SiRi: A Simple Selective Retraining Mechanism for Transformer-based Visual Grounding

Mengxue Qu[1,2⋆], Yu Wu[3], Wu Liu[4], Qiqi Gong[1,2], Xiaodan Liang[5], Olga Russakovsky[3], Yao Zhao[1,2], and Yunchao Wei[1,2]

[1]Institute of Information Science, Beijing Jiaotong University
[2]Beijing Key Laboratory of Advanced Information Science and Network Technology
[3]Princeton University [4]JD Explore Academy [5]Sun Yat-sen University
qumengxue@bjtu.edu.cn, yuwu@princeton.edu, wychao1987@gmail.com

Abstract. In this paper, we investigate how to achieve better visual grounding with modern vision-language transformers, and propose a simple yet powerful **S**elect**i**ve **R**etra**i**ning (SiRi) mechanism for this challenging task. Particularly, SiRi conveys a significant principle to the research of visual grounding, *i.e.*, a better initialized vision-language encoder would help the model converge to a better local minimum, advancing the performance accordingly. In specific, we continually update the parameters of the encoder as the training goes on, while periodically re-initialize rest of the parameters to compel the model to be better optimized based on an enhanced encoder. SiRi can significantly outperform previous approaches on three popular benchmarks. Specifically, our method achieves 83.04% Top1 accuracy on RefCOCO+ *testA*, outperforming the state-of-the-art approaches (training from scratch) by more than 10.21%. Additionally, we reveal that SiRi performs surprisingly superior even with limited training data. We also extend it to transformer-based visual grounding models and other vision-language tasks to verify the validity. Code is available at https://github.com/qumengxue/siri-vg.git.

Keywords: Visual grounding, Transformer, Generalization

1 Introduction

Visual grounding [51, 32], also known as Referring Expression Comprehension (REC), aims to predict the location of a region referred to by the language expression in an image. Previous solutions can be roughly divided into two-stage methods [16, 17, 27, 41, 42, 44, 50, 52, 55] and one-stage methods [3, 26, 34, 46, 48]. The two-stage methods start with the process of generating region proposals via object detectors [9] and then learning to identify the expected object from hundreds of candidates. On the other hand, the one-stage methods perform the grounding in an end-to-end manner, and often with inferior performances. However, the performance of these models is significantly limited due to the

⋆ Work done during an internship at JD Explore Academy.

huge semantic gap between diverse referring descriptions and various visual appearances. The reason is that visual grounding needs to consider many open or fine-grained (*e.g.*, girl, boy, child) categories, which is significantly different from the common vision tasks (*e.g.*, classification, detection, and segmentation) where each image or individual object has a clear class label. Therefore, due to the diversity of descriptions in the human world, the model may easily overfit the descriptions in *train* while hard to correctly understand the referring expressions in *val* and *test* when the training data is insufficient.

Recently, many researchers focus on using the attention mechanism in Transformer for Vision-Language (V-L) modeling [38, 30, 6, 21]. With both visual and linguistic elements as the inputs, the Transformer encoder can perceive multimodal data and thoroughly model the visual-linguistic relationship. Although these Transformer-based methods have achieved great success in vision-language modeling, they heavily rely on pre-training with extra large-scale vision-language data pairs to improve the generalization ability of the encoder and relieve the over-fitting issue, accordingly.

However, without large-scale data pre-training, the model shows significant performance degradation on visual grounding tasks. We observe that the relationship between the given expression and the image perceived by the Transformer encoder leaves much to be desired based on the poor V-L interaction attention map in Fig. 1. The reason may be that the Transformer encoder, started with randomly initialized parameters, may easily over-fit a small number of training pairs and make the model be trapped into a poor local minimum. With such an observation, we raise the question of *whether the V-L model will converge to a better local minimum by equipping the Transformer encoder with better-initialized parameters?*

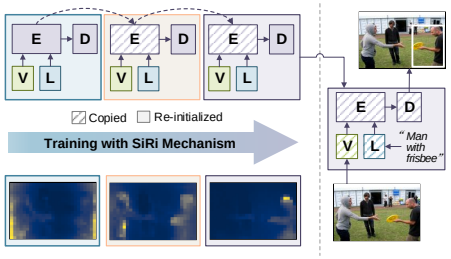

Fig. 1. The sketch of our SiRi mechanism of three retraining periods. "V": Visual Backbone, "L": Language Backbone, "E": Visual-Language Transformer Encoder, "D": Transformer Decoder. The right part shows that we only take the last retrained model for the final test. Best viewed in color.

To answer the above question, in this paper, we investigate a new training mechanism to improve the Transformer encoder, named **S**elect**i**ve **R**etra**i**ning (SiRi), which repeatedly reactivates the learning of the encoder in the process of continuous retraining and progressively provide better-initialized parameters for the encoder in the next stage. Specifically, while we *continually update* parameters of the encoder as the training goes on, we *periodically re-initialize* all the other modules (*e.g.*, vision/language backbones and the Transformer decoder). In this way, the SiRi promotes the encoder to continually learn better vision-language relationships by periodically getting out of the sub-optimal

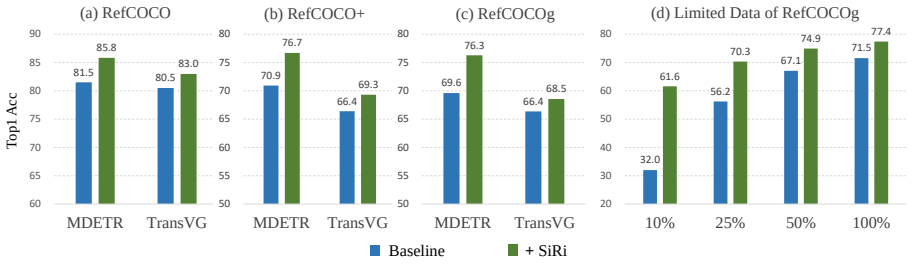

Fig. 2. (a)-(c) illustrates the performance enhancement of SiRi on MDETR [21] and TransVG [6]. We test on three popular visual grounding datasets RefCOCO, RefCOCO+, RefCOCOg. (d) shows that when training with 10%, 25%, 50%, 100% *training* data, the top1 accuracy improvement of SiRi on the RefCOCOg *validation* set.

saddle point. Fig. 1 shows the sketch of SiRi and the visualization of the encoder's attention weight after each retraining period, where we can clearly see the progress of the encoder in multi-modal modeling.

