-based methods, demonstrating the efficacy of our strategy in extending CLIP's image-text alignment to RIS.

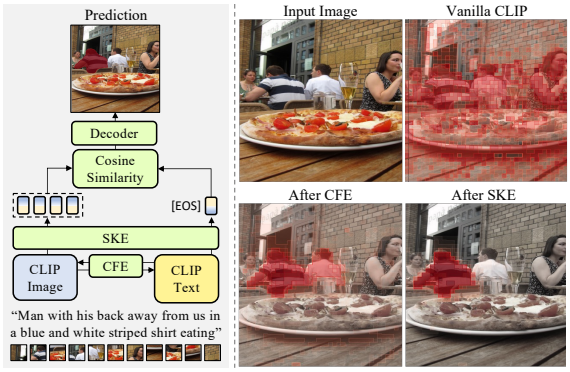

Figure 1: CLIP's image-text alignment produces preliminary patch-level groundings through cosine similarity between patch-level image and sentence-level text features. Building upon this alignment, we refine CLIP's groundings into accurate segmentations with three modules. Cross-modal Feature Extraction (CFE) modules enhance CLIP's unimodal image and text features by aligning them at candidate regions. Shared-space Knowledge Exploitation (SKE) modules leverage the rich alignment knowledge in CLIP's image-text shared-embedding space to discern the target referent. Lastly, a decoder transforms the patch-level grounding into a pixel-wise segmentation.

1 Introduction

Referring Image Segmentation (RIS) is a multi-modal task that aims to produce a pixel-wise mask of an instance referred to by a natural language expression. The task holds great potential with various applications, such as language-based image editing (Chen et al., 2018; Parmar et al., 2023; Brooks et al., 2023) and human-robot interaction (Wang et al., 2019).

RIS poses a formidable challenge, demanding a nuanced understanding of both visual and linguistic modalities. Thus, conventional methods (Li and Sigal, 2021; Wang et al., 2022; Zhu et al., 2022; Yang et al., 2022; Liu et al., 2023b) leverage the profound knowledge learned by large-scale pretrained models, employing image and text encoders as backbones, such as Swin-T (Liu et al., 2021) trained on ImageNet-21K (Ridnik et al., 2021) and BERT (Devlin et al., 2019) trained on Wikipedia and Google's BookCorpus. Additionally, various fusion techniques (Hu et al., 2016; Ye et al., 2019; Ding et al., 2021; Hu et al., 2020; Hui et al., 2020) have been introduced to enable joint reasoning across modalities, significantly advancing RIS.

However, the inherent cross-modal nature of RIS raises questions about the effectiveness of unimodal backbones. In contrast, we posit that the cross-modal nature of CLIP (Radford et al., 2021) makes it a better candidate for RIS. Extending on the findings of MaskCLIP (Zhou et al., 2022), we observe that CLIP possesses image-text alignment beneficial for RIS—cosine similarity maps between patch- and sentence-level features (Zhou et al., 2022) produce preliminary groundings. To fully utilize this image-text alignment, we freeze CLIP

Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 4611–4628

and build upon this promising starting point. While previous work (Wang et al., 2022; Xu et al., 2023b) demonstrates the potential of adopting CLIP for RIS, their performances trail behind the current state of the arts, indicating an opportunity for improvement.

In our framework, *RISCLIP*, we effectively adapt CLIP to RIS, capitalizing on its image-text alignment. Firstly, we enhance the unimodal feature extraction by introducing cross-modal interaction between the image and text encoders with Cross-modal Feature Extraction (CFE) modules. These modules effectively align the image and text features at candidate regions—regions described by or related to the target text. Then, we leverage the rich alignment knowledge captured in CLIP's image-text shared-embedding space by introducing inter- and intra-modal interactions after the feature extraction process with Shared-space Knowledge Exploitation (SKE) modules. The comprehensive interactions allow RISCLIP to discern the target from the candidate regions. Our CFE and SKE modules effectively adapt CLIP to RIS, evolving CLIP's preliminary image-text alignment into accurate groundings, as shown in Fig. 1.

RISCLIP exhibits outstanding performance on all three major RIS benchmarks. Particularly, RISCLIP excels on the more challenging datasets, such as RefCOCOg (Mao et al., 2016) with complicated texts. Such result indicates that adopting a cross-modal backbone like CLIP, which trains on varied captions including extensive expressions, is beneficial for RIS. Furthermore, RISCLIP also surpasses previous CLIP-based methods (Wang et al., 2022; Xu et al., 2023b), proving that such performance arises from both CLIP and our effective adaptation strategy.

2 Related Work

Referring image segmentation. RIS aims at predicting a pixel-wise mask of an object described by a natural language text. The pioneering work (Hu et al., 2016) extracts image and text features with recurrent LSTMs and a CNN and concatenates them along the channel dimension into multi-modal features. Follow-up work expands on this framework by incorporating recurrent multi-modal interactions (Liu et al., 2017) along with more fine-grained segmentation with hierarchical visual features (Li et al., 2018; Margffoy-Tuay et al., 2018; Chen et al., 2019; Jain and Gandhi, 2022). Another line of research focuses on attending to more important words in the referring expression (Yu et al., 2018; Shi et al., 2018; Liu et al., 2022) and proposes effective cross-modal attention modules (Ye et al., 2019; Ding et al., 2021; Hu et al., 2020; Hui et al., 2020). Recent methods adopt pretrained transformer encoders to extract image and text features (Kim et al., 2022; Tang et al., 2023), and others further leverage the encoder transformer layers for multi-modal feature extraction (Feng et al., 2021; Yang et al., 2022; Zhang et al., 2019; OuYang et al., 2023). Moving towards real-world conditions, recent work tackles settings where expressions describe none to multiple objects (Hu et al., 2023; Liu et al., 2023a), image-expression pairs only are provided without segmentation masks (Strudel et al., 2022; Liu et al., 2023a; Kim et al., 2023; Lee et al., 2023), and no labels are provided for training (Yu et al., 2023; Suo et al., 2023).

Contrastive language-image pre-training. CLIP (Radford et al., 2021) is well-known for its general cross-modal capacity. Acquired through extensive contrastive pretraining on large-scale image-text pairs, CLIP carries not only expertise knowledge in both visual and linguistic modalities but also general alignment between image and text features. Various multi-modal tasks, including text-to-image generation (Ramesh et al., 2022; Rombach et al., 2022) and visual captioning (Mokady et al., 2021; Hessel et al., 2021), benefit from CLIP's rich multi-modal alignment. Several works attempt to adapt CLIP to dense prediction tasks, such as open vocabulary object detection (Du et al., 2022; Rasheed et al., 2022) and semantic segmentation (Luo et al., 2022; Xu et al., 2023a). In particular, MaskCLIP (Zhou et al., 2022) exploits the alignment between patch- and sentence-level features for zero-shot open vocabulary segmentation. We hypothesize that such patch-level alignment is a good starting point for RIS, and, consistent with such hypothesis, observe that the alignment produces a noticeable 23.86 mIoU on the most challenging RIS benchmark—RefCOCOg-UMD (Nagaraja et al., 2016)—with a upsampling decoder attached. Thus, we propose a new framework that effectively exploits such informative cross-modal alignment of CLIP to produce accurate RIS predictions.

CLIP for RIS. Although methods that adopt CLIP for RIS exist (Radford et al., 2021), they either fully finetune CLIP and risk losing its general knowledge (Wang et al., 2022) or do not explicitly lever-

age the alignment between the image and text features learned from millions of image-text pairs (Xu et al., 2023b). Above all, their performance falls behind the state of the art, suggesting room for improvement. Thus, we take a new approach of explicitly exploiting CLIP's rich image-text alignment by extracting preliminary grounding maps from frozen CLIP and enhancing them into accurate segmentations with our adaptive modules. Our framework achieves compelling results, showing that we effectively extend CLIP's image-text alignment to RIS.

3 Method

Fig. 2 illustrates the overall pipeline of our method, RISCLIP. We exploit CLIP's cross-modal alignment between patch- and sentence-level features and evolve cosine similarity maps between them into precise pixel-wise groundings with our new framework. The Cross-modal Feature Extraction (CFE) modules enhance the unimodal feature extraction of CLIP with cross-modal communication, aligning image and text features at candidate regions related to the text. The Shared-space Knowledge Exploitation (SKE) modules exploit CLIP's rich knowledge captured within the image-text shared embedding space to discern the target referent from candidate regions, particularly those described by complicated expressions. Together, CFE and SKE modules adapt CLIP to RIS, producing precise patch-level grounding maps. Finally, a simple decoder refines these maps into pixel-level segmentations. The following sections detail each module—CFE, SKE, and the decoder—starting with the original CLIP feature extraction.

