SLAN: Self-Locator Aided Network for Vision-Language Understanding

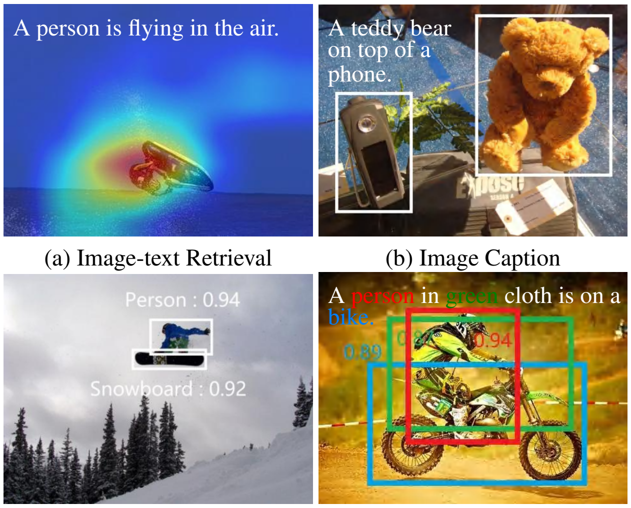
Jiang-Tian Zhai[1]* Qi Zhang[2]* Tong Wu[2] Xing-Yu Chen[2] Jiang-Jiang Liu[1]† Ming-Ming Cheng[1]†
[1]VCIP, CS, Nankai University [2]Tencent Youtu Lab
{jtzhai30,j04.liu}@gmail.com, townswu@tencent.com, cmm@nankai.edu.cn

Abstract

*Learning fine-grained interplay between vision and language contributes to a more accurate understanding for Vision-Language tasks. However, it remains challenging to extract key image regions according to the texts for semantic alignments. Most existing works are either limited by text-agnostic and redundant regions obtained with the frozen region proposal module, or failing to scale further due to their heavy reliance on scarce grounding (gold) data to pre-train detectors. To solve these problems, we propose Self-Locator Aided Network (SLAN) for vision-language understanding tasks without any extra gold data. SLAN consists of a region filter and a region adaptor to localize regions of interest conditioned on different texts. By aggregating vision-language information, the region filter selects key regions and the region adaptor updates their coordinates with text guidance. With detailed region-word alignments, SLAN can be easily generalized to many downstream tasks. It achieves fairly competitive results on five vision-language understanding tasks (*e.g.*, 85.7% and 69.2% on COCO image-to-text and text-to-image retrieval, surpassing previous SOTA methods). SLAN also demonstrates strong zero-shot and fine-tuned transferability to two localization tasks. The code is available at https://github.com/scok30/SLAN.*

1. Introduction

Recent years have witnessed growing interest in exploring relationships between vision and language modalities. A wide range of applications have been boosted by its rapid development, such as multi-modal search engines [3,7,12] and recommender systems [6,34,35]. It motivates researchers to find semantic correspondence between two modalities and bridging their visual-semantic discrepancy. Some earlier works [14,16,24,31] focused on learning joint embeddings for the two modalities, while more recent ones [17,25,47,48] have turned to considering latent vision-language alignments at the level of regions and words.


A person is flying in the air.
A teddy bear on top of a phone.
Person : 0.94
Snowboard : 0.92
A person in green cloth is on a bike.
0.89
0.94

(a) Image-text Retrieval (b) Image Caption

(c) Object Detection (d) Phrase Grounding

Figure 1. Visualization on four different tasks. We visualize the activation map for text-to-image retrieval task in (a). As for the caption task in (b), we visualize regions selected by our model. Besides vision-language understanding task, SLAN can transfer to localization tasks, shown in (c) and (d), and we list the confidence score for each region.

In order to achieve fine-grained vision-language alignments, some works [20, 21, 26] use object detectors to extract key regions in images. Treated as black boxes, the detectors only support for fixed vocabulary object detection. Meanwhile, the extracted regions cannot adapt to different text information due to the freezing parameters of the detectors. To alleviates the problem, VinVL [47] applies a pre-trained object detector with more than 2000 classes and attributes to enrich local visual representations. However, the extended label set still limits the perceptive capability of object detectors for vision-language understanding compared to free-form text from real-world scenes.

Recently, more works have attempted to apply learnable region locators for vision-language tasks, which extract regions of interest conditioned on different texts.

[*] Indicates equal contributions. This work was done when J.T. Zhai and J.J. Liu were interning at Tencent Youtu Lab.

[†] J.J. Liu and M.M. Cheng are the corresponding authors.

Unlike previous methods using frozen object detectors, MDETR [17] builds an end-to-end framework on datasets with region-to-word annotations. GLIP [25] directly proposes grounded language-image pre-training for learning object-level, language-aware, and semantic-rich visual representations. These methods demonstrate their effectiveness in vision-language reasoning by introducing trainable locators. However, in order to supervise the training of locators, these methods require a certain amount of region-to-word grounding annotations (gold data), which are based on burdensome and expensive annotation efforts. It limits their applications on existing larger scale of vision-language datasets which have abundant but coarse-grained image and text pairs.

To address the problems above, we propose Self-Locator Aided Network (SLAN) for vision-language understanding. The designed self-locator is capable of accurately locating regions of interest based on different texts. Specifically, the self-locator consists of a region filter to select important regions and a region adaptor to update coordinates of regions with text guidance. By incorporating the self-locator into our framework, SLAN performs context-aware region extraction and vision-language feature fusion. Moreover, SLAN is trained solely on datasets with paired images and texts, making it scalable to larger pre-training settings for further performance improvements. With fine-grained region-word alignments, SLAN has a more detailed understanding of interactions in vision and language modalities.

To sum up, our contributions have three aspects:

- We propose a framework termed SLAN to capture fine-grained interplay between vision and language modalities. A self-locator is introduced to perform text-guided region adaptation, enabling dynamic region-word alignments for vision-language understanding tasks, as shown in Fig. 1.
- We demonstrate that SLAN can be easily applied to large-scale pre-training on vision-language datasets for being free from training with gold data. SLAN can also be naturally generalized to typical localization tasks, such as object detection and phrase grounding, due to its ability to locate key regions in images.
- Experiments on five vision-language understanding and two localization tasks demonstrate the effectiveness of our method. For example, SLAN achieves state-of-the-art performance on COCO image-text retrieval.

