An Anchor-based Relative Position Embedding Method for Cross-Modal Tasks

Ya Wang[1,*], **Xingwu Sun**[1,2,*], **Fengzong Lian**[1], **Zhanhui Kang**[1], **Chengzhong Xu**[2]
Machine Learning Platform Department, Tencent[1]
State Key Lab of IOTSC, Department of Computer Science, University of Macau[2]
connorywang@tencent.com, sammsun@tencent.com
faxonlian@tencent.com, kegokang@tencent.com, czxu@um.edu.mo

Abstract

Position Embedding (PE) is essential for transformer to capture the sequence ordering of input tokens. Despite its general effectiveness verified in Natural Language Processing (NLP) and Computer Vision (CV), its application in cross-modal tasks remains unexplored and suffers from two challenges: 1) the input text tokens and image patches are not aligned; 2) the encoding space of each modality is different, making it unavailable for feature comparison. In this paper, we propose a unified position embedding method for these problems, called AnChor-basEd Relative Position Embedding (ACE-RPE), in which we first introduce an *anchor* locating mechanism to bridge the semantic gap and locate anchors from different modalities. Then we conduct the distance calculation of each text token and image patch by computing their shortest paths from the located anchors. Last, we embed the anchor-based distance to guide the computation of cross-attention. In this way, it calculates cross-modal relative position embeddings for cross-modal transformer. Benefiting from ACE-RPE, our method obtains new SOTA results on a wide range of benchmarks, such as Image-Text Retrieval on MS-COCO and Flickr30K, Visual Entailment on SNLI-VE, Visual Reasoning on NLVR2 and Weakly-supervised Visual Grounding on RefCOCO+.

1 Introduction

Transformer (Vaswani et al., 2017) has shown excellent performance in Natural Language Processing (NLP), Computer Vision (CV) as well as cross-modal tasks, including natural language inference (Devlin et al., 2018), image classification (Wu et al., 2021), visual question answering (Wu et al., 2017) and visual entailment (Xie et al., 2019), etc. Nevertheless, transformer module lacks the capability to capture the