3.1 CLIP for referring image segmentation

To fully utilize CLIP's invaluable image-text alignment, we freeze CLIP and introduce modules that enhance the features for RIS. We explain the feature extraction process of CLIP and detail the construction of the patch-level grounding map in the following paragraphs.

Feature extraction. Both CLIP's image and text encoders consist of repeated transformer layers (Vaswani et al., 2017), followed by a layer normalization (LN) (Ba et al., 2016) and a linear projection to a shared image-text embedding space. Each transformer layer has two submodules: a multi-head self-attention (MHSA) and a multilayer perceptron (MLP) with each submodule preceded by LN. Text features are extracted via a text encoder as a sequence of word- and sentence-level representations. Firstly, the sentence is transformed into word embeddings via byte pair encoding (Gage, 1994) and encased with a learnable `[SOS]` and `[EOS]` token. This sequence is then passed through transformer layers and linearly projected to the shared image-text embedding space. The output `[EOS]` token acts as the sentence-level representation of the text, and we denote it $\mathbf{t}_{\text{eos}} \in \mathbb{R}^{1\times d}$, where d is the dimensionality of the shared embedding space. Analogously, image features are extracted with an image encoder as a sequence of patch-level representations prepended with an image-level embedding. Specifically, the image is divided into a sequence of patches, prepended with a learnable `[CLS]` token, passed through transformer layers, and linearly projected to the shared embedding space. We denote the output patch-level features as $\mathbf{V}_{\text{patch}} \in \mathbb{R}^{N_{\text{visual}}\times d}$, where N_{visual} is the number of patches.

Patch-level grounding map. We build upon the findings of MaskCLIP (Zhou et al., 2022) that CLIP possesses cross-modal alignment between patch- and sentence-level features to produce preliminary patch-level RIS groundings. Slightly modifying the image feature extraction process of the last image transformer layer, we adopt the value tokens from the MHSA as patch tokens and pass them through the subsequent LN, MLP, and linear projection to the image-text shared embedding space to produce a new $\mathbf{V}_{\text{patch}}$. Cosine similarity between $\mathbf{V}_{\text{patch}}$ and $\mathbf{t}_{\text{eos}}$ produces preliminary grounding maps for RIS, highlighting regions related to the text. We observe that the patch-level grounding maps with a decoder attached provide a promising mIoU of 23.86 on the RefCOCOg-UMD test set (Nagaraja et al., 2016). Adopting this as a good starting point, we introduce modules to transform the preliminary maps into accurate segmentations.

Adapters. Since the CLIP backbone is trained with contrastive learning between the image- and sentence-level features, its features are suboptimal for dense prediction tasks like RIS. Thus, we introduce adapters to transform the features into representations more appropriate for RIS. We adopt the adapter architecture from (Houlsby et al., 2019; Chen et al., 2022), which consists of a down-projection linear layer, a non-linear activation, and an up-projection linear layer. These structures are residually attached after the MHSA and MLP modules of CLIP's transformer layers. The residual

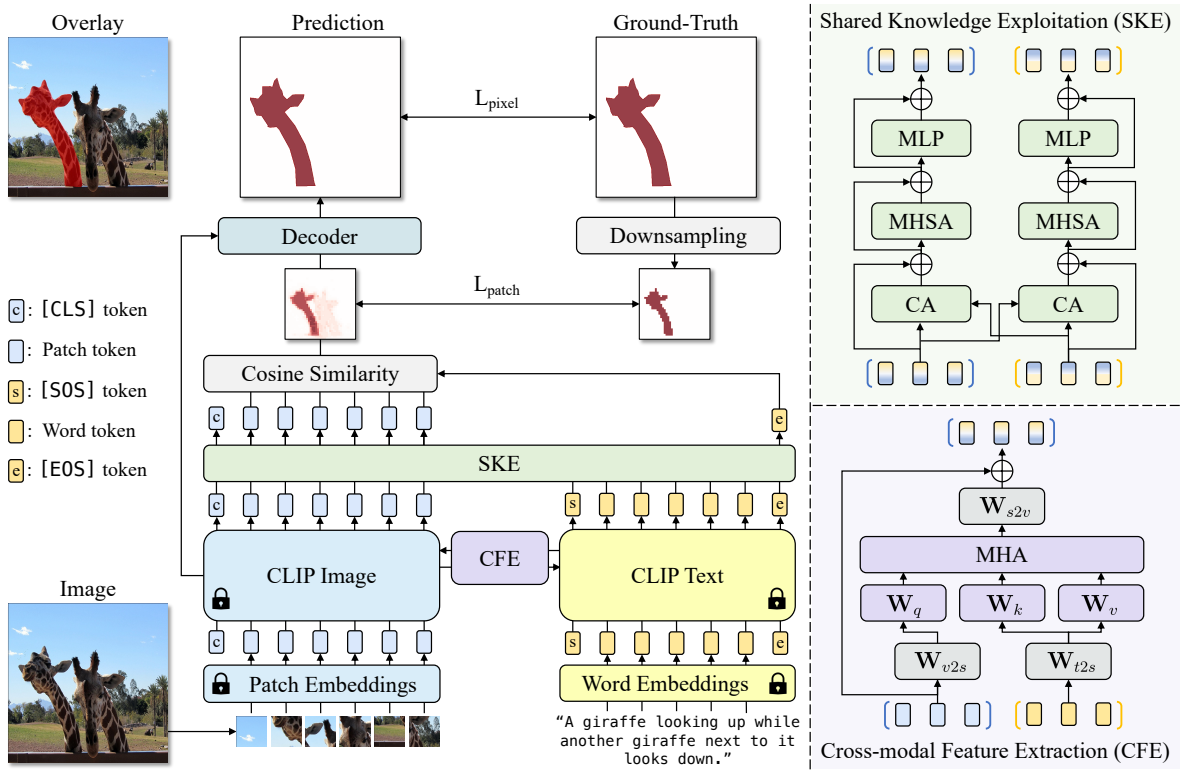

Figure 2: The overall pipeline of RISCLIP. We adopt frozen CLIP image and text encoders as backbones to exploit their aligned image and text features and adapt them to RIS with two modules, CFE and SKE. Firstly, the CFE modules between the encoders enable cross-modal commnuication between the two encoders to align their unimodal features at candidate regions. Secondly, the SKE modules on top of the encoders leverage the rich cross-modal alignment knowledge in CLIP's image-text shared embedding space to discern the target referent. Then a cosine similarity between the patch- and sentence-level features produces a patch-level grounding map. Lastly, a decoder refines the map into a pixel-level segmentation prediction.

summations introduce fine-grained structural details that refine the segmentation boundaries of the patch-level grounding map, thereby increasing the mIoU on the RefCOCOg-UMD test set to 48.29 mIoU.

3.2 Cross-modal feature extraction

The independent feature extraction of image and text features in CLIP yields unimodal features which are bound to limited alignment between the target instance and referring text. Consider the image-text pair in Fig. 2. Given an image of two giraffes and the text "A giraffe looking up while another giraffe next to it looks down", the patch-level features of the target giraffe are unlikely to perfectly align with the sentence-level feature, as the giraffe can be described with different texts like "the taller giraffe behind the fence" and "giraffe sticking its chin up." For the target patch-level features to better align with the sentence-level feature, they must evolve to be like the text feature, or vice versa, through cross-modal interaction. Therefore, we introduce Cross-modal Feature Extraction (CFE) modules between the unimodal image and text encoders to enable cross-modal communication via cross-attention which aligns the image and text features at text-relevant regions. Starting from the deepest layers of the backbone, we pair an image and text encoder layer and attach a single CFE module in between to communicate intermediate image and text features, where the number of CFE modules introduced is a hyperparameter.