We conduct extensive experiments to validate the effectiveness of our method. With the proposed SiRi mechanism, our model remarkably outperforms previous approaches on three popular benchmarks. Particularly, we achieve 83.04% at top-1 accuracy on RefCOCO+ *testA* [51], outperforming the state-of-the-art approaches by more than 10.21%.

More importantly, we further observe that the SiRi mechanism helps model generalize well to small-scale training data as shown in Fig. 2 (d). To be specific, our model with a quarter of training data outperforms previous state-of-the-art methods (with full training data) by 1.65% on the RefCOCOg *val* set. With even less training data (*e.g.*, only 10%), we almost double the accuracy (61.58% *versus* 32.00%) compared to the baseline. Additionally, we complement more extensibility studies in other visual grounding model and other V-L tasks related to visual grounding. We found SiRi can further improve the top-1 accuracy by an average of 2% in TransVG [6], which is also a Transformer-based visual grounding model. We visualize the improvement of different model with SiRi on three datasets in Fig. 2 (a) - (c). In other V-L tasks, including referring expression segmentation, phrase grounding, and visual question answering tasks, we can also improve the baseline using the SiRi mechanism.

2 Related Work

2.1 Visual Grounding

Existing methods for Visual Grounding based on CNN can be roughly divided into two categories, namely two-stage methods and one-stage methods.

Two-stage methods [16, 17, 24, 25, 27, 41, 42, 43, 44, 50, 52, 55] typically utilize an object detector to generate region proposals in the first stage, and then

find the best matched region-text pair. The object-text pair matching is commonly used in visual grounding task and other V-L tasks, *e.g.*, retrieval tasks [54]. MattNet [50] takes a modular approach to progressively understand and unify visual and linguistic semantic information in terms of attributes, relationships, and location. Additionally, some approaches further enhance the modeling ability of multi-modal relations using graph structures [42, 44, 45], multi-modal tree structures [27].

One-stage methods [3, 26, 34, 46, 48] avoid being constrained by the quality of the proposal by directly fusing visual and linguistic features. FAOA [48] represents the text input with a language vector and leverages it into the YOLOv3 detector [33] to align the referred instance. RCCF [26] regards the visual grounding problem as a correlation filtering process [1, 14], and the peak value in the correlation heatmap is selected as the center of target objects. In ReSC [46], the limitation of FAOA [48] on grounding complex queries is broken through with a recursive sub-query construction module.

In the previous CNN-based visual grounding model, the V-L fusion is performed throughout the decoding process, which is weak interpretability and performance compared to the V-L fusion module in Transformer-based model. Therefore, we adopt Transformer-based model for better V-L interaction.

2.2 Transformer-based Methods in REC

Recently, Transformer [40] has been widely used to address the multi-modal semantic alignment problem. However, Transformer is data-hungry and thus usually needs additional large-scale pretraining. Motivated by the excellent performance of BERT [7], some researchers [38, 4, 30, 49, 8, 22, 39] construct similar structures and propose multi-modal pre-training for Visual-Language Pretraining (VLP) tasks. These approaches introduce pretext tasks for better interaction of vision and language, *e.g.*, masked language modeling [30, 38], image-text matching [22]. However, these VLP methods usually require pre-training with large-scale data and fine-tuning on downstream tasks to achieve good results. Recently, TransVG [6] study the Transformer-based framework without pretraining. Without extracting region proposals in advance, TransVG directly regresses bounding box coordinates and predicts the referring objects.

These works have validated the effectiveness of Transformer for multimodal modeling. However, most of them require large-scale data to pretrain a Transformer-based model. Differently, in this work, we focus on exploring a way to train better encoders *without* large-scale pretraining.

2.3 Re-training

Some early works avoid getting trapped in a local minimum by introducing randomness. For example, ensemble learning [12, 23] introduces randomness by retraining the model with different random initialized parameters to converge to different local minimums. Due to these studies requiring an overwhelming cost, a number of retraining methods, *e.g.*, Dropout [37], Distillation [15], are

proposed to reduce the cost of retraining in ensemble learning. More recently, Snapshot Ensemble [18] proposes to retrain the same model to access multiple local minimums by the cyclic learning rate. Similarly, the cyclic learning rate is used in the retraining process to detect noisy labels in O2U-Net [19]. However, Transformer [40] is very sensitive to the learning rate and sometimes requires a warm-up or inverse square root learning rate, which makes the cyclic learning rate [36] inapplicable. The proposed weight initialization scheme T-Fixup in [20] enables Transformer training without warmup or layer normalization. Han *et al.* [11] proposes DSD retraining mechanism with reference to the model pruning, which avoids over-fitting caused by over-capturing of noisy data.

The SiRi mechanism proposed in this paper is somehow similar to the above methods but SiRi is designed for the V-L fusion module in V-L tasks. The main motivation of re-training in this paper is to provide the V-L fusion Transformer with better-initialized parameters.

3 Method

In this section, we first briefly review the basic visual grounding architecture adopted by this work in Sec. 3.1. Then we elaborate on our proposed SiRi mechanism in Sec. 3.2 and the Multi-task SiRi in Sec. 3.3.

3.1 Base Architecture

We follow the state-of-the-art model MDETR [21] as our base architecture, which consists of four main modules: (1) Visual Backbone; (2) Language Backbone; (3) Visual-Language Transformer Encoder; (4) Transformer Decoder Module.

Visual Backbone $\mathcal{V}$ & Language Backbone $\mathcal{L}$. We adopt the convolutional backbone ResNet-101 [13] to obtain the visual representation for an input image **I**. In previous work MDETR [21], they only take the output of the last CNN stage as visual features. Differently, we believe the features of shallow stages (*e.g.*, the third stage in ResNet-101) benefit localizing objects if the sentence contains a detailed low-level description such as color. Therefore, we take the output of the third stage of ResNet-101 and transform it with two dilated convolution layers. Then we add the adjusted dimensionality low-level feature together using the final-stage output of ResNet-101 as the final visual representations. Then we encode referring expressions with the pretrained language model RoBERTa [28].