2. Related Work

2.1. Vision-language Task

Previous research has explored the relationship between visual and textual modalities and applied this knowledge to various downstream multi-modal tasks. Methods such as DeViSE [13], TBNN [36], and [49] have proposed loss functions and network structures to learn semantic visual-language alignments. Other approaches like SGG [41] and ViSTA [8] leverage prior tools or knowledge for image-text matching analysis.

Recently, leveraging visual backbone networks [11, 15, 40] and language encoders [18], vision-language pre-training on larger datasets has become increasingly popular. CLIP [31] pre-trains using 400M image-text pairs from the web, establishing global relations between images and texts. BLIP [23] benefits from extensive web data for vision-language understanding and generation tasks. Beit-3 [37] adopts mask-then-predict self-supervised training on large-scale monomodal and multi-modal data to learn internal vision-language dependencies.

However, these methods are constrained by the expense of fine-grained region-word datasets, making it challenging to directly provide local matching signals during pre-training for more accurate cross-modal knowledge. This knowledge enables models to precisely localize objects according to corresponding words, providing cues for downstream tasks.

2.2. Localization for Vision-language Task

Localization of image regions and words in sentences helps models learn local alignment. There are two kinds of methods based on whether the region proposal module is frozen or trained for vision-language tasks.

The first kind uses a frozen object detector (e.g., Faster R-CNN) pre-trained on Visual Genomes to extract detailed visual representations. Some later works (e.g., VinVL [47], Oscar [26]) increase the number of detection labels and introduce attribute information to complement visual concepts.

The other kind relies on fine-grained annotations of the vision-language dataset for pre-training. MDETR [17] introduces a modulated detector with multi-modal datasets that have precise alignments between phrases in text and objects in images. GLIP [25] applies grounded pre-training to learn object-level, language-aware, and semantic-rich visual representations. However, these methods require vision-language data with fine-grained annotations, limiting their application on larger-scale pre-training settings.

3. Self-Locator Aided Network (SLAN)

The framework of SLAN is shown in Fig. 2. We first briefly introduce the two unimodel encoders and then the detailed structures of other components. SLAN adaptively proposes and selects informative regions with text guidance, as described in Fig. 3. Finally, we list our pre-training objectives. The relevant symbols are described in Tab. 1.

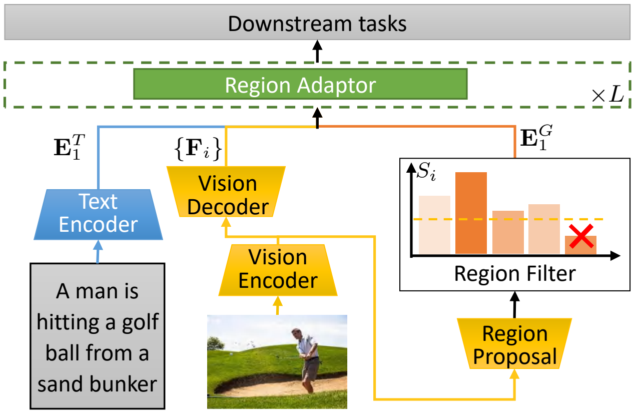
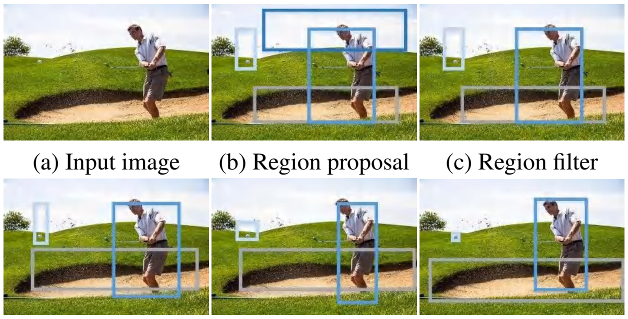


Figure 2. The SLAN framework. Two unimodal encoders extract textual and visual representations, respectively. The self-locator automatically generates, filter, and then iteratively adapts the image regions for fine-grained region-word alignments. The learned vision and language features can be used for downstream tasks.



(a) Input image (b) Region proposal (c) Region filter

(d) Results after the 1st, 2nd, and 3rd region adaptors

Figure 3. Sample intermediate results of the self-locator.

3.1. Unimodal Encoding

Two unimodal encoders learn textual and visual representations with D dimensions. We use BERT [18] as our text encoder, encoding words into a shared semantic space. The encoded embeddings $\mathbf{E}^T \in \mathbb{R}^{N^T \times D}$ summarize the whole sentence, including a textual token $\mathbf{T}_t \in \mathbb{R}^D$ from BERT's classification token and $N^T - 1$ word embeddings.

For image feature extraction, we encode images with classic vision backbone(*e.g.,* ResNet50 [15], Vit-Base [11], Vit-Large, and Vit-Huge) to obtain the vision feature map $\mathbf{V}$ with high-level semantics.

3.2. Self-locator for Vision-language Understanding

Since fine-grained region-word alignments are important for vision-language relation exploration, our self-locator follows the region proposal network [32] to output regions, where each region i contains spatial coordinates (x, y, w, h) and corresponding region embedding $\mathbf{E}_i^G \in \mathbb{R}^D$. The text-relevant local features $\mathbf{E}_i^G$ is extracted from $\mathbf{V}$ using RoIAlign. A vision token $\mathbf{T}_v$ is then obtained from global average pooling of $\mathbf{V}$ as a global summary of this image.