Consider a CFE module between the k-th text and l-th image transformer layers which take as input text and image features, $\mathbf{t}_k$ and $\mathbf{v}_l$, respectively. First, the CFE module projects $\mathbf{t}_k$ and $\mathbf{v}_l$ to a shared image-text embedding space with linear projections, $\mathbf{W}_{t2s}$ and $\mathbf{W}_{v2s}$, to produce $\mathbf{t}_k^s$ and $\mathbf{v}_l^s$. Then, two separate multi-head cross-attention (MHCA) modules input $\mathbf{t}_k^s$ and $\mathbf{v}_l^s$, where each modality is set as query and the other key and value, to produce text and image multi-modal features, $\mathbf{t}_k^m$ and $\mathbf{v}_l^m$. Lastly, the multi-modal features are projected from the shared embedding space back

to each modalities' space with linear projections, $\mathbf{W}_{s2t}$ and $\mathbf{W}_{s2v}$ to produce the final multi-modal features $\mathbf{t}_k^{m'}$ and $\mathbf{v}_l^{m'}$. We elaborate on this process to output $\mathbf{t}_k^{m'}$ below, where $\mathbf{v}_l^{m'}$ can be computed in vice versa, as

$$\mathbf{t}_k^s = \mathbf{W}_{t2s}\mathbf{t}_k, \quad \mathbf{v}_l^s = \mathbf{W}_{v2s}\mathbf{v}_l, \tag{1}$$

$$\mathbf{t}_k^m = \mathrm{MHCA}(\mathbf{t}_k^s, \mathbf{v}_l^s, \mathbf{v}_l^s) \tag{2}$$

$$\mathbf{t}_k^{m'} = \mathbf{W}_{s2t}\mathbf{t}_k^m. \tag{3}$$

These multi-modal features, $\mathbf{t}_k^{m'}$ and $\mathbf{v}_l^{m'}$, are added back to the input features as $\mathbf{t}_k = \mathbf{t}_k + \mathbf{t}_k^{m'}$ and $\mathbf{v}_l = \mathbf{v}_l + \mathbf{v}_l^{m'}$ to inject multi-modal information into the CLIP features. Then, $\mathbf{t}_k$ and $\mathbf{v}_l$ are input to the k-th text and l-th image transformer layers to be processed by the subsequent MHSA and MLP. We visualize the feedforward process in the bottom-right of Fig 2.

These modules effectively inject cross-modal information into CLIP's previously unimodal features, allowing the target patch-level features to align with the sentence-level features and vice versa. The better aligned features output more accurate groundings, increasing performance to 60.58 mIoU on the RefCOCOg-UMD test set.

3.3 Shared-space knowledge exploitation

Although the CFE modules effectively adapt CLIP to RIS, we observe that they often miss the target described by complex texts about intricate relationships among multiple instances. Thus, we attempt to leverage the alignment between images and lengthy captions residing in CLIP's image-text shared embedding space, which was obtained during its extensive contrastive learning pretraining. We introduce Shared-space Knowledge Exploitation (SKE) modules which execute both inter- and intra-modal conditioning on the image and text features within the shared embedding space via residually connected MHCA, MHSA, and MLPs, with LN applied before every submodule. We visualize the feedforward in the top-right of Fig. 2. The intra-modal interactions allow the model to grasp intricate relationships between objects within in each modality, while the inter-modal interactions allow cross-modal alignment and grounding. These comprehensive interactions handle lengthy, intricate descriptions effectively, increasing the performance to 62.64 mIoU on the RefCOCOg-UMD test set.

3.4 Decoder

The patch-level grounding maps are upsampled to pixel-level predictions ($\mathbf{map}_{\text{pixel}}$) with a decoder. Since the role of the decoder is to figure out the boundary of the referred instance detected by the grounding map, we adopt a simple decoder (Yang et al., 2022). To accurately restore the geometric details of the target referent, the decoder exploits intermediate patch-level features from the first four layers of the CLIP image backbone, $\mathbf{V}_i, i \in \{1, 2, 3, 4\}$.

The decoder consists of four layers, $\mathbf{D}_i, i \in \{1, 2, 3, 4\}$, where $\mathbf{D}_i$ comprises of two repetitions of 3×3 convolutions, ReLU (Agarap, 2018), and batch normalization (Ioffe and Szegedy, 2015), followed by double-resolution upsampling. The feed forward process is given by

$$\mathbf{d}_4 = \mathbf{D}_4([\mathbf{v}_4; \mathbf{map}_{\text{patch}}])$$
$$\mathbf{d}_i = \mathbf{D}_i([\mathbf{d}_{i+1}; u(\mathbf{v}_i)]), i = 1, 2, 3,$$

where $\mathbf{d}_i$ is the output features of $\mathbf{D}_i$, $\mathbf{map}_{\text{patch}}$ the sigmoided patch-level grounding map, u the upsampling operation, and [;] channel-wise concatenation. Finally, $\mathbf{d}_1$ undergoes a linear projection that produces background and foreground score maps, which are sigmoided into the pixel-wise prediction, $\mathbf{map}_{\text{pixel}}$. The binary prediction mask is obtained via argmax during inference.

3.5 Loss functions

RISCLIP is trained in two stages. In the first stage, the Adapters, CFE, and SKE modules align the sigmoided patch-level grounding map, $\mathbf{map}_{\text{patch}}$, with a patch-level downsampled ground truth mask, $\mathbf{mask}_{\text{patch}}$. Once $\mathbf{map}_{\text{patch}}$ converges, the decoder is introduced in the second stage to upsample $\mathbf{map}_{\text{patch}}$ to a pixel-wise map, $\mathbf{map}_{\text{pixel}}$, aligning with the pixel-level ground truth mask, $\mathbf{mask}_{\text{pixel}}$. During this stage, all modules except the decoder remain frozen, and the decoder only is trained for a single epoch. Although training the entire framework end-to-end with the decoder is feasible, this approach inevitably makes the decoder receive random patch-level grounding maps during initial training steps, resulting in meaningless training signals and wasted computation. Conversely, the two-stage training simplifies the decoder's role to delineating patch-level grounding boundaries, requiring only one epoch for convergence in the second stage. This approach enhances efficiency by minimizing unnecessary computation.

Following (Li and Sigal, 2021), we adopt a linear combination of DICE/F-1 loss (Milletari et al., 2016) and focal loss (Lin et al., 2017) for both training stages, first between $\mathbf{map}_{\text{patch}}$ and $\mathbf{mask}_{\text{patch}}$ and after between $\mathbf{map}_{\text{pixel}}$ and $\mathbf{mask}_{\text{pixel}}$.

4 Experiments

4.1 Datasets and evaluation metrics

Datasets. We evaluate RISCLIP on three major RIS datasets: RefCOCO (Yu et al., 2016), RefCOCO+ (Yu et al., 2016), and RefCOCOg (Mao et al., 2016). The RefCOCO family originates from the same MSCOCO (Lin et al., 2014) dataset and thus shares images but possesses different texts. RefCOCO (Yu et al., 2016) and RefCOCO+ (Yu et al., 2016) texts are relatively concise, consisting of 3.6 words and 1.6 nouns on average. RefCOCO+ (Yu et al., 2016) differs from RefCOCO (Yu et al., 2016) in that the texts do not include absolute positional information, such as first, second, left, and right, is thus more difficult. Lastly, RefCOCOg (Mao et al., 2016) comprises of longer, more complex texts (8.4 words and 2.8 nouns per text) and is thus the most challenging. We evaluate on the conventionally used UMD split (Nagaraja et al., 2016).

Evaluation metrics. We employ two metrics widely used in RIS: the overall intersection-over-union (oIoU) and the mean intersection-over-union (mIoU). The oIoU is the sum of all intersections over the sum of all unions, while the mIoU is the average of intersection over unions. The mIoU is a fairer metric than the oIoU, which is biased towards large objects (Yang et al., 2022). Hence, we report both oIoU and mIoUs but adopt mIoUs when comparing with previous methods.

4.2 Model settings

To explore the effect of the CLIP backbone size, we experiment with two backbones trained with ViT-B and ViT-L (Vaswani et al., 2017) and dub our framework RISCLIP-B and -L, respectively. In RISCLIP-B, we use the 12-layer ViT-B (Vaswani et al., 2017) with patch size 16×16 as the image encoder and a 12-layer transformer as the text encoder. In RISCLIP-L, we use ViT-L (Vaswani et al., 2017) with patch size 14×14 and the same 12-layer text transformer as in RISCLIP-B. For both RISCLIP-B and -L, we attach Adapters in all layers of both encoders, six CFE, and six SKE modules. Other hyperparameters are detailed in Appendix A.1.

4.3 Comparison with state of the arts

We compare RISCLIP with previous methods on the three aforementioned datasets in Table 1. On RefCOCOg-UMD (Nagaraja et al., 2016), RISCLIP-B achieves superior performance compared to DMMI (Hu et al., 2023), with an average improvement of 1.01 mIoU points across both the validation and test sets. Analogously, RISCLIP-B surpasses DMMI by 0.63 and 1.13 mIoU points on RefCOCO (Yu et al., 2016) and RefCOCO+ (Yu et al., 2016), respectively. Furthermore, RISCLIP-L, which adopts a larger image encoder, advances the frontier set by RISCLIP-B by an average of 3.96, 3.20, and 5.54 mIoU points, respectively. Such performance improvement across all datasets demonstrates the competency of RISCLIP.

We compare RISCLIP to previous work that leverage CLIP: CRIS (Wang et al., 2022) and ETRIS (Xu et al., 2023b). Different from RISCLIP-L which uses ViT-L (Vaswani et al., 2017) as the image encoder, CRIS uses ResNet-101 (He et al., 2016). RISCLIP-L surpasses CRIS by an average of 11.62, 8.66, and 11.99 mIoU points on the three datasets, respectively. RISCLIP-B surpasses ETRIS by an average of 4.73, 3.05, and 4.82 oIoU points, respectively. Such performance difference shows that we utilize CLIP effectively.