Visual-Language Transformer Encoder $\mathcal{E}$. We use a Transformer [40] as the encoder for vision-language interaction, where the model performs the cross-modal fusion and association. To do so, we flatten the visual features and add 2-D positional embeddings to conserve spatial information. After that, we project both the flattened visual features and text features into a shared embedding space and then concatenate them into a single sequence of image and text features. The sequence is then input to the cross encoder Transformer for further visual-language interaction.

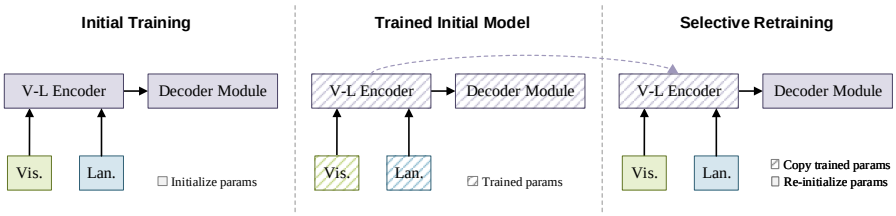

Fig. 3. The training process of our SiRi mechanism. The parameters of the module with solid color background are initialized as the original rules, while those with slash background are trained. The base architecture contains four main modules: (1) "Vis.": Visual Backbone; (2) "Lan.": Language Backbone; (3) "V-L Encoder": Visual-Language Transformer Encoder; (4) "Decoder Module": Transformer Decoder Module.

Transformer Decoder $\mathcal{D}$. Following DETR [2], we use a Transformer decoder to predict the target bounding boxes. The decoder takes as input a set of learnable object queries, cross-attends to the encoder output and predicts embeddings for each query. After that, we decode the embeddings into box coordinates and class labels by the regression and classification heads. Considering that the number of relevant referred targets is fewer than the total number of objects of an image, we limit the decoder to have 16 query inputs only. Considering there is only sentence-level correspondence in visual grounding, we remove box-token contrastive alignment loss [21]. Accordingly, we also reduce the length of the soft tokens to 2, standing for whether the object box belongs to the expression.

3.2 SiRi: Selective Retraining Mechanism

The transformer model may easily get over-fitted without large-scaled pre-training. As shown in Fig. 4, the test loss increases even though the training loss still declines after point A of the initial training stage. Simply having more training iterations would not further improve the test performance.

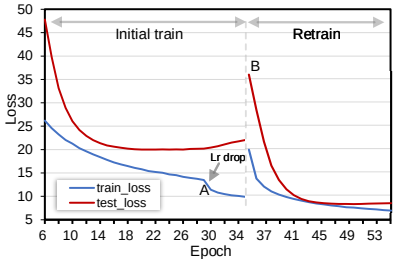

Fig. 4. The train and test loss curves in *Initial train* stage and *Retrain* stage.

Motivated by our hypothesis that a V-L model may converge to a better local minimum by equipping the Transformer encoder with better initialized parameters, we design the Selective Retraining (SiRi) mechanism. After the initial training, we continually update the parameters of the encoder as the training goes on, while periodically re-initializing the parameters of the decoder to compel the model to be

better optimized based on an enhanced encoder. By applying our SiRi mechanism at point B in Fig. 4, both training loss and test loss further decline, thus we obtain better optimization results (lower test loss). To be specific, our Selective Retraining Mechanism is set up as follows.

Initial Training. We initialize the visual Backbone $\mathcal{V}$ and the language Backbone $\mathcal{L}$ using the ResNet-101 [13] model pre-trained from ImageNet [5] and the RoBERTa model pre-trained from language corpus datasets, respectively.

The rest of our model (*e.g.*, Transformer encoder and decoder) are randomly initialized using the Xavier initialization [10]. We denote the initialized parameters of the Visual Backbone together with the visual linear projection layer as $\mathcal{V}_0$, and Language Backbone together with the corresponding linear projection layer as $\mathcal{L}_0$. Similarly, the model weights of Transformer Encoder and Transformer Decoder are denoted as $\mathcal{E}_0$ and $\mathcal{D}_0$, respectively. We then train the model using a combination of the object coordinates regression losses (L1 & GIoU) and soft-token prediction loss (cross-entropy loss) while keeping the learning rate unchanged. The model training stops when the validation performance stays stable. We denote the trained model weights to be $\mathcal{V}_0^{'}, \mathcal{L}_0^{'}, \mathcal{E}_0^{'}, \mathcal{D}_0^{'}$ after the initial training.

Selective Retraining. To further improve the encoder with better vision-language understanding, we continually train the encoder after the initial training, while *re-initialize* the other modules to avoid getting stuck in local minimums. We show the pipeline of SiRi in Fig. 3. Specifically, for the t-th round of the selective retraining, we only keep the encoder $\mathcal{E}_t$ to be up-to-date, *i.e.*, $\mathcal{E}_t \leftarrow \mathcal{E}_{t-1}^{'}$, where $\mathcal{E}_{t-1}^{'}$ is the previous trained encoder from $t-1$ round. As for other modules including the decoder $\mathcal{D}_t$, the visual backbone $\mathcal{V}_t$, and the language backbone $\mathcal{L}$, we drop the trained weights and re-initialize them using their original initialization at the initial training stage, *i.e.*, either initializing from the pre-trained weights (*e.g.*, $\mathcal{V}_0$ and $\mathcal{L}_0$), or random initialization (*e.g.*, the decoder D). We then re-train the whole model using the same learning rate until it converges.

3.3 Multi-task SiRi

As a common practice for transformer models, multi-task learning usually benefits the model optimization and thus alleviates over-fitting issues. Therefore, we further extend SiRi to a multi-task version by incorporating an auxiliary decoder. Specifically, we use two diverse decoders to generate predictions based on the same encoder output and then optimize the encoder using the two decoder losses.