Symbol	Dimensions	Meaning
D	1×1	token dimension
L	1×1	number of layers/stages
K	1×1	number of grids per axis
N_i^h, N_i^w	1×1	grid size of neighbour
S_i	1×1	saliency score of region
p_{w_i}, p_{h_i}	1×1	scaling parameter
$\mathbf{E}^T$	$N^T \times D$	text embedding
$\mathbf{E}^G$	$N^G \times D$	region embedding
$\mathbf{F}_i$	$H_i \times W_i \times D$	pyramid feature map
$\mathbf{G}_i$	$N^G \times 4$	region coordinates
$\mathbf{T}_v, \mathbf{T}_t$	$1 \times D$	global visual/textual token
$\mathbf{A}_i$	$N^G \times N^T$	cross attention map

Table 1. Table of symbols, their dimensions, and meaning.

Different from most traditional object detection tasks that use the pre-defined label set, vision-language tasks usually have a wider vocabulary and free-form textual expressions. Therefore, our self-locator introduces a region filter for region importance prediction and a region adaptor for progressive region regression. By replacing fixed vocabulary prediction with region importance prediction, our self-locator assigns each region a saliency score S_i to estimate the probability that the region is useful for the alignment process. For traditional detection settings, the regression targets are annotated region coordinates. Since there is no grounding (gold) annotations in our setting, we propose progressive region regression in the multi-stage region adaptor, producing intermediate updated regions in each level. These updated regions are then used for supervising the internal region proposal module. As shown in Fig. 3, SLAN dynamically adapts region embeddings in $L = 3$ levels, yielding more flexible and accurate visual representations than the global visual feature maps, or patch embeddings from the vision transformer.

3.2.1 Vision Decoder: Pyramid Feature Extraction

Our proposed self-locator is designed for regression in a coarse-to-fine manner, requiring visual features of multi-scale. Considering these characteristics, we adopt a vision decoder after the global visual feature to extract multi-scale feature maps $\{\mathbf{F}_i\}$, where $i \in \{1, 2, ..., L\}$. $\mathbf{F}_i$ denotes the i-th level of decoder features, and $L = 3$ is the default number of layers of the self-locators. $\mathbf{F}_i$ is then fed to the i-th level of region adaptor. The structure of vision encoder and decoder follows the feature pyramid network [27].

3.2.2 Region Filter: Region Importance Prediction

When describing images, people usually focus on limited salient regions in the images [9, 10]. However, region proposal module [32] typically outputs a large number of re-

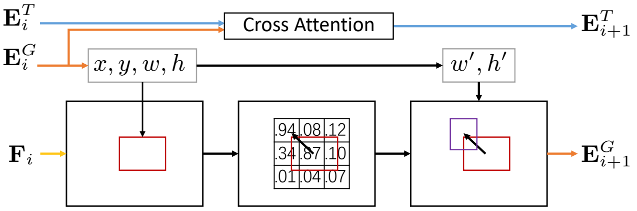


Figure 4. The i-th stage of the region adaptor. The region adaptor update each region's coordinate with text guidance. We use the feature map from vision decoder to extract region embeddings and explore latent region-word alignments.

gion proposals (*e.g.,* 100) for an image. Directly selecting all regions will lead to unnecessary computational cost and may also cause the model to learn from some meaningless region-to-word pairs. The strategy to control the maximum number of selected regions has three steps. (a) Normalize all saliency scores of the regions. After this process, the scores are represented as $S = \{S_1, ..., S_k\}, S_i \in [0, 1]$. (b) Sort these regions in descending order according to their saliency scores. (c) We pick no more than top T regions with saliency scores above a threshold h. Finally, we weight region embeddings by the scores. The saliency score of each proposed region is updated with gradients from downstream vision-language supervision, which will be described in Sec. 3.3.

3.2.3 Region Adaptor: Progressive Region Regression

The region adaptor aims at adjusting the coordinates of proposed regions to align with words with the same semantics. The difficulty comes from no annotated text-referenced regions as ground truths. We turn this problem into a L-level cascaded coarse-to-fine progressive regression progress, with $L = 3$ by default. As shown in Fig. 4, the i-th level of the region regression process receives three inputs: word embeddings $\mathbf{E}_i^T \in \mathbb{R}^{N^T \times D}$, region embeddings $\mathbf{E}_i^G \in \mathbb{R}^{N^G \times D}$ with their coordinates $\mathbf{G}_i \in \mathbb{R}^{N^G \times 4}$, and a global decoder feature map $\mathbf{F}_i \in \mathbb{R}^{H_i \times W_i \times D}$, where N^T and N^G denotes the number of words and selected regions, respectively. D denotes the dimension of embeddings.

The detailed procedure of progressive region regression is described in Algorithm 1. The vision-language multi-head attention layers fuse region and word embeddings and model their interactions as follows:

$$\begin{aligned} \mathbf{A}_i &= \frac{\mathbf{E}_i^G \mathbf{E}_i^{T\top}}{\sqrt{D}}, \\ \mathbf{E}_{i+1}^G &= Softmax(\mathbf{A}_i)\mathbf{E}_i^T, \\ \mathbf{E}_{i+1}^T &= Softmax(\mathbf{A}_i^\top)\mathbf{E}_i^G. \end{aligned} \tag{1}$$

With vision-language semantics, the updated vision-aware word embeddings $\mathbf{E}_i^T$ are able to guide region coordinate updates by searching for highly correlated regions around the original one. Specifically, the neighborhood of region $g = (x, y, w, h)$ is defined as a region of size (N_i^h, N_i^w) centered on it, where N_i^h and N_i^w are predefined parameters for the i-th level region regression process. The neighborhood is split to $K \times K$ regions to compute region-word similarities. As shown in Fig. 4, each region embedding is extracted with RoIAlign and then average pooling from F_i.