Also, we compare RISCLIP to PolyFormer (Liu et al., 2023b) in a separate Table 2, since PolyFormer was trained on the combined RefCOCO family while the others were trained on each dataset separately. We also train RISCLIP on the combined dataset following PolyFormer for fair comparison. RISCLIP-B attains comparable performance to PolyFormer-B, but with bigger backbones, RISCLIP-L outperforms PolyFormer-L by an average of 2.83, 2.39, and 2.43 mIoU points on the three datasets. In summary, RISCLIP achieves a new state of the art.

4.4 Ablation studies

We conduct ablation studies on the test set of RefCOCOg-UMD (Nagaraja et al., 2016) to prove the effectiveness of our framework and verify its architectural designs. For expedited experiments, we conduct them with a small image size of 240×240 for 50 epochs. Other hyperparameters are the same as those written in Appendix in A.1.

Module ablation. We validate the effectiveness of Adapters, CFE, and SKE modules by progressively introducing each module to frozen CLIP in Table 3.

Table 1: Comparison with state of the arts on RefCOCO (Yu et al., 2016), RefCOCO+ (Yu et al., 2016), and RefCOCOg-UMD (Mao et al., 2016; Nagaraja et al., 2016). We reproduce DMMI (Hu et al., 2023) on RefCOCO and RefCOCO+ with the official code to report its mIoU scores unprovided in the original paper. RN101 is ResNet-101 (He et al., 2016), DN53 Darknet-53 (Redmon and Farhadi, 2018), and WRN101 Wide ResNet-101 (Zagoruyko and Komodakis, 2016). CLIP-B and CLIP-L denote the Transformer-based CLIPs which adopt ViT-B and ViT-L (Vaswani et al., 2017) as the image encoder, respectively, while CLIP-L* is the ResNet-based CLIP which utilizes ResNet-101 (He et al., 2016).

Method	Image Encoder	Text Encoder	RefCOCOg		RefCOCO			RefCOCO+		
			Val	Test	Val	Test A	Test B	Val	Test A	Test B
oIoU										
BRINet (Hu et al., 2020)	RN101	LSTM	-	-	60.98	62.99	59.21	48.17	52.32	42.11
CMPC (Huang et al., 2020)	RN101	LSTM	-	-	61.36	64.53	59.64	49.56	53.44	43.23
LSCM (Hui et al., 2020)	RN101	LSTM	-	-	61.47	64.99	59.55	49.34	53.12	43.50
CMPC+ (Liu et al., 2022)	RN101	LSTM	-	-	62.47	65.08	60.82	50.25	54.04	43.47
MCN (Luo et al., 2020b)	DN53	Bi-GRU	49.22	49.40	62.44	64.20	59.71	50.62	54.99	44.69
BUSNet (Yang et al., 2021)	RN101	Self-Attn	-	-	63.27	66.41	61.39	51.76	56.87	44.13
CGAN (Luo et al., 2020a)	DN53	Bi-GRU	51.01	51.69	64.86	68.04	62.07	51.03	55.51	44.06
LTS (Jing et al., 2021)	DN53	Bi-GRU	54.40	54.25	65.43	67.76	63.08	54.21	58.32	48.02
ReSTR (Kim et al., 2022)	ViT-B	BERT	-	-	67.22	69.30	64.45	55.78	60.44	48.27
ETRIS (Xu et al., 2023b)	CLIP-B	CLIP-B	59.82	59.91	70.51	73.51	66.63	60.10	66.89	50.17
LAVT (Yang et al., 2022)	Swin-B	BERT	61.24	62.09	72.73	75.82	68.79	62.14	68.38	55.10
SLViT (OuYang et al., 2023)	SegNeXt	BERT	62.75	63.57	74.02	76.91	70.62	64.07	69.28	56.14
DMMI (Hu et al., 2023)	Swin-B	BERT	63.46	64.19	74.13	77.13	70.16	63.98	69.73	57.03
DMMI (Reproduced)	Swin-B	BERT	-	-	73.79	75.67	69.96	63.85	69.65	55.71
RISCLIP-B	CLIP-B	CLIP-B	64.10	65.09	73.57	76.46	69.76	65.53	70.61	55.49
RISCLIP-L	CLIP-L	CLIP-L	**67.96**	**68.71**	**76.92**	**80.99**	**73.04**	**71.24**	**76.99**	**61.56**
mIoU										
CRIS (Wang et al., 2022)	CLIP-L*	CLIP-L*	59.87	60.36	70.47	73.18	66.10	62.27	68.06	53.68
SeqTR (Zhu et al., 2022)	DN53	Bi-GRU	64.69	65.74	71.70	73.31	69.82	63.04	66.73	58.97
RefTR (Li and Sigal, 2021)	RN101	BERT	66.63	67.39	74.34	76.77	70.87	66.75	70.58	59.40
LAVT (Yang et al., 2022)	Swin-B	BERT	63.34	63.62	74.46	76.89	70.94	65.81	70.97	59.23
VLT (Ding et al., 2023)	Swin-B	Bi-GRU	63.49	66.22	72.96	75.96	69.60	63.53	68.43	56.92
DMMI (Hu et al., 2023)	Swin-B	BERT	66.48	67.07	-	-	-	-	-	-
DMMI (Reproduced)	Swin-B	BERT	-	-	75.26	76.96	72.05	67.51	72.1	60.38
RISCLIP-B	CLIP-B	CLIP-B	67.61	67.95	75.68	78.01	72.46	69.16	73.53	60.68
RISCLIP-L	CLIP-L	CLIP-L	**71.82**	**71.65**	**78.87**	**81.46**	**75.41**	**74.38**	**78.77**	**66.84**

Table 2: Comparison with PolyFormer (Liu et al., 2023b) in mIoU. Both RISCLIP and PolyFormer are trained on the combined RefCOCO dataset (Yu et al., 2016; Mao et al., 2016; Nagaraja et al., 2016).

Method	Image Encoder	Text Encoder	RefCOCOg		RefCOCO			RefCOCO+		
			Val	Test	Val	Test A	Test B	Val	Test A	Test B
PolyFormer-B (Liu et al., 2023b)	Swin-B	BERT	69.36	**69.88**	75.96	77.09	**73.22**	**70.65**	**74.51**	**64.64**
RISCLIP-B	CLIP-B	CLIP-B	**69.61**	69.56	**76.01**	**78.63**	71.94	69.67	74.30	61.37
PolyFormer-L (Liu et al., 2023b)	Swin-L	BERT	71.15	71.17	76.94	78.49	74.83	72.15	75.71	66.73
RISCLIP-L	CLIP-L	CLIP-L	**73.45**	**74.52**	**79.53**	**82.13**	**75.78**	**74.88**	**78.88**	**68.09**

Introducing Adapters boosts performance by an mIoU of 24.43, proving that Adapters effectively adapt CLIP to the segmentation task. Moreover, attaching CFE improves performance by 12.29 mIoU, indicating that transforming unimodal feature extraction into a cross-modal one is beneficial for RIS. Introducing SKE further pushes the performance by 2.06 mIoU, showing that comprehensive interaction within CLIP's image-text shared-embedding space is helpful. In contrast, finetuning CLIP along with the modules performs worse than its frozen CLIP twin, with a mIoU drop of 4.76. Thus, our choice of residually adapting frozen CLIP features with Adapters, CFE, and SKE is a viable approach.

Table 3: Performance when Adapters, CFE, and SKE modules are successively introduced into frozen CLIP. The last row ('Fine-tuned') denotes the setting where CLIP is fine-tuned along with the introduced modules.

RISCLIP-B	Adapter	CFE	SKE	mIoU	oIoU
Frozen	✗	✗	✗	23.86	33.13
Frozen	✓	✗	✗	48.29	50.98
Frozen	✓	✓	✗	60.58	58.39
Frozen	✓	✓	✓	62.64	62.02
Fine-tuned	✓	✓	✓	57.88	55.75

Table 4: Performance when MHCA in CFE and SKE modules are replaced with other more complex fusion methods, including state-of-the-art fusion mechanisms (Yang et al., 2022; Ding et al., 2023).