To ensure the two decoders are different from each other, we design two different object queries (positional embeddings) for decoders. Previous DETR [2] uses *learnable* positional embeddings as the object query to attend to the encoder output. Differently, we adopt a *constant* positional encoding sequence, *i.e.*, the sine-cosine position encoding function, to generate the object queries for the other decoder. The two decoders take different queries to attend to the same

encoder output, which would urge the encoder to be more robust in vision-language interaction. The details are shown in the supplementary materials.

4 Experiments

4.1 Datasets

RefCOCO/RefCOCO+ are proposed in [51]. There are 19,994 images in RefCOCO with 142,209 refer expressions for 50,000 objects. Similarly, 19,992 images are included in RefCOCO+ which contains 141,564 expressions for 49,856 objects. In these datasets, each image contains two or more objects from the same category. In RefCOCO+ dataset, positional words are not allowed in the referring expression, which is a pure dataset with appearance-based referring expression, whereas RefCOCO imposes no restriction on the phrase. In addition to the training set and validation set, the test set for RefCOCO/RefCOCO+ is divided into a *testA* set (containing several people in an image) and a *testB* set (containing multiple instances of other objects in an image).

RefCOCOg [32] contains 26,711 images with 85,474 referring expressions for 54,822 objects, and each image usually contains 2-4 objects of the same category. The length of referring expressions in this dataset is almost twice as long as those in RefCOCO and RefCOCO+.

4.2 Experimental Settings

Implementation Details. Following MDETR [21], all parameters in the network are optimized using AdamW [29] with the learning rate warm-up strategy. The model is trained using 4 GPUs with a batch size of 72. We set the learning rate of the language backbone RoBERTa [28] to be 1×10^{-5}, and all the rest parameters to be 5×10^{-5}. In initial training, the model with a single decoder is trained for 55 epochs, and the model with a dual decoder (multi-task SiRi) is trained for 35 epochs since it converges quickly. Each retraining stage takes another 30 training epochs. We set the maximum side length of the input image as 640 while keeping the original aspect ratio. Images in the same batch are padded with zeros until acquiring the largest size of that batch. Similarly, sentences in one batch will be adjusted to the same length as well. We continually retrain the model until the validation performance converges (usually 5 to 8 rounds).

Evaluation Metrics. Following the proposal setting in the previous work, we use the metric Prec@0.5 to evaluate our method, where a predicted region will be regarded as a positive sample if its intersection over union (IoU) with the ground-truth bounding box is greater than 0.5.

4.3 Comparison with State-of-the-art Methods

We compare our method with other state-of-the-art methods on three common benchmarks of Referring Expression Comprehension, *i.e.*, RefCOCO, Re-

Table 1. Comparisons with state-of-the-art methods on RefCOCO [51], RefCOCO+ [51], and RefCOCOg [32] in terms of top-1 accuracy. We also report official MDETR implementation [21] without pretraining (denoted as MDETR w/o pretrain) and our improved MDETR implementation (see Sec. 3.1) (denoted as MDETR*). "MT SiRi" means "Multi-task SiRi".

Method	Venue	Visual backbone	RefCOCO			RefCOCO+			RefCOCOg	
			val	testA	testB	val	testA	testB	val	test
CNN-based:										
CMN [17]	CVPR'17	VGG16 [35]	-	71.03	65.77	-	54.32	47.76	-	-
MAttNet [50]	CVPR'18	ResNet-101 [13]	76.65	81.14	69.99	65.33	71.62	56.02	66.58	67.27
RvG-Tree [16]	TPAMI'19	ResNet-101	75.06	78.61	69.85	63.51	67.45	56.66	66.95	66.51
NMTree [27]	ICCV'19	ResNet-101	76.41	81.21	70.09	66.46	76.02	57.52	65.87	66.44
FAOA [48]	ICCV'19	DarkNet-53 [33]	72.54	74.35	68.50	56.81	60.23	49.60	61.33	60.36
RCCF [26]	CVPR'20	DLA-34 [53]	-	81.06	71.85	-	70.35	56.32	-	65.73
MCN [31]	CVPR'20	DarkNet-53	80.08	82.29	74.98	67.16	72.86	57.31	66.46	66.01
ReSC-Large [46]	ECCV'20	DarkNet-53	77.63	80.45	72.30	63.59	68.36	56.81	67.30	67.20
Transformer-based Pretrained:										
ViLBERT [30]	NeurIPS'19	ResNet-101	-	-	-	72.34	78.52	62.61	-	-
ERNIE-ViL [49]	AAAI'20	ResNet-101	-	-	-	75.95	82.07	66.88	-	-
UNTIER [4]	ECCV'20	ResNet-101	81.41	87.04	74.17	75.90	81.45	66.70	74.86	75.77
VILLA [8]	NeurIPS'20	ResNet-101	82.39	87.48	74.84	76.17	81.54	66.84	76.18	76.71
MDETR [21]	ICCV'21	ResNet-101	86.75	89.58	81.41	79.52	84.09	70.62	81.64	80.89
Transformer-based without Pretrained:										
TransVG [6]	ICCV'21	ResNet-101	81.02	82.72	78.35	64.82	70.70	56.94	68.67	67.73
MDETR (w/o pretrain)	ICCV'21	ResNet-101	78.01	82.18	72.56	68.01	72.83	55.57	65.54	65.99
MDETR*	-	ResNet-101	81.49	84.67	76.58	70.93	75.65	59.27	69.59	70.22
MDETR* + SiRi	-	ResNet-101	**85.83**	88.56	**81.27**	76.68	82.01	66.33	76.63	76.46
MDETR* + MT SiRi	-	ResNet-101	85.82	**89.11**	81.08	**77.47**	**83.04**	**67.11**	**77.39**	**76.80**

fCOCO+, and RefCOCOg. Results are reported in Table 1. Our method displays significant improvement over previous methods on all three datasets. Compared to models without large-scale pretraining, which is a fair comparison, we outperform them by more than 6.39% on RefCOCO@testA, 10.21% on RefCOCO+@testA, and 9.07% on RefCOCOg@test. Even compared to those large-scaled pretrained models, *e.g.*, MDETR pretrained using more than one million aligned image-text pairs, our method still achieves comparable results on RefCOCO without those extra data.