Algorithm 1 Self-localization

```
Input: Image $I$, region embeddings $\mathbf{E}_i^G$, text embeddings $\mathbf{E}_i^T$, pyramid feature map $\mathbf{F}_i$, neighbour size $(N_i^h, N_i^w)$, total region regression layers $L$.
    $p_{w_i}, p_{h_i}$ are learnable parameters independent for every region in each levels.
Output: Updated regions $\mathbf{G}_{out}$, region supervision on the region proposal module $\overline{\mathbf{G}}$, visual token $\mathbf{T}_v$, textual token $\mathbf{T}_t$.
    $\mathbf{G}_1, \mathbf{E}_1^G \leftarrow$ RegionProposal $(I)$
    $\mathbf{G}_1, S, \mathbf{E}_1^G \leftarrow$ RegionImportancePrediction $(\mathbf{G}_1, \mathbf{E}_1^G)$
    for $i \in \{1, 2, ..., L\}$ do
        $\mathbf{E}_{i+1}^G, \mathbf{E}_{i+1}^T \leftarrow$ CrossAttention$(\mathbf{E}_i^G, \mathbf{E}_i^T)$
        $\mathbf{E}_i^N \leftarrow$ NeighbourEmbedding$(N_i^h, N_i^w, \mathbf{G}_i)$
        $\Delta x_i, \Delta y_i \leftarrow$ Offset(Similarity$(\mathbf{E}_i^N, \mathbf{E}_{i+1}^T)$)
        $\mathbf{G}_{i+1} \leftarrow$ Update$(\mathbf{G}_i, \Delta x_i, \Delta y_i, p_{w_i}, p_{h_i})$
        $\mathbf{E}_{i+1}^G \leftarrow$ Embedding$(\mathbf{G}_{i+1}, \mathbf{F}_i)$
    end for
    $\mathbf{T}_v, \mathbf{T}_t \leftarrow$ ExtractCLS$(\mathbf{E}_{L+1}^G, \mathbf{E}_{L+1}^T)$
    $\mathbf{G}_{out} \leftarrow \mathbf{G}_{L+1}$
    $\overline{\mathbf{G}} \leftarrow (\sum_{i=2}^{L+1} \mathbf{G}_i)/L$
```

With different response scores to words, neighbor regions aggregate context information to the central one. The coordinate update for the central region is in the form of weighted summation of coordinates of its neighbor center points, as shown in Equ. (2):

$$\begin{aligned} \Delta x &= \sum_{j=0}^{K^2-1} M_j N_j^h(\lfloor \frac{j}{K} \rfloor - \lfloor \frac{K}{2} \rfloor), \\ \Delta y &= \sum_{j=0}^{K^2-1} M_j N_j^w(j \bmod K - \lfloor \frac{K}{2} \rfloor), \\ x' &= x + \Delta x, \quad y' = y + \Delta y, \\ w' &= p_w w, \quad h' = p_h h, \end{aligned} \tag{2}$$

where $\lfloor \cdot \rfloor$ is the round down operation. Every region in all levels of the region adaptor has its own p_w and p_h, which are set as learnable parameters. M_j is the maximum cosine similarity between the embedding of the j-th neighbor region and all word embeddings. The purpose of the last term in the first two lines of Equ. (2) is to map the 1D index to a 2D index (*e.g.,* from $\{0, 1, .., 8\}$ to $\{(0,0), (0,1), ..., (2,2)\}$).

For each original region g, let g_i denotes its updated version after the i-th layer in region regression. We take the average of them as the ground truth and apply the L_1 and GIoU regression loss:

$$\begin{aligned} \overline{g} &= \frac{\sum_{i=2}^{L+1} g_i}{L}, \\ \mathcal{L}_{reg}(g) &= \mathcal{L}_{L1}(g, \overline{g}) + \mathcal{L}_{GIoU}(g, \overline{g}). \end{aligned} \tag{3}$$

3.3. Pre-training Objectives with SLAN

SLAN is pre-trained on image-text pairs and learns fine-grained region-word alignments with the supervision from three common losses.

Image-Text Matching Loss (ITM) predicts whether a given image-text pair is positive or not, which can be viewed as a binary classification problem. The visual and textual tokens $(\mathbf{T}_v, \mathbf{T}_t)$ are concatenated and sent to a linear layer f_c. The ITM loss is formalized as follows:

$$\mathcal{L}_{itm}(\mathbf{I}, \mathbf{T}) = H(f_c(cat(\mathbf{T}_v, \mathbf{T}_t)), y_{v,t}), \tag{4}$$

where $y_{v,t}$ denotes the matching relation (1 for matched and 0 for unmatched), and H is the cross-entropy loss for classification. We directly select positive pairs from the dataset and build hard negative samples with batch sampling, following ALBEF [24].

Image-Text Contrastive Loss (ITC) ensures that visual and textual embeddings share the same semantic space and the positive (matched) image-text pairs are pulling closer than negative (unmatched) ones. We use two queues I_q, T_q to save the latest visited image and text samples. For each image-text pair $(\mathbf{I}, \mathbf{T})$, the softmax-normalized vision-language similarity is computed as as:

$$\begin{aligned} p_{i2t}(\mathbf{I}, \mathbf{T}, T_q) &= \frac{exp(sim(\mathbf{T}_v, \mathbf{T}_t)/\tau)}{\sum_{\mathbf{T}' \in T_q} exp(sim(\mathbf{T}_v, \mathbf{T}'_t)/\tau)} \\ p_{t2i}(\mathbf{T}, \mathbf{I}, I_q) &= \frac{exp(sim(\mathbf{T}_t, \mathbf{T}_v)/\tau)}{\sum_{\mathbf{T}'_v \in I_q} exp(sim(\mathbf{T}_t, \mathbf{T}'_v)/\tau)} \end{aligned}, \tag{5}$$

where τ is a temperature parameter and $sim(\cdot)$ measures vision-language similarity, which is implemented by the dot product between the image and text embeddings. Following ALBEF [24], we compute ITC loss as:

$$\mathcal{L}_{itc}(\mathbf{I}, \mathbf{T}) = -log(p_{i2t}(\mathbf{I}, \mathbf{T}, T_q)) - log(p_{t2i}(\mathbf{T}, \mathbf{I}, I_q)). \tag{6}$$

Language Modeling Loss (LM) encourages the model to predict masked words with context information. We randomly mask 15% text tokens and apply the masked language modeling loss as follows:

$$\mathcal{L}_{lm}(\mathbf{I}, \mathbf{T}) = H(p_{mask}(\mathbf{T}_v, \mathbf{T}_t), y_{mask}), \tag{7}$$

where y_{mask} denotes the masked word to predict and $p_{mask}(\mathbf{I}, \mathbf{T})$ is its predicted probability. $\mathcal{L}_{ds}$ is the downstream loss, which is computed by the sum of previous three losses.

$$\mathcal{L}_{ds}(\mathbf{I}, \mathbf{T}) = \mathcal{L}_{itm}(\mathbf{I}, \mathbf{T}) + \mathcal{L}_{itc}(\mathbf{I}, \mathbf{T}) + \mathcal{L}_{lm}(\mathbf{I}, \mathbf{T}). \tag{8}$$

The full pre-training objective is the combination of the downstream loss and our constraint on progressive region regression, computed as follows:

$$\mathcal{L} = \mathcal{L}_{ds} + \mathcal{L}_{reg}. \tag{9}$$

$\mathcal{L}_{reg}$ denotes the summation of the regression loss in Equ. (3) for all regions. The model is supervised by $\mathcal{L}$ during training.