	Fusion Direction	mIoU	oIoU
a) MHCA replaced with complex attention-based fusion modules			
MHCA (Ours)	Bidirectional	62.64	62.02
MHSA on Concat	Bidirectional	62.63	61.65
MHCA on Concat	Bidirectional	62.04	60.87
b) MHCA replaced with state-of-the-art fusion modules			
PWAM (Yang et al., 2022)	Text-to-Image	60.58	59.29
PWAM (Yang et al., 2022)	Bidirectional	61.01	59.39
SDF (Ding et al., 2023)	Text-to-Image	59.8	57.81
SDF (Ding et al., 2023)	Bidirectional	60.28	58.87

Fusion ablation. We validate our choice of a simple MHCA for cross-modal fusion instead of more complex fusion methods. Specifically, we replace the MHCA in CFE and SKE modules with other attention-based fusion methods such as MHSA on concatenated image and text tokens (MHSA on Concat) and MHCA where the query is one modality's tokens and the key, value are the concatenated image and text tokens (MHCA with Concat). Firstly, 'MHSA on Concat' produces a slight performance decrease (0.01 mIoU and 0.37 oIoU), indicating it's a viable option. Yet, the computation increase due to attention between the summed number of image and text tokens makes it less efficient than the simple MHCA. Secondly, 'MHCA with Concat' decreases performance by 0.6 mIoU and 1.15 oIoU. In summary, the simple MHCA is an efficient yet effective attention mechanism for our CFE and SKE modules. The results are summarized in section a) of Table 4.

We also demonstrate the superiority of the simple MHCA over existing state-of-the-art fusion modules like Pixel-Word Attention Module (PWAM) in LAVT (Yang et al., 2022) and Spatial-Dynamic Fusion (SDF) in VLT (Ding et al., 2023) for adapting CLIP to RIS. Since both modules perform unidirectional fusion by conditioning image features on text features, we implement bidirectional versions for fair comparison to our CFE and SKE modules that execute bidirectional fusion. As shown in Table 4, replacing the simple MHCA with these modules in CFE and SKE all results in performance drops, suggesting that complex fusion modules are not needed to adapt CLIP to RIS.

Table 5: Performance when the number of Adapters, CFE, and SKE modules are varied. The original setting of 12 Adapters, six CFE, and six SKE modules is marked with asterisk.

	Prec@0.5	Prec@0.7	Prec@0.9	mIoU	oIoU
a) Adapters attached to N last CLIP encoder layers					
$N = 3$	71.81	55.3	11.29	61.40	60.84
$N = 6$	72.73	56.53	11.87	62.15	61.33
$N = 9$	72.50	57.44	14.03	62.31	60.68
$N = 12$*	73.19	57.68	14.21	62.64	62.02
b) CFE modules attached to N last CLIP encoder layers					
$N = 2$	72.56	57.05	14.16	62.33	61.34
$N = 4$	72.33	57.31	13.89	62.41	61.47
$N = 6$*	73.19	57.68	14.21	62.64	62.02
c) SKE modules of N layers attached behind CLIP encoders					
$N = 2$	72.17	56.72	14.23	62.30	61.56
$N = 4$	72.73	57.68	14.57	62.79	61.95
$N = 6$*	73.19	57.68	14.21	62.64	62.02

Architecture ablation. We investigate the effect of our modules by varying their numbers in a baseline model. The results are summarized in Table 5. Section a) shows that performance improves with the number of Adapters attached to the latter CLIP encoder layers. Such a trend suggests that Adapters can beneficially adapt CLIP features to RIS at all layers. In section b), performance increases with the number of CFE modules, indicating that using more cross-modal interaction during feature extraction is advantageous. In section c), the performance plateaus from 4 to 6 SKE modules, suggesting that there is a limit to the benefits that interaction within the image-text space can bring.

4.5 Visualizations

We visualize predictions of RISCLIP-B on the RefCOCOg-UMD (Nagaraja et al., 2016) test set. Fig. 3 shows our model's ability to capture a wide variety of instances, detect partially visible or blurry targets, and differentiate the ground truth from resemblances, even with complicated expressions. More visualizations are provided in Appendix A.2.2.

Figure 3: Visualization of RISCLIP-B predictions on RefCOCOg-UMD (Nagaraja et al., 2016) test set. Row a) shows RISCLIP's understanding of various instances, row b) RISCLIP's detection of partial, blurry instances and differentiate similar objects, row c) RISCLIP's discernment of the target instance among resembling instances described by lengthy texts.

5 Conclusion

RISCLIP effectively extends the image-text alignment of CLIP to RIS, achieving outstanding performance on all major RIS benchmarks. We effectively build upon CLIP's patch-level image-text alignment by introducing cross-modal communication during feature extraction and leveraging the rich cross-modal alignment within CLIP's image-text shared-embedding space to successfully delineate referents described by complicated texts.

6 Limitations

We can improve our work by adopting other image-text alignment backbones such as ALIGN (Jia et al., 2021) and Florence (Yuan et al., 2021). Such extension would allow us to investigate image-text alignment within various cross-modal foundation models and the effectiveness of adapting them to RIS. Also, while RISCLIP achieves state-of-the-art results with impressive margins, there are complex cases where our framework struggles to identify the target instance accurately. We include these cases in Appendix A.2.1.

7 Broader impacts

RIS holds the potential to impact numerous domains that use human-computer interaction, such as autonomous driving and assistant robots. For example, a user could instruct a domestic service robot to "fetch the blue cup, not the red one", and the RIS-built-in robot will be able to accurately detect the blue cup and serve his/her owner. Nevertheless, potential ethical concerns, including privacy, model bias, and data processing should be considered. Even the RefCOCO (Yu et al., 2016; Mao et al., 2016) dataset includes offensive expressions and provocative images that require removal. In summary, RIS will impact diverse fields adopting human-computer interaction, but ethical issues should be addressed to ensure beneficial development and safe deployment.

Acknowledgement. This work was supported by the NRF grant and the IITP grant funded by Ministry of Science and ICT, Korea (NRF-2021R1A2C3012728–35%, IITP-2019-0-01906–5%, IITP-2021-0-01343–5%, IITP-2021-0-02068–20%, IITP-2022-0-00290–35%).

References

Abien Fred Agarap. 2018. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450*.

Tim Brooks, Aleksander Holynski, and Alexei A. Efros. 2023. InstructPix2Pix: Learning to Follow Image Editing Instructions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Neil Burgess, Jelena Milanovic, Nigel Stephens, Konstantinos Monachopoulos, and David Mansell. 2019. Bfloat16 processing for neural networks. In *2019 IEEE 26th Symposium on Computer Arithmetic (ARITH)*. IEEE.

Ding-Jie Chen, Songhao Jia, Yi-Chen Lo, Hwann-Tzong Chen, and Tyng-Luh Liu. 2019. See-through-text grouping for referring image segmentation. In *IEEE International Conference on Computer Vision (ICCV)*.

Jianbo Chen, Yelong Shen, Jianfeng Gao, Jingjing Liu, and Xiaodong Liu. 2018. Language-Based Image Editing with Recurrent Attentive Models. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Shoufa Chen, Chongjian Ge, Zhan Tong, Jiangliu Wang, Yibing Song, Jue Wang, and Ping Luo. 2022. AdaptFormer: Adapting Vision Transformers for Scalable Visual Recognition. In *Conference on Neural Information Processing Systems (NeurIPS)*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, 2019*.

Henghui Ding, Chang Liu, Suchen Wang, and Xudong Jiang. 2021. Vision-language transformer and query generation for referring segmentation. In *IEEE International Conference on Computer Vision (ICCV)*.

Henghui Ding, Chang Liu, Suchen Wang, and Xudong Jiang. 2023. VLT: Vision-Language Transformer and Query Generation for Referring Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*.

Yu Du, Fangyun Wei, Zihe Zhang, Miaojing Shi, Yue Gao, and Guoqi Li. 2022. Learning to prompt for open-vocabulary object detection with vision-language model. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Guang Feng, Zhiwei Hu, Lihe Zhang, and Huchuan Lu. 2021. Encoder Fusion Network with Co-Attention Embedding for Referring Image Segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Philip Gage. 1994. A new algorithm for data compression. *C Users Journal*.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. In *IEEE International Conference on Computer Vision (ICCV)*.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. 2021. Clipscore: A reference-free evaluation metric for image captioning. *arXiv preprint arXiv:2104.08718*.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-Efficient Transfer Learning for NLP. In *International Conference on Machine Learning (ICML)*.

Ronghang Hu, Marcus Rohrbach, and Trevor Darrell. 2016. Segmentation from natural language expressions. In *European Conference on Computer Vision (ECCV)*.

Yutao Hu, Qixiong Wang, Wenqi Shao, Enze Xie, Zhenguo Li, Jungong Han, and Ping Luo. 2023. Beyond One-to-One: Rethinking the Referring Image Segmentation. In *IEEE International Conference on Computer Vision (ICCV)*.