4.4 Ablation Studies

Different Retraining Module. Besides continually updating the encoder while periodically re-initializing all the other parts, we also evaluate different re-initializing modules.

We show eight variants of our SiRi Mechanism in Fig. 5, For a fair comparison, we keep all hyperparameters the same and retrain these variants from the same initial trained model. We show their correspondence results after the first retraining in Table 2. The encoder with better initialized parameters is the critical factor for the whole model converging to a better local minimum.

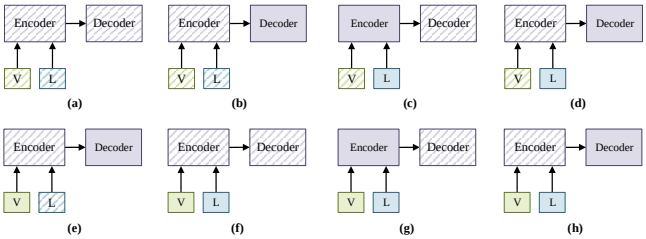

Fig. 5. Schematic of the eight retraining variants with different combinations of selective modules. The solid color background means re-initializing parameters, while the slash background means continually updated parameters from previous periods. Best viewed in color.

Mode	RefCOCO+@val
Initial	71.45
(a) V,L,E,D	71.98
(b) V,L,E	72.12
(c) V,D	72.14
(d) V,E	73.25
(e) L,E	73.44
(f) E,D	73.80
(g) D	72.76
(h) E	**74.14**

Table 2. Performance comparison of different selective modules. The eight mode are shown in Fig. 5.

Comparing mode (d) with mode (h), we find that re-initializing the *visual* backbone has great impact on performance boosting, which verifies our motivation that re-initializing the input of encoder helps to get out of local minimums while keeping the essential cross-modeling ability of previous models. Similar results can be found for *language* backbone by comparing mode (e) with mode (h). Interestingly, we find that the performance is competitive to Mode (h) when we use Mode (f), where we keep the parameters of both encoder and decoder. For simplicity, we only keep the encoder updated continually in all the other experiments.

Retraining Periods. In Fig. 6, we show the validation performance curves during selective retraining. Zero indicates the initial trained model in the figure. We can see the model performance increases a lot in the first three retraining periods and then tends to converge after several retraining periods. The highest performances are achieved in the fifth retraining period, where SiRi outperforms the initial trained model by 5.18% (72.29% versus 77.47%) and 5.86% (71.53% versus 77.39%) on RefCOCO+ and RefCOCOg, respectively.

Different Object Queries in Multi-task SiRi. We can also see the consistent performance gap between the single SiRi and the multi-task SiRi in Fig. 6. The multi-task SiRi always performs better than single SiRi during all the retraining periods. We further study the impact of different object queries (*e.g.*, learnable queries and constant queries) used in Multi-task SiRi. The results of the initial trained models using different quires in multi-task learning are shown in Table 3.

Although learnable and constant object queries achieve similar results for single task training, the combination of them in multi-task learning achieves higher performance (72.29% *versus* 70.93% on RefCOCO+). Note that multi-task structure with two identical object query types (*e.g.*, both learnable or both constant) does not outperform single task learning. It indicates that taking different queries to attend the same encoder output may help the encoder to be more robust on vision-language interaction.

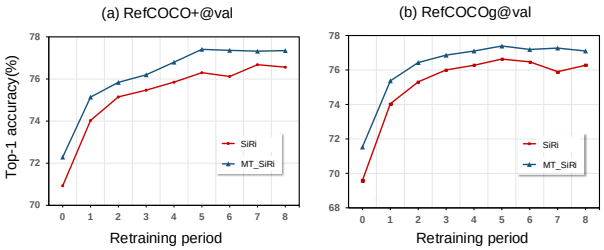

Fig. 6. Performance achieved by increasing the training periods. The blue line indicates the single SiRi model and the red line indicates the multi-task SiRi model. "MT" indicates multi-task.

Structure	Object Queries *1st* Dec.	Object Queries *2nd* Dec.	RefCOCO+
Single-task	L	–	70.93
	C	–	70.72
Multi-task	L	L	70.27
	C	C	71.24
	L	C	**72.29**

Table 3. Ablation studies on different object query types in multi-task SiRi. ("L": learnable queries, and "C": constant queries, "Dec.": Decoder.)

4.5 Qualitative Results

We visualize the attention weight of encoders along with the retraining progress in Fig. 7. To be specific, we calculate the cross-modal attention weights (vision output tokens based on language input tokens) from the last layer of the Transformer encoder, and then visualize them in the original image size. We believe the values of cross-modal attention weights indicate the encoder's ability of vision-language understanding.

We show two test samples in the figure with the corresponding input sentences. From left to right, we show the bounding box predictions together with the attention maps generated by the initial trained, 1st, 3rd, 5th, and 7th retrained encoders, respectively. It can be intuitively seen that the encode learns to better perceive the relationship between expressions and images as the continuous SiRi training goes. Taking the upper sample as an example, the predicted bounding box is incorrect from the initial trained model, where we can see the attention map of the first encoder does not highlight the referred object, either.

After selective retraining, the encoder gets better and better, which can be seen from the more accurate attention maps. Therefore, the predicted boxes are also better than the initial ones. It validates our motivation that the better encoder initialization helps the model converge to a better local minimum. Continually updating the encoder while periodically re-initializing other modules can strengthen the visual-linguistic modeling.

4.6 Extensibility Studies

To better show the generality, we further extend SiRi to more visual grounding settings, models, and tasks.