4. Experiments

We first pre-train our method on a combined dataset of 14M image-text pairs from five datasets: COCO [28], Visual Genome [19] (excluding COCO images), Conceptual Captions [5], Conceptual [5], and SBU Captions [29]. We evaluate the proposed SLAN by comparing it to other state-of-the-art cross model methods on several downstream tasks. The ablation studies are further conducted to study how each component of our method influences the performance.

4.1. Implementation Details

We choose BERT$_{base}$ [18] as our text encoder, which is initialized from HuggingFace [39]. For the vision encoder, we explore four options: one is the CNN model ResNet50, and three kinds of ViT: ViT-Base, ViT-Large and ViT-Huge, which are all random initialized. As for the neighbour size for each region adaptor, we use a ratio r_i to denote them: (N_i^h, N_i^w) = $(r_i H_i, r_i W_i)$, where $r_1, r_2, r_3 = 1, 0.5, 0.25$, respectively. We pre-train SLAN for 20 epochs. For different options of the vision encoder, the batch size is set to 1280, 960, 640, 640 for ResNet50, Vit-B, Vit-L and Vit-H, respectively. The AdamW optimize is adopted with the initial learning rate 3e-4, and the learning rate is linearly decayed to 0. We resize the input images to 224×224.

4.2. Comparison on Downstream Tasks

We compare our approach with state-of-the-art methods on five challenging vision-language understanding tasks, including image-text retrieval, image captioning, visual question answering, natural language visual reasoning, zero-shot video-text retrieval. Besides, we transfer our method to two localization tasks: object detection and phrase grounding. The default vision encoder is Vit-Huge, if not specified.

Method	Backbone	Pre-training	Zero-shot						Fine-tune					
		Data	Image → Text			Text → Image			Image → Text			Text → Image		
			R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
ALIGN [16]	EfficientNet	1.8B	88.6	98.7	99.7	75.7	93.8	96.8	95.3	99.8	100.0	84.9	97.4	98.6
FILIP [44]	Vit-Large	300M	89.8	99.2	99.8	75.0	93.4	96.3	96.6	100.0	100.0	87.1	97.7	99.1
BLIP [23]	Vit-Large	14M	94.8	99.7	100.0	84.9	96.7	98.3	96.6	99.8	100.0	87.2	97.5	98.8
Beit-3 [37]	Vit-Giant	21M	94.9	99.9	100.0	81.5	95.6	97.8	98.0	100.0	100.0	**90.3**	98.7	99.5
Ours	Vit-Huge	14M	**96.0**	**100.0**	**100.0**	**86.1**	**97.0**	**98.5**	**98.1**	**100.0**	**100.0**	90.2	**99.0**	**99.6**

Table 2. Comparison with state-of-the-art image-text retrieval methods on Flickr30k. We use Recall@k scores as the evaluation metric under zero-shot and fine-tuning settings.

Method	Backbone	Pre-training	Retrieval(COCO)		Caption(COCO)				VQA(VQAv2)		NLVR(NLVR2)	
		Data	I2T R@1	T2I R@1	B@4	M	C	S	test-dev	test-std	dev	test-P
Oscar [26]	ResNet101	6.5M	73.5	57.5	37.4	30.7	127.8	23.5	73.6	73.8	79.1	80.3
VinVL [47]	ResNeXt152-C4	8.85M	75.4	58.8	38.5	30.4	130.8	23.4	76.5	76.6	82.6	83.9
SimVLM [38]	Vit-Huge	1.8B	-	-	40.6	33.7	143.3	25.4	80.0	80.3	84.5	85.1
GLIPv2-H [46]	Swin-Huge	16M	-	-	-	-	131.0	-	74.6	74.8	-	-
CoCa [45]	Vit-Giant	4.8B	-	-	40.9	33.9	143.6	24.7	82.3	82.3	86.1	87.0
BLIP [23]	Vit-Large	14M	82.4	65.1	40.4	-	136.7	-	78.2	78.3	82.1	82.2
Beit-3	Vit-Giant	21M	-	-	44.1	32.4	147.6	25.4	84.2	84.0	**91.5**	**92.5**
Ours	Vit-Huge	14M	**85.7**	**69.2**	**44.2**	**34.3**	**147.8**	**25.8**	**84.5**	**84.7**	91.0	91.7

Table 3. Comparison on more downstream tasks. For COCO retrieval, I2T and T2I represent image to text and text to image retrieval task, respectively. For COCO image captioning, we report BLEU@4 (B@4), METEOR (M), CIDEr (C), and SPICE (S) on the Karpathy test split. For VQA, we evaluate the vqa-score on VQAv2 test-dev and test-standard (test-std) splits. For NLVR, we report accuracy on NLVR2 development set (dev) and public test set (test-P).

4.2.1 Image-Text Retrieval

Given an image, the task expects to retrieve the corresponding text from the text gallery through the input image, and vice versa. We evaluate our method on Flickr30k [30] under zero-shot and fine-tune settings with Karpathy split and the performance is evaluated in terms of Recall@k. The comparative results are shown in Tab. 2. Specifically, on the same pre-training setting, SLAN also outperform BLIP [23] by 3.3% in average recall@1 on COCO.

4.2.2 Image Captioning

Given an input image, this task generates a sentence description to describe the image in detail. We use COCO Karpathy split to fine-tune and evaluate our method. Our SLAN outperforms most previous methods under this efficient setting, as shown in Tab. 3.