Zhiwei Hu, Guang Feng, Jiayu Sun, Lihe Zhang, and Huchuan Lu. 2020. Bi-directional relationship inferring network for referring image segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Shaofei Huang, Tianrui Hui, Si Liu, Guanbin Li, Yunchao Wei, Jizhong Han, Luoqi Liu, and Bo Li. 2020. Referring image segmentation via cross-modal progressive comprehension. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Tianrui Hui, Si Liu, Shaofei Huang, Guanbin Li, Sansi Yu, Faxi Zhang, and Jizhong Han. 2020. Linguistic structure guided context modeling for referring image segmentation. In *European Conference on Computer Vision (ECCV)*.

Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. 2021. Openclip. *https://doi.org/10.5281/zenodo.5143773*.

Sergey Ioffe and Christian Szegedy. 2015. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *International Conference on Machine Learning (ICML)*.

Kanishk Jain and Vineet Gandhi. 2022. Comprehensive multi-modal interactions for referring image segmentation. In *Findings of the Association for Computational Linguistics (Findings of ACL)*.

Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V. Le, Yunhsuan Sung, Zhen Li, and Tom Duerig. 2021. Scaling Up Visual and Vision-Language Representation Learning With Noisy Text Supervision. In *International Conference on Machine Learning (ICML)*.

Ya Jing, Tao Kong, Wei Wang, Liang Wang, Lei Li, and Tieniu Tan. 2021. Locate then segment: A strong pipeline for referring image segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Dongwon Kim, Namyup Kim, Cuiling Lan, and Suha Kwak. 2023. Shatter and Gather: Learning Referring Image Segmentation with Text Supervision. In *IEEE International Conference on Computer Vision (ICCV)*.

Namyup Kim, Dongwon Kim, Cuiling Lan, Wenjun Zeng, and Suha Kwak. 2022. ReSTR: Convolution-free Referring Image Segmentation Using Transformers. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Jungbeom Lee, Sungjin Lee, Jinseok Nam, Seunghak Yu, Jaeyoung Do, and Tara Taghavi. 2023. Weakly supervised referring image segmentation with intra-chunk and inter-chunk consistency. In *IEEE International Conference on Computer Vision (ICCV)*.

Muchen Li and Leonid Sigal. 2021. Referring transformer: A one-step approach to multi-task visual grounding. In *Conference on Neural Information Processing Systems (NeurIPS)*.

Ruiyu Li, Kaican Li, Yi-Chun Kuo, Michelle Shu, Xiaojuan Qi, Xiaoyong Shen, and Jiaya Jia. 2018. Referring image segmentation via recurrent refinement networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. 2017. Focal loss for dense object detection. In *IEEE International Conference on Computer Vision (ICCV)*.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision (ECCV)*.

Chang Liu, Henghui Ding, and Xudong Jiang. 2023a. GRES: Generalized Referring Expression Segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Chenxi Liu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, and Alan Yuille. 2017. Recurrent multimodal interaction for referring image segmentation. In *IEEE International Conference on Computer Vision (ICCV)*.

Jiang Liu, Hui Ding, Zhaowei Cai, Yuting Zhang, Ravi Kumar Satzoda, Vijay Mahadevan, and R Manmatha. 2023b. PolyFormer: Referring Image Segmentation as Sequential Polygon Generation. *arXiv preprint arXiv:2302.07387*.

Si Liu, Tianrui Hui, Shaofei Huang, Yunchao Wei, Bo Li, and Guanbin Li. 2022. Cross-modal progressive comprehension for referring segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. 2021. Swin transformer: Hierarchical vision transformer using shifted windows. In *IEEE International Conference on Computer Vision (ICCV)*.

Ilya Loshchilov and Frank Hutter. 2019a. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations (ICLR)*.

Ilya Loshchilov and Frank Hutter. 2019b. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations (ICLR)*.

Gen Luo, Yiyi Zhou, Rongrong Ji, Xiaoshuai Sun, Jinsong Su, Chia-Wen Lin, and Qi Tian. 2020a. Cascade grouped attention network for referring expression segmentation. In *Proceedings of the 28th ACM International Conference on Multimedia*.

Gen Luo, Yiyi Zhou, Xiaoshuai Sun, Liujuan Cao, Chenglin Wu, Cheng Deng, and Rongrong Ji. 2020b. Multi-task collaborative network for joint referring expression comprehension and segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Huaishao Luo, Junwei Bao, Youzheng Wu, Xiaodong He, and Tianrui Li. 2022. SegCLIP: Patch Aggregation with Learnable Centers for Open-Vocabulary Semantic Segmentation. *arXiv preprint arXiv:2211.14813*.

Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. 2016. Generation and comprehension of unambiguous object descriptions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Edgar Margffoy-Tuay, Juan C Pérez, Emilio Botero, and Pablo Arbeláez. 2018. Dynamic multimodal instance segmentation guided by natural language queries. In *European Conference on Computer Vision (ECCV)*.

Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. 2016. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 fourth international conference on 3D vision (3DV)*.

Ron Mokady, Amir Hertz, and Amit H Bermano. 2021. ClipCap: CLIP Prefix for Image Captioning. *arXiv preprint arXiv:2111.09734*.

Varun K. Nagaraja, Vlad I. Morariu, and Larry S. Davis. 2016. Modeling Context Between Objects for Referring Expression Understanding. In *European Conference on Computer Vision (ECCV)*.

S-R OuYang, Hongyi Wang, Shiao Xie, Ziwei Niu, Ruofeng Tong, Yen-Wei Chen, and Lanfen Lin. 2023. SLViT: Scale-Wise Language-Guided Vision Transformer for Referring Image Segmentation. In *International Joint Conference on Artificial Intelligence*.

Gaurav Parmar, Krishna Kumar Singh, Richard Zhang, Yijun Li, Jingwan Lu, and Jun-Yan Zhu. 2023. Zero-shot image-to-image translation. *arXiv preprint arXiv:2302.03027*.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *International Conference on Machine Learning (ICML)*.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. 2022. Hierarchical Text-Conditional Image Generation with CLIP Latents. *arXiv preprint arXiv:2204.06125*.

Hanoona Rasheed, Muhammad Maaz, Muhammad Uzair Khattak, Salman Khan, and Fahad Shahbaz Khan. 2022. Bridging the Gap between Object and Image-level Representations for Open-Vocabulary Detection. In *Conference on Neural Information Processing Systems (NeurIPS)*.

Joseph Redmon and Ali Farhadi. 2018. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.

Tal Ridnik, Emanuel Ben-Baruch, Asaf Noy, and Lihi Zelnik-Manor. 2021. ImageNet-21K Pretraining for the Masses. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. 2021. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. In *Conference on Neural Information Processing Systems (NeurIPS) Workshop*.

Hengcan Shi, Hongliang Li, Fanman Meng, and Qingbo Wu. 2018. Key-word-aware network for referring expression image segmentation. In *European Conference on Computer Vision (ECCV)*.

Robin Strudel, Ivan Laptev, and Cordelia Schmid. 2022. Weakly-supervised segmentation of referring expressions. *arXiv preprint arXiv:2205.04725*.

Yucheng Suo, Linchao Zhu, and Yi Yang. 2023. Text Augmented Spatial-aware Zero-shot Referring Image Segmentation. In *Proc. Empirical Methods in Natural Language Processing (EMNLP)*.

Jiajin Tang, Ge Zheng, Cheng Shi, and Sibei Yang. 2023. Contrastive Grouping with Transformer for Referring Image Segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Conference on Neural Information Processing Systems (NeurIPS)*.

Xin Wang, Qiuyuan Huang, Asli Celikyilmaz, Jianfeng Gao, Dinghan Shen, Yuan-Fang Wang, William Yang Wang, and Lei Zhang. 2019. Reinforced Cross-Modal Matching and Self-Supervised Imitation Learning for Vision-Language Navigation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zhaoqing Wang, Yu Lu, Qiang Li, Xunqiang Tao, Yandong Guo, Mingming Gong, and Tongliang Liu. 2022. CRIS: Clip-driven referring image segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Mengde Xu, Zheng Zhang, Fangyun Wei, Han Hu, and Xiang Bai. 2023a. Side Adapter Network for Open-Vocabulary Semantic Segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zunnan Xu, Zhihong Chen, Yong Zhang, Yibing Song, Xiang Wan, and Guanbin Li. 2023b. Bridging Vision and Language Encoders: Parameter-Efficient Tuning for Referring Image Segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Sibei Yang, Meng Xia, Guanbin Li, Hong-Yu Zhou, and Yizhou Yu. 2021. Bottom-up shift and reasoning for referring image segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zhao Yang, Jiaqi Wang, Yansong Tang, Kai Chen, Hengshuang Zhao, and Philip HS Torr. 2022. LAVT: Language-Aware Vision Transformer for Referring Image Segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Linwei Ye, Mrigank Rochan, Zhi Liu, and Yang Wang. 2019. Cross-modal self-attention network for referring image segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Licheng Yu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, Mohit Bansal, and Tamara L Berg. 2018. Mattnet: Modular attention network for referring expression comprehension. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. 2016. Modeling context in referring expressions. In *European Conference on Computer Vision (ECCV)*.