Extend to Small Data Size. First, we study how SiRi performs with fewer training data, where the over-fitting issue is more severe. To do so, we randomly sample 10%, 25%, and 50% of training data from the RefCOCOg training set as the new training splits, respectively. Then we train the model following the

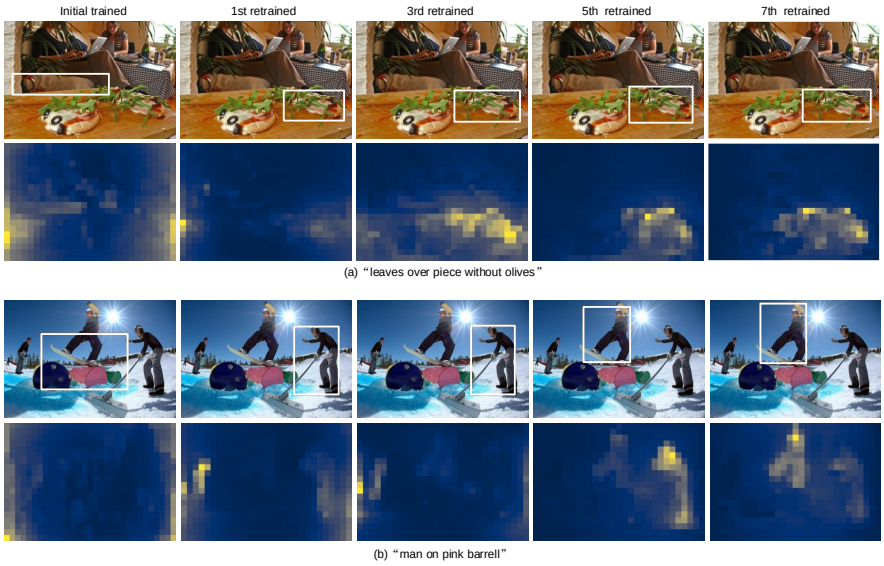

Fig. 7. Visualization of the predicted box and the encoder's cross-modal attention weights in inference. The columns represent initial trained, 1st retrained, 3rd retrained, 5th retrained, 7th retrained model, respectively, from left to right. As we can see, the model prediction gets better as the encoder attention map gets clear.

SiRi mechanism[1] and then evaluate the performance on the full validation set of RefCOCOg (the same validation set for all). The results are shown in Fig. 8. Compared with the initial trained model, our SiRi model shows very impressive performance gains, *e.g.*, almost doubling the performance at 10% sampling rate.

As can be seen from the figure, the performance is improved much more significantly when employing the SiRi mechanism on fewer training data, which verifies that our SiRi can generalize the vision-language encoder and avoid over-fitting. It suggests that our SiRi mechanism may be potentially treated as a strong alternative to large-scale pre-training models.

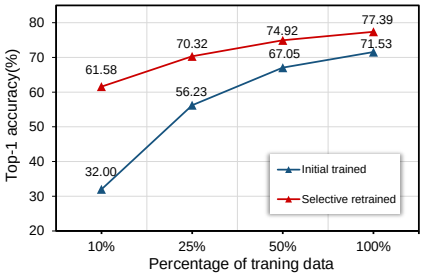

Fig. 8. Performance improvement of the model with SiRi with limited training samples. We randomly sample 10%, 25%, 50% of training data from RefCOCOg and train with SiRi. All models are evaluated on the same RefCOCOg *val* set.

Extend to other V-L models. The application of SiRi mechanism on

[1] We train more epochs until converging in small-scale experiments.

Table 4. REC and phrase grounding results of TransVG [6] with SiRi mechanism.

Model	Backbone	Referring Expression Comprehension									PhraseGround	
		RefCOCO			RefCOCO+			RefCOCOg	ReferIt		Flickr30k	
		val	testA	testB	val	testA	testB	g-val	val	test	val	test
TransVG	ResNet-50	80.49	83.28	75.24	66.39	70.55	57.66	66.35	71.60	69.76	77.19	78.47
+SiRi	ResNet-50	**82.97**	**84.42**	**79.04**	**69.30**	**73.27**	**59.93**	**68.54**	**74.28**	**71.36**	**77.99**	**79.17**

Table 5. Referring Expression Segmentation results of LAVT [47] with SiRi.

RefCOCO+	Model	P@0.5	P@0.6	P@0.7	P@0.8	P@0.9	oIoU	mIoU
val	LAVT	74.44	70.91	65.58	56.34	30.23	62.14	65.81
	+SiRi	**75.56**	**72.39**	**67.88**	**58.33**	**30.79**	**62.86**	**66.78**
testA	LAVT	80.68	77.96	72.90	62.21	32.36	68.38	70.97
	+SiRi	**82.20**	**79.18**	**74.54**	**63.99**	**32.62**	**68.87**	**71.93**
testB	LAVT	65.66	61.85	55.94	47.56	27.24	**55.10**	59.23
	+SiRi	**66.41**	**62.86**	**57.37**	**49.23**	**27.90**	55.03	**59.70**

other V-L models can be achieved by simply following the principle: keeping the parameters of V-L fusion module continuously training, while reinitializing the other parts. We applied our SiRi to Transformer-based Visual Grounding model TransVG [6] and RES model LAVT [47]. Experimental details are presented in the supplementary materials. For TransVG [6], we report REC and Phrase Grounding results in Table 4. We found that SiRi could further improve the performance of TransVG by an average of 2% at top-1 accuracy on all four REC datasets, and the performance has also been effectively improved on Phrase Grounding dataset Flickr30k dataset. For LAVT [47], We report the results of SiRi in RES dataset RefCOCO+ three splits *val*, *testA*, *testB* in Table 5.

Extend to other V-L tasks. We also test our SiRi in more vision-language tasks, including referring expression segmentation, phrase grounding, and visual question answering. For these experiments, we took the transformer-based MDETR model (without pre-training) as our baseline. The specific settings of how to apply SiRi on these tasks are stated as follows.

-Referring Expression Segmentation (RES). RES is to segment the objects according to the given language description. We further perform the segmentation task on the trained visual grounding model. We keep the original MDETR model architecture the same but modify the hyperparameters according to the settings used in training visual grounding in this paper. We test the SiRi model on three RES datasets, *i.e.*, RefCOCO, RefCOCO+, RefCOCOg. In Table 6, we report the RES performance of the SiRi model after *Initial-train*, *3rd-train*, and *5th-train* stages. It can be seen that SiRi can steadily improve RES models during the retraining process.