4.2.3 Visual Question Answering

Visual Question Answering (VQA) [1] requires the model to predict an answer from an image and a question. We follow [23] and treat VQA as an open-ended question-generation task. We fuse the image embedding with the question embedding and send them to the question decoder to get the result. As shown in Tab. 3, SLAN achieves higher performance with Beit-3 on VQAv2 test-dev and test-std, which adopts a larger vision backbone and more pre-training data.

4.2.4 Natural Language Visual Reasoning

Natural Language Visual Reasoning (NLVR2) [33] measures whether a sentence describes a pair of images. We extract the image and text embeddings from the image-text input, which are fused by a cross-attention layer. We use a binary classification module to predict their relations. Our results surpass most other methods by a large margin, and achieve comparable performance with Beit-3, showing the importance of learning fine-grained vision-language alignments.

4.2.5 Zero-shot Video-Text Retrieval

Besides the image-text tasks mentioned above, our method can generalize to the video-text retrieval task. We randomly select m frames of the video input and concatenate them to get an image-text sequence, then we feed them directly into our image-text retrieval model. The performance in Tab. 5 is comparable to others, demonstrating the vision-language knowledge learned in our method is semantic-rich.

Method	Backbone	Pretrain Data (M)		Object Detection (COCO)		Phrase Grounding (Flickr30k)		
		Image-Text	Region-Word	Zero-shot	Fine-tune	R@1	R@5	R@10
DETR [4]$_{\text{ECCV'20}}$	ResNet50	0	0	-	42.0	-	-	-
MDETR [17]$_{\text{ICCV'21}}$	ResNet101	0	0.2	-	-	84.3	93.9	95.8
GLIP [25]$_{\text{CVPR'22}}$	Swin-Large	24	3	49.8	60.8	87.1	96.9	98.1
GLIPv2 [46]$_{\text{NurIPS'22}}$	Swin-Huge	16	3	-	60.2	87.7	97.3	98.5
Beit-3 [37]$_{\text{CVPR'23}}$	Vit-Giant	21	0	-	**63.7**	-	-	-
Ours	ResNet50	14	0	46.9	59.2	86.8	96.6	97.4
Ours	Vit-Base	14	0	47	59.6	87.4	96.9	98.2
Ours	Vit-Large	14	0	48.5	60.5	89.1	98.0	98.9
Ours	Vit-Huge	14	0	**50.1**	63.5	**90.6**	**98.6**	**99.3**

Table 4. Comparison on two localization tasks: Object Detection on COCO and Phrase Grounding on Flickr30k. The pre-training data covers: image-text pairs and word-specific region annotations. We evaluate zero-shot and fine-tune settings on object detection. We use recall@k scores to evaluate phrase grounding task.

Method	R@1 ↑	R@5 ↑	R@10 ↑	MdR ↓
ClipBERT [22]	22.0	46.8	59.9	6
VideoCLIP [42]	30.9	55.4	66.8	-
FiT† [2]	43.3	65.6	74.7	2
BLIP† [23]	43.3	65.6	74.7	2
Ours†	**46.8**	**70.5**	**83.6**	**1.5**

Table 5. Comparisons with state-of-the-art methods for text-video retrieval on the 1k test split of the MSRVTT [43] dataset. †denotes the zero-shot settings, while others are fine-tuned ones.

Trainable	Adaptor	COCO		Flickr30k	
Region proposal	Number	TR@1	IR@1	TR@1	IR@1
✘	0	68.5	53.5	85.0	74.1
✔	0	69.1	53.8	86.7	76.2
✔	1	70.0	57.2	88.3	77.4
✔	2	70.8	57.5	88.7	78.1
✔	3	**72.1**	**58.3**	**90.3**	**78.9**

Table 6. Effect of training region proposal module and region adaptor. ✘ in the first coloum denotes applying a frozen region proposal module and no self-locator. TR@1 and IR@1 denote recall@1 of image to text and text to image retrieval. To evaluate the effect of the self-locator against a frozen region proposal module, we load the pre-trained weight from COCO Detection to compare with our method for line 1. The remaining experiments train from scratch. We use Vit-Base as the vision encoder.

Top K	Threshold	COCO		Flickr30k	
		TR@1	IR@1	TR@1	IR@1
-	-	69.4	54.1	85.9	74.7
10	-	70.6	56.8	87.5	77.3
10	0.3	71.2	57.6	89.1	78.2
10	0.5	**72.1**	**58.3**	**90.3**	**78.9**

Table 7. Different settings of the region filter.

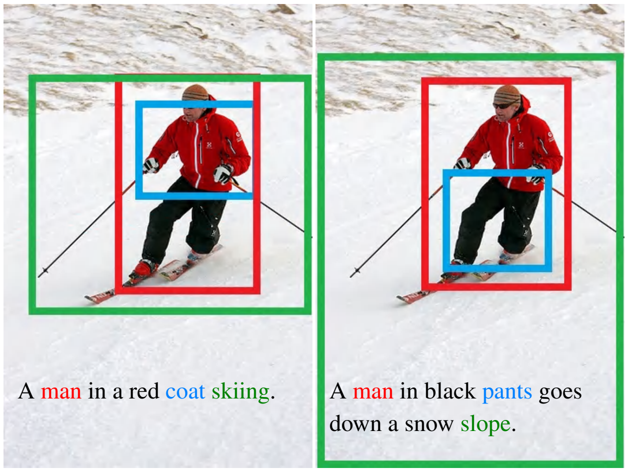

A man in a red coat skiing.
A man in black pants goes down a snow slope.

Figure 5. Illustration of text-specific region adaptation. We colorize three words per sentence and use the corresponding colors to mark the regions with the highest matching scores. This denotes the interpretable region adaptation of our method, which brings fine-grained vision-language feature fusion for downstream tasks.