Seonghoon Yu, Paul Hongsuck Seo, and Jeany Son. 2023. Zero-shot referring image segmentation with global-local context features. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, et al. 2021. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*.

Sergey Zagoruyko and Nikos Komodakis. 2016. Wide Residual Networks. In *Proc. British Machine Vision Conference (BMVC)*.

Zicheng Zhang, Yi Zhu, Jianzhuang Liu, Xiaodan Liang, and Wei Ke. 2019. Coupalign: Coupling word-pixel with sentence-mask alignments for referring image segmentation. In *Conference on Neural Information Processing Systems (NeurIPS)*.

Chong Zhou, Chen Change Loy, and Bo Dai. 2022. Extract Free Dense Labels from CLIP. In *European Conference on Computer Vision (ECCV)*.

Chaoyang Zhu, Yiyi Zhou, Yunhang Shen, Gen Luo, Xingjia Pan, Mingbao Lin, Chao Chen, Liujuan Cao, Xiaoshuai Sun, and Rongrong Ji. 2022. SeqTR: A Simple Yet Universal Network for Visual Grounding. In *European Conference on Computer Vision (ECCV)*.

A Appendices

A.1 Training details

Training scheme. We train both RISCLIP-B and -L for 60 epochs with AdamW (Loshchilov and Hutter, 2019b) optimizer, using weight decay of 5e-3 and an initial learning rate of 5e-5 with polynomial learning rate decay. Images are resized to 640×640 for RISCLIP-B and 560×560 for RISCLIP-L, such that the visual encoders are both fed 40×40 patch tokens. We apply random affine transformation and random intensity saturation data augmentations following RefTR (Li and Sigal, 2021). The ratio between dice (Milletari et al., 2016) and focal loss (Lin et al., 2017), λ_{dice} and λ_{focal}, is empirically set to 1.0 to 1.75, and alpha and gamma, α_{focal} and γ_{focal}, in the focal loss are set to 0.65 and 2.0. We use batch size of 32 for the models trained on separate RefCOCO datasets (Yu et al., 2016; Mao et al., 2016) (reported in Table 1), whilst we use bigger batch sizes of 96 for RISCLIP-B and 56 for RISCLIP-L trained on the combined RefCOCO family (Yu et al., 2016; Mao et al., 2016) (reported in Table 2) to prevent prolonged training.

Initializations. The backbone encoders are initialized from different sources for RISCLIP-B and -L. In RISCLIP-B, the backbone encoders are initialized with the official weights of OpenCLIP (Ilharco et al., 2021) pretrained on LAION-400M (Schuhmann et al., 2021). On the other hand, RISCLIP-L's backbone encoders are initialized with the official weights of CLIP (Radford et al., 2021) pretrained on 400 million image-text pairs collected by OpenAI. We use different sources for the pretrained weights because each source provides a model pretrained with a bigger image size than the other source (*i.e.* OpenCLIP provides a ViT-B backbone pretrained with image size 240×240 pixels whilst OpenAI provides one with 224×224 pixels). We empirically find that using a backbone pretrained with a bigger image size provides better segmentation ability.

The Adapters adopt different initializations. For the Adapters, we follow (Chen et al., 2022) and initialize the down-projection linear layer with Kaiming Normal (He et al., 2015) and the up-projection layer with zeros. Initializing the up-projection with zeros makes the initial adapter output zero, which is required for stable training (Chen et al., 2022). Inspired by this, we also initialize CFE and SKE modules such that the outputs are initially zero. In detail, for CFE modules, we initialize the image-text shared embedding projections in the MHSA as zeros, and, for SKE modules, the value projections in MHA and MHSA as zeros. We experiment with other compositions and find that the adopted initialization provides the best performance, which is slightly better than the others (about 0.6 IoUs).

Additional techniques. Furthermore, we observe that incorporating learnable temperatures in the attention modules of the Adapters and introducing learnable channel-wise scalers before residual summation of the Adapter outputs lead to a slight enhancement in performance (up to 0.5 IoU points). All hyperparameters are listed in Table A1.

A.2 Analysis

In Sections A.2.1 and A.2.2, we analyse RISCLIP-B and RISCLIP-L trained on RefCOCOg-

Table A1: Hyperparameters for training RISCLIP-B and -L on the separate RefCOCO datasets (Yu et al., 2016; Mao et al., 2016; Nagaraja et al., 2016). The only difference when training on the combined RefCOCO family (Yu et al., 2016; Mao et al., 2016; Nagaraja et al., 2016) is the batch size, which is increased from 32 to 96 and 56 for RISCLIP-B and -L, respectively. We denote Adam with decoupled weight decay (Loshchilov and Hutter, 2019a) as AdamW, rectified linear unit (Agarap, 2018) as ReLU, Brain Floating Point (Burgess et al., 2019) format as BF16, and single-precision floating-point format as FP32.

Hyperparameters	**RISCLIP-B**	**RISCLIP-L**
Backbone		
Pretrained Weight Source	OpenAI	OpenCLIP
Image Encoder Patch Size	16	14
Image Encoder Transformer Layers	12	24
Text Encoder Transformer Layers	12	12
Image Encoder MHA Head Number	14	16
Text Encoder MHA Head Number	10	12
$\mathbf{f}_L^v$ dimension	896	1024
$\mathbf{f}_L^t$ dimension	640	768
$\mathbf{v}$ dimension	640	768
$\mathbf{t}$ dimension	640	768
Adapters		
Image Encoder Adapter Bottleneck dimension	449	512
Text Encoder Adapter Bottleneck dimension	320	384
Non-linear Activation	ReLU	ReLU
Scaler Initial value	0.6	0.6
Cross-modal Feature Extraction (CFE)		
Module Number	6	6
$\mathbf{s}_{m-1}^v$	640	768
$\mathbf{s}_{m-1}^t$	640	768
MHA Head Number	10	12
Scaler Initial value	0.5	0.5
Shared-space Knowledge Exploitation (SKE)		
Module Number	6	6
MHA, MHSA Head Number	8	8
Scaler Initial value	0.5	0.5
Others		
Image Size	640	560
Batch Size	32	32
Epochs	60	60
Optimizer	AdamW	AdamW
β_1 for AdamW	0.9	0.9
β_2 for AdamW	0.999	0.999
Learning Rate Initial Value	5e-5	5e-5
Weight Decay Strength	5e-3	5e-3
λ_{dice}	1.0	1.0
λ_{focal}	1.75	1.75
α_{focal}	0.65	0.65
γ_{focal}	2.0	2.0
Locator Precision	BF16	BF16
Refiner Precision	FP32	FP32

UMD (Nagaraja et al., 2016)). We choose RefCOCOg (Mao et al., 2016) among the three datasets since it possesses longer and more expressive texts, which offer greater insight about the types of texts that RISCLIP understands and struggles with.

A.2.1 Failure cases

Referring Image Segmentation is a challenging task that involves a various expressions and images. Thus, how to group and categorize the image-text pairs is ambiguous. Nevertheless, we attempt to identify common scenarios where RISCLIP often makes false predictions. Analysing predictions made by RISCLIP-B on the test set, we observe that RISCLIP tends to struggle in two situations: "Recognition of Characters" and "Comprehension of Absence". We illustrate each case with visualizations, where the ground-truth masks are displayed in blue and predictions made by RISCLIP in pink.

Recognition of characters. The first case involves the recognition of characters. Figure A1 shows that RISCLIP fails to detect numbers '13' and '48', the letter 'B', and the word 'STOP'.

Comprehension of absence. The second case concerns texts that describe the target instance with the 'absence' of some attribute. Figure A2 shows examples where RISCLIP struggles to comprehend instances described as "A squat vase with *no* flowers" and "The man with the bat wearing his shirt *un*tucked".

We hypothesize that RISCLIP's relatively poor performance in the two scenarios arises from the limited number of such texts in the dataset. Improving RISCLIP to excel in these cases is another direction for future research.

A.2.2 Visualizations

RISCLIP-B. We provide visualizations of cases where RISCLIP-B successfully segments the target instance on the RefCOCOg-UMD (Nagaraja et al., 2016)) test set in Figure A3. Even when the texts are lengthy and similar instances exist in the image, RISCLIP-B successfully discerns the referred instance.

RISCLIP-L. As observed in Table 1, RISCLIP-L performs better than RISCLIP-B. Thus, we provide visual representations of examples where RISCLIP-L successfully identifies target instances that are overlooked by RISCLIP-B on the RefCOCOg-UMD (Nagaraja et al., 2016)) test set in Figure A4. The segments colored in pink on the left are the predictions made by RISCLIP-B, while the purple segments on the right are those made by RISCLIP-L.