-Phrase Grounding. The task is to locate objects in an image based on the phrases which may be inter-related. We evaluate the SiRi mechanism on the Flickr30k entities dataset. For the input image, we set the maximum size to 800. We show the model performance of different SiRi stages in Table 7. We

Table 6. Experiment results on RES. We report precision Pr@0.5, 0.7, 0.9 and overall IoU on the *val* set of RefCOCO, RefCOCO+, RefCOCO.

Stage	RefCOCO				RefCOCO+				RefCOCOg			
	Pr@0.5	Pr@0.7	Pr@0.9	oIoU	Pr@0.5	Pr@0.7	Pr@0.9	oIoU	Pr@0.5	Pr@0.7	Pr@0.9	oIoU
Initial-train	77.76	68.89	28.58	62.12	68.36	61.11	25.89	52.48	64.34	54.84	20.42	51.39
3rd-retrain	82.58	74.33	32.57	68.02	75.27	67.76	28.21	60.11	72.20	61.46	25.12	58.33
5th-retrain	**83.56**	**75.37**	**32.79**	**69.34**	**76.46**	**68.47**	**28.26**	**61.15**	**73.24**	**63.25**	**25.08**	**59.69**

Table 7. Experiment results of Phrase Grounding on the validation set of Flickr30k and the VQA performance on the GQA *balance test* set.

Stage	Phrase Grounding@Flickr30k			GQA
	R@1	R@5	R@10	Accuracy
Initial-train	76.22	87.19	90.26	55.75
1st-retrain	78.41	88.42	91.31	56.38
2nd-retrain	**78.63**	**88.62**	**91.62**	**57.25**

can see SiRi further improves the initial trained model by 1%∼2% on Recall@1, Recall@5, Recall@10 (denoted as R@1, R@5, R@10, respectively).

-Visual Question Answering. Given an image and a question in natural language, this task is to infer the correct answer. We use the scene graph provided in GQA to align question words and the boxes as in MDETR. We verify the validity of SiRi on the visual question answering task in GQA *balanced* split dataset. The results of SiRi model from different training stages are reported in Table 7. The accuracy is improved from 55.75 to 57.45.

5 Conclusion

In this paper, we present a novel training mechanism namely Selective Retraining (SiRi) for visual grounding, where we keep updating the Transformer encoder while re-initialize the other modules to get out of local minimums. We further propose multi-task SiRi to train a better encoder by incorporating an auxiliary decoder with constant input queries. Extensive experiments prove our method helps the Transformer encoder better perceive the relationship between the visual and the corresponding expression, outperforming state-of-the-art methods on the three visual grounding datasets. Interestingly, we find SiRi also performs superior even with very limited training data. Even with a quarter of training data, we outperform state-of-the-art methods (with full training data) by 1.65% on the RefCOCOg validation set. We also extend SiRi to other Transformer-based visual grounding models and other V-L tasks. We hope our work will help motivate more researchers in the V-L research community in the future.

Acknowledgements. This work was supported in part by the National Key R&D Program of China (No.2021ZD0112100), the National NSF of China (No.U1936212, No.62120106009), the Fundamental Research Funds for the Central Universities (No.K22RC00010). We thank Princeton Visual AI Lab members (Dora Zhao, Jihoon Chung, and others) for their helpful suggestions.

References

1. David S Bolme, J Ross Beveridge, Bruce A Draper, and Yui Man Lui. Visual Object Tracking Using Adaptive Correlation Filters. In *CVPR*, 2010.
2. Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-End Object Detection with Transformers. In *ECCV*, 2020.
3. Xinpeng Chen, Lin Ma, Jingyuan Chen, Zequn Jie, Wei Liu, and Jiebo Luo. Real-time Referring Expression Comprehension by Single-stage Grounding Network. *arXiv preprint arXiv:1812.03426*, 2018.
4. Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. UNITER: Learning Universal Image-Text Representations. 2019.
5. Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR*, 2009.
6. Jiajun Deng, Zhengyuan Yang, Tianlang Chen, Wengang Zhou, and Houqiang Li. TransVG: End-to-End Visual Grounding with Transformers. *ICCV*, 2021.
7. Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv preprint arXiv:1810.04805*, 2018.
8. Zhe Gan, Yen-Chun Chen, Linjie Li, Chen Zhu, Yu Cheng, and Jingjing Liu. Large-Scale Adversarial Training for Vision-and-Language Representation Learning. In *NeruIPS*, 2020.
9. Ross Girshick. Fast R-CNN. In *ICCV*, 2015.
10. Xavier Glorot and Yoshua Bengio. Understanding the Difficulty of Training Deep Feedforward Neural Networks. In *AISTATS*, 2010.
11. Song Han, Jeff Pool, Sharan Narang, Huizi Mao, Enhao Gong, Shijian Tang, Erich Elsen, Peter Vajda, Manohar Paluri, John Tran, et al. DSD: Dense-Sparse-Sense Training for Deep Neural Networks. In *ICLR*, 2017.
12. Lars Kai Hansen and Peter Salamon. Neural Network Ensembles. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1990.
13. Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *CVPR*, 2016.
14. João F Henriques, Rui Caseiro, Pedro Martins, and Jorge Batista. High-Speed Tracking with Kernelized Correlation Filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2014.
15. Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the Knowledge in A Neural Network. *arXiv preprint arXiv:1503.02531*, 2015.
16. Richang Hong, Daqing Liu, Xiaoyu Mo, Xiangnan He, and Hanwang Zhang. Learning to Compose and Reason with Language Tree Structures for Visual Grounding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.
17. Ronghang Hu, Marcus Rohrbach, Jacob Andreas, Trevor Darrell, and Kate Saenko. Modeling Relationships in Referential Expressions with Compositional Modular Networks. In *CVPR*, 2017.
18. Gao Huang, Yixuan Li, Geoff Pleiss, Zhuang Liu, John E Hopcroft, and Kilian Q Weinberger. Snapshot Ensembles: Train 1, Get m for Free. In *ICLR*, 2017.
19. Jinchi Huang, Lie Qu, Rongfei Jia, and Binqiang Zhao. O2U-Net: A Simple Noisy Label Detection Approach for Deep Neural Networks. In *ICCV*, 2019.
20. Xiao Shi Huang and et al. Improving transformer optimization through better initialization. In *ICML*, 2020.
21. Aishwarya Kamath, Mannat Singh, Yann LeCun, Gabriel Synnaeve, Ishan Misra, and Nicolas Carion. MDETR-Modulated Detection for End-to-End Multi-Modal Understanding. In *ICCV*, 2021.