4.2.6 Localization Tasks

We conduct two localization tasks: object detection on COCO, and phrase grounding on Flickr30k. For the text input on the object detection task, we use a prompt composed of concatenated labels of COCO (e.g. detect: person, bicycle, car, ... , toothbrush). We adopt the output from the last layer of the Tab. 4 has exciting results about SLAN on localization tasks. For example, in the task of object detection with Vit-Base as the backbone, SLAN achieves comparable results to GLIP using a larger backbone and 3M gold data. Though not designed for localization tasks, our method with Vit-Huge as backbone outperforms almost all comparative methods.

Method	Backbone	Params(M)	FLOPS(G)	COCO	
				TR@1	IR@1
BLIP	Vit-Base	370	558	81.9	64.3
BLIP	Vit-Large	810	1594	82.4	65.1
Coca	Vit-Giant	2100	4103	83.0	65.5
Beit-3	Vit-Giant	1900	-	84.8	67.2
Ours	ResNet50	322	324	**85.1**	**68.9**

Table 8. Comparison of model parameter and FLOPS on vision-language retrieval task. The FLOPS is calculated with the image of 384x384 resolution. The Backbone denotes the vision encoder.

4.3. Ablation Study

4.3.1 Effect of Self-locator

Importance of learnable region proposal module. As shown in the first row of Tab. 6, we replace our self-locator with a frozen detector pre-trained on COCO Detection, and the second row is the result of our learnable region proposal module. We do not load the region proposal module with pre-trained weights, but only fine-tune on the downstream task datasets. Our method improves on average about 0.5% and 2% on COCO and Flickr30k's image-to-text and text-to-image retrieval tasks, respectively.

Number of region adaptors for region regression. The region adaptor performs progressive regression on the regions output by the region proposal module to provide more accurate region localization for vision-language understanding tasks. As shown in Tab. 6, when the number of region adaptors increases from 0 to 3, the retrieval performance can be significantly improved by an average of more than 3%.

Region filter for saliency prediction. Tab. 7 illustrates how the region filter affects the performance on COCO and Flickr30k retrieval tasks. Learnable region proposal module is trained from scratch and the number of region adaptors is set to 3. The first and second rows show that when the regions are sorted by saliency score and selected by a certain number, we can achieve an performance gain of 2% on each dataset. When using the saliency score threshold, our region filter is able to remove redundant regions that negatively affect vision-language adaptation with higher performance.

4.3.2 Computational Cost

Tab. 8 illustrates the computational cost of our method and other state-of-the-art methods. Our method has the smallest amount of parameters and FLOPS because in this experiments our vision backbone is a relatively lightweight ResNet50. However, our retrieval performance significantly outperforms other methods. As far as our concerned, our method is efficient and high-performance.

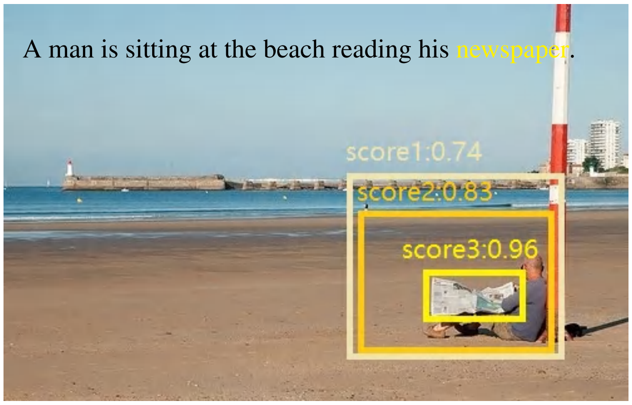


Figure 6. Coarse-to-fine process of the region adaptation. We also list the matching score between the regions and counterpart words. Note that each region of different levels in the region adaptor has independent scaling and moving behavior with our implementation.

4.4. Visualization Analysis

4.4.1 Text-guided Region Adaptation

As shown in Fig. 5, our region adaptor produces text-specific results with relatively high confidence. When we change the detailed description of the sentence, e.g., "a man in a red coat" to "a man in black pants", the interesting phenomenon is that the attention regions of our self-locator are also shifted accordingly with relatively high confidence.

4.4.2 Coarse-to-fine Region Adaptation

To verify the calibration effect of region adaptation, we visualize an image with its text in Fig. 6. Model locates more accurate regions of interest with higher similarity scores after three levels of region adaptor. This shows that our self-locator hierarchically refines the related regions of provided words.

5. Conclusions and Future Work

In this paper, we introduce the Self-Locator Aided Network (SLAN), which leverages the self-locator to adapt proposed regions for vision-language alignments without the need for extra grounding (region-to-word) annotations. We aim to further investigate and optimize the self-locator's performance for various localization applications.

Aknowledgements. This research was supported by the NSFC (NO. 62225604, 62176130) and the Fundamental Research Funds for the Central Universities (Nankai University, 070-63233089). The Supercomputing Center of Nankai University supports computation.

References

[1] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Int. Conf. Comput. Vis.*, 2015. 6

[2] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *Int. Conf. Comput. Vis.*, 2021. 7

[3] Yue Cao, Mingsheng Long, Jianmin Wang, and Shichen Liu. Collective deep quantization for efficient cross-modal retrieval. In *AAAI*, 2017. 1

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Eur. Conf. Comput. Vis.*, 2020. 7

[5] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 5

[6] Hui Chen, Guiguang Ding, Xudong Liu, Zijia Lin, Ji Liu, and Jungong Han. Imram: Iterative matching with recurrent attention memory for cross-modal image-text retrieval. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 1

[7] Kan Chen, Trung Bui, Chen Fang, Zhaowen Wang, and Ram Nevatia. Amc: Attention guided multi-modal correlation learning for image search. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017. 1

[8] Mengjun Cheng, Yipeng Sun, Longchao Wang, Xiongwei Zhu, Kun Yao, Jie Chen, Guoli Song, Junyu Han, Jingtuo Liu, Errui Ding, et al. Vista: Vision and scene text aggregation for cross-modal retrieval. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022. 2

[9] Ming-Ming Cheng, Niloy J. Mitra, Xiaolei Huang, Philip H. S. Torr, and Shi-Min Hu. Global contrast based salient region detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 37(3):569–582, 2015. 3