The visualizations suggest that RISCLIP-L possesses an additional capability to detect targets that are only partially visible or require the recognition of subtle visual cues. Such ability can be attributed to the more fine-grained CLIP image encoder of RISCLIP-L: during CLIP (Radford et al., 2021) pretraining, the CLIP image encoder of RISCLIP-L is trained with image size 336×336 and patch size 14×14 which results in $24\times24=576$ tokens, whilst that of RISCLIP-B is pretrained with image size 240×240 and patch size 16×16 which amounts to $15\times15=225$ tokens. Thus, RISCLIP-L possesses are more fine-grained image feature extractor and thereby perceives subtle visual cues better.

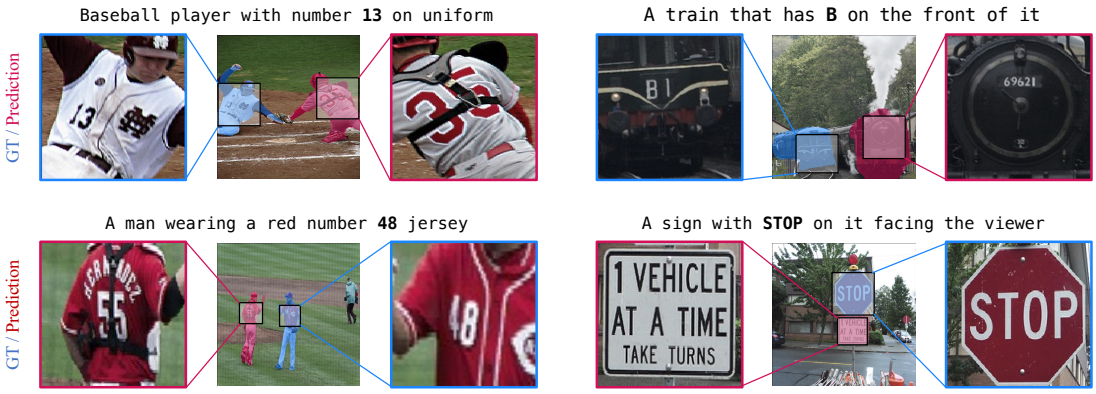

Figure A1: Visualization of RISCLIP-B predictions on RefCOCOg-UMD (Nagaraja et al., 2016) test set samples. RISCLIP fails to recognize alphabetic and numeric characters.

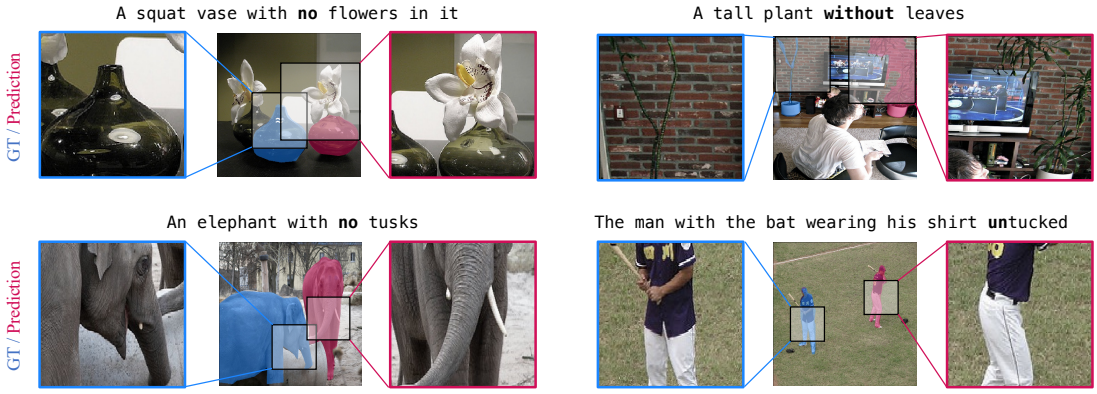

Figure A2: Visualization of RISCLIP-B predictions on RefCOCOg-UMD (Nagaraja et al., 2016) test set samples. RISCLIP fails to comprehend texts that describe the target object with the 'absence' of some attribute.

Figure A3: Visualization of RISCLIP-B predictions on RefCOCOg-UMD (Nagaraja et al., 2016) test set samples. 'L' denotes the text of the left subfigure whilst 'R' denotes that of the right. RISCLIP succeeds in locating different target instances within the same image, even when the texts are long and complex. We also present cases where there are similar instances to the target.

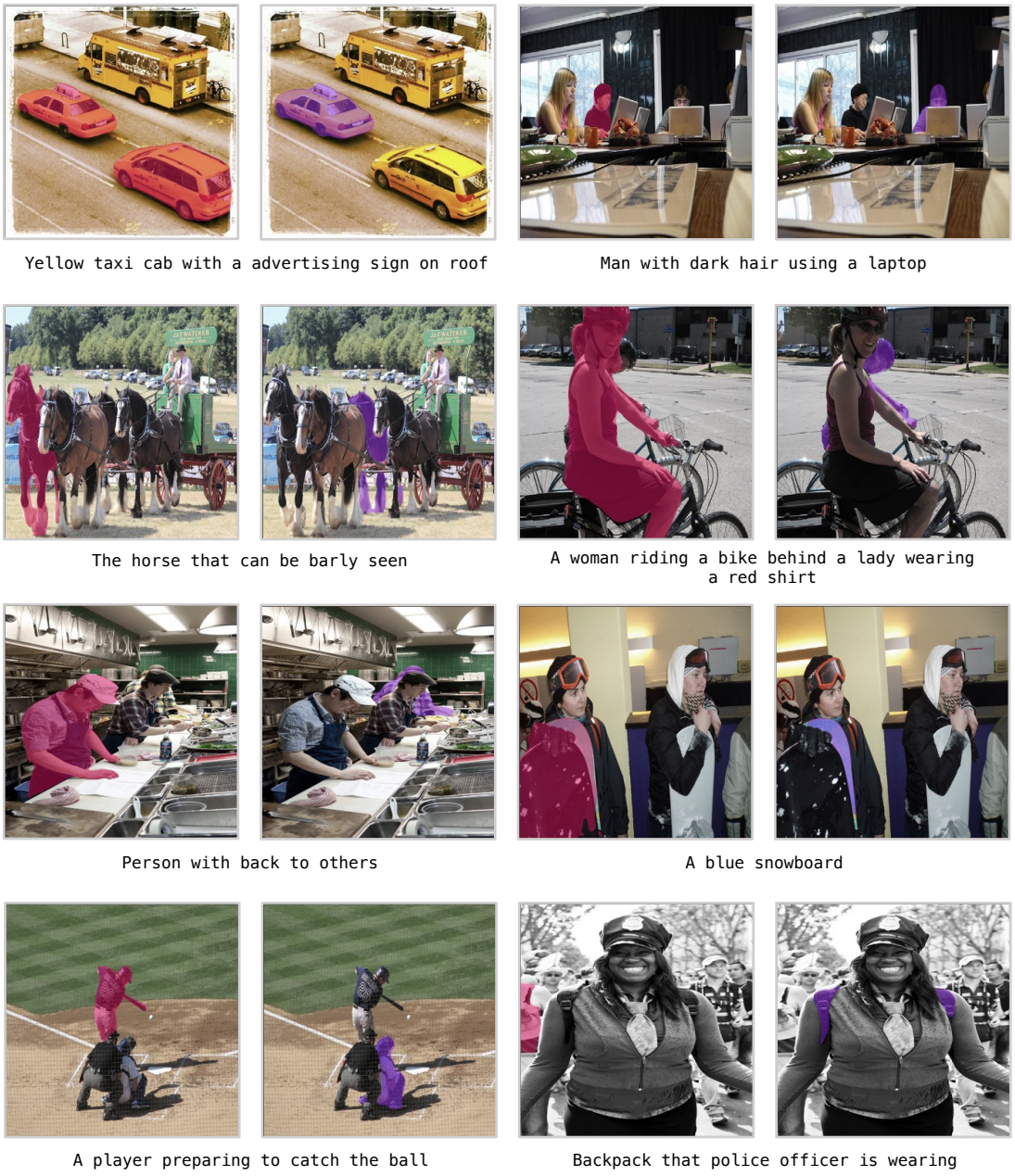

Figure A4: Visualizations of RISCLIP-B (left subfigures in pink) and RISCLIP-L (right subfigures in blue) predictions on RefCOCOg-UMD (Nagaraja et al., 2016) test set samples. RISCLIP-L detects instances that have small detecting cues or that are partially visible which are omit by RISCLIP-B.