22. Wonjae Kim, Bokyung Son, and Ildoo Kim. ViLT: Vision-and-Language Transformer Without Convolution or Region Supervision. In *ICML*, 2021.
23. Anders Krogh, Jesper Vedelsby, et al. Neural Network Ensembles, Cross Validation, and Active Learning. In *NeruIPS*, 1995.
24. Chen Liang, Yu Wu, Yawei Luo, and Yi Yang. Clawcranenet: Leveraging object-level relation for text-based video segmentation. *arXiv preprint arXiv:2103.10702*, 2021.
25. Chen Liang, Yu Wu, Tianfei Zhou, Wenguan Wang, Zongxin Yang, Yunchao Wei, and Yi Yang. Rethinking cross-modal interaction from a top-down perspective for referring video object segmentation. *arXiv preprint arXiv:2106.01061*, 2021.
26. Yue Liao, Si Liu, Guanbin Li, Fei Wang, Yanjie Chen, Chen Qian, and Bo Li. A Real-time Cross-modality Correlation Filtering Method for Referring Expression Comprehension. In *CVPR*, 2020.
27. Daqing Liu, Hanwang Zhang, Feng Wu, and Zheng-Jun Zha. Learning to Assemble Neural Module Tree Networks for Visual Grounding. In *ICCV*, 2019.
28. Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv preprint arXiv:1907.11692*, 2019.
29. Ilya Loshchilov and Frank Hutter. Fixing Weight Decay Regularization in Adam. 2018.
30. Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks. In *NeruIPS*, 2019.
31. Gen Luo, Yiyi Zhou, Xiaoshuai Sun, Liujuan Cao, Chenglin Wu, Cheng Deng, and Rongrong Ji. Multi-task Collaborative Network for Joint Referring Expression Comprehension and Segmentation. In *CVPR*, 2020.
32. Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. Generation and Comprehension of Unambiguous Object Descriptions. In *CVPR*, 2016.
33. Joseph Redmon and Ali Farhadi. YOLOv3: An Incremental Improvement. *arXiv preprint arXiv:1804.02767*, 2018.
34. Arka Sadhu, Kan Chen, and Ram Nevatia. Zero-Shot Grounding of Objects from Natural Language Queries. In *ICCV*, 2019.
35. Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *ICLR*, 2015.
36. Leslie N. Smith. Cyclical learning rates for training neural networks. In *WACV*, pages 464–472. IEEE Computer Society, 2017.
37. Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *The Journal of Machine Learning Research*, 2014.
38. Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. VL-BERT: Pre-training of Generic Visual-Linguistic Representations. In *ICLR*, 2020.
39. Hao Tan and Mohit Bansal. LXMERT: Learning Cross-Modality Encoder Representations from Transformers. In *EMNLP*, 2019.
40. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is All You Need. In *NeuruIPS*, 2017.
41. Liwei Wang, Yin Li, Jing Huang, and Svetlana Lazebnik. Learning Two-Branch Neural Networks for Image-Text Matching Tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.

42. Peng Wang, Qi Wu, Jiewei Cao, Chunhua Shen, Lianli Gao, and Anton van den Hengel. Neighbourhood Watch: Referring Expression Comprehension via Language-Guided Graph Attention Aetworks. In *CVPR*, 2019.
43. Yu Wu, Lu Jiang, and Yi Yang. Switchable novel object captioner. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2022.
44. Sibei Yang, Guanbin Li, and Yizhou Yu. Dynamic Graph Attention for Referring Expression Comprehension. In *ICCV*, 2019.
45. Sibei Yang, Guanbin Li, and Yizhou Yu. Graph-Structured Referring Expression Reasoning in the Wild. In *CVPR*, 2020.
46. Zhengyuan Yang, Tianlang Chen, Liwei Wang, and Jiebo Luo. Improving One-Stage Visual Grounding by Recursive Sub-Query Construction. In *ECCV*, 2020.
47. Zhao Yang and et al. Lavt: Language-aware vision transformer for referring image segmentation. In *CVPR*, 2022.
48. Zhengyuan Yang, Boqing Gong, Liwei Wang, Wenbing Huang, Dong Yu, and Jiebo Luo. A Fast and Accurate One-Stage Approach to Visual Grounding. In *ICCV*, 2019.
49. Fei Yu, Jiji Tang, Weichong Yin, Yu Sun, Hao Tian, Hua Wu, and Haifeng Wang. ERNRE-ViL: Knowledge Enhanced Vision-Language Representations through Scene Graph. 2020.
50. Licheng Yu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, Mohit Bansal, and Tamara L Berg. MAttNet: Modular Attention Network for Referring Expression Comprehension. In *CVPR*, 2018.
51. Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. Modeling Context in Referring Expressions. In *ECCV*, 2016.
52. Hanwang Zhang, Yulei Niu, and Shih-Fu Chang. Grounding Referring Expressions in Images by Variational Context. In *CVPR*, 2018.
53. Yifu Zhang, Chunyu Wang, Xinggang Wang, Wenjun Zeng, and Wenyu Liu. FairMOT: On the Fairness of Detection and Re-Identification in Multiple Object Tracking. *International Journal of Computer Vision*, 2021.
54. Kecheng Zheng, Wu Liu, Jiawei Liu, Zheng-Jun Zha, and Tao Mei. Hierarchical gumbel attention network for text-based person search. In *ACM Multimedia*, pages 3441–3449. ACM, 2020.
55. Bohan Zhuang, Qi Wu, Chunhua Shen, Ian Reid, and Anton Van Den Hengel. Parallel Attention: A Unified Framework for Visual Object Discovery Through Dialogs and Queries. In *CVPR*, 2018.