[10] Robert Desimone and John Duncan. Neural mechanisms of selective visual attention. *Annual review of neuroscience*, 18(1):193–222, 1995. 3

[11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2020. 2, 3

[12] Benjamin Elizalde, Shuayb Zarar, and Bhiksha Raj. Cross modal audio search and retrieval with joint embeddings based on text and audio. In *ICASSP*, 2019. 1

[13] Andrea Frome, Gregory S Corrado, Jonathon Shlens, Samy Bengio, Jeffrey Dean, Marc'Aurelio Ranzato, and Tomas Mikolov. Devise: A deep visual-semantic embedding model. In *NIPS*, 2013. 2

[14] Zhe Gan, Yen-Chun Chen, Linjie Li, Chen Zhu, Yu Cheng, and Jingjing Liu. Large-scale adversarial training for vision-and-language representation learning. *NIPS*, 2020. 1

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016. 2, 3

[16] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *Int. Conf. Mach. Learn.*, 2021. 1, 6

[17] Aishwarya Kamath, Mannat Singh, Yann LeCun, Gabriel Synnaeve, Ishan Misra, and Nicolas Carion. Mdetr-modulated detection for end-to-end multi-modal understanding. In *Int. Conf. Comput. Vis.*, 2021. 1, 2, 7

[18] Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, 2019. 2, 3, 5

[19] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *Int. J. Comput. Vis.*, 2017. 5

[20] Chia-Wen Kuo and Zsolt Kira. Beyond a pre-trained object detector: Cross-modal textual and visual context for image captioning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022. 1

[21] Kuang-Huei Lee, Xi Chen, Gang Hua, Houdong Hu, and Xiaodong He. Stacked cross attention for image-text matching. In *Eur. Conf. Comput. Vis.*, 2018. 1

[22] Jie Lei, Linjie Li, Luowei Zhou, Zhe Gan, Tamara L Berg, Mohit Bansal, and Jingjing Liu. Less is more: Clipbert for video-and-language learning via sparse sampling. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 7

[23] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *Int. Conf. Mach. Learn.*, 2022. 2, 6, 7

[24] Junnan Li, Ramprasaath R Selvaraju, Akhilesh Gotmare, Shafiq R Joty, Caiming Xiong, and Steven Chu-Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. In *NIPS*, 2021. 1, 5

[25] Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. Grounded language-image pre-training. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022. 1, 2, 7

[26] Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, et al. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *Eur. Conf. Comput. Vis.*, 2020. 1, 2, 6

[27] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017. 3

[28] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Eur. Conf. Comput. Vis.*, 2014. 5

[29] Vicente Ordonez, Girish Kulkarni, and Tamara Berg. Im2text: Describing images using 1 million captioned photographs. *NIPS*, 2011. 5

[30] Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Int. Conf. Comput. Vis.*, 2015. 6

[31] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *Int. Conf. Mach. Learn.*, 2021. 1, 2

[32] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, 2015. 3

[33] Alane Suhr, Stephanie Zhou, Ally Zhang, Iris Zhang, Huajun Bai, and Yoav Artzi. A corpus for reasoning about natural language grounded in photographs. *ACL*, 2019. 6

[34] Rui Sun, Xuezhi Cao, Yan Zhao, Junchen Wan, Kun Zhou, Fuzheng Zhang, Zhongyuan Wang, and Kai Zheng. Multimodal knowledge graphs for recommender systems. In *CIKM*, 2020. 1

[35] Hao Wang, Doyen Sahoo, Chenghao Liu, Ee-peng Lim, and Steven CH Hoi. Learning cross-modal embeddings with adversarial networks for cooking recipes and food images. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019. 1

[36] Liwei Wang, Yin Li, Jing Huang, and Svetlana Lazebnik. Learning two-branch neural networks for image-text matching tasks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2018. 2

[37] Wenhui Wang, Hangbo Bao, Li Dong, Johan Bjorck, Zhiliang Peng, Qiang Liu, Kriti Aggarwal, Owais Khan Mohammed, Saksham Singhal, Subhojit Som, and Furu Wei. Image as a foreign language: BEiT pretraining for vision and vision-language tasks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2023. 2, 6, 7

[38] Zirui Wang, Jiahui Yu, Adams Wei Yu, Zihang Dai, Yulia Tsvetkov, and Yuan Cao. Simvlm: Simple visual language model pretraining with weak supervision. In *Int. Conf. Learn. Represent.*, 2021. 6

[39] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. Transformers: State-of-the-art natural language processing. In *EMNLP*, 2020. 5

[40] Yu-Huan Wu, Yun Liu, Xin Zhan, and Ming-Ming Cheng. P2t: Pyramid pooling transformer for scene understanding. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2022. 2

[41] Danfei Xu, Yuke Zhu, Christopher B Choy, and Li Fei-Fei. Scene graph generation by iterative message passing. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017. 2

[42] Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. Videoclip: Contrastive pre-training for zero-shot video-text understanding. *EMNLP*, 2021. 7

[43] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016. 7

[44] Lewei Yao, Runhui Huang, Lu Hou, Guansong Lu, Minzhe Niu, Hang Xu, Xiaodan Liang, Zhenguo Li, Xin Jiang, and Chunjing Xu. Filip: Fine-grained interactive language-image pre-training. In *Int. Conf. Learn. Represent.*, 2021. 6

[45] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022. 6

[46] Haotian Zhang, Pengchuan Zhang, Xiaowei Hu, Yen-Chun Chen, Liunian Harold Li, Xiyang Dai, Lijuan Wang, Lu Yuan, Jenq-Neng Hwang, and Jianfeng Gao. Glipv2: Unifying localization and vision-language understanding. *NIPS*, 2022. 6, 7

[47] Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. Vinvl: Revisiting visual representations in vision-language models. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 1, 2, 6

[48] Qi Zhang, Zhen Lei, Zhaoxiang Zhang, and Stan Z Li. Context-aware attention network for image-text retrieval. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 1

[49] Zhedong Zheng, Liang Zheng, Michael Garrett, Yi Yang, Mingliang Xu, and Yi-Dong Shen. Dual-path convolutional image-text embeddings with instance loss. *TOMM*, 2020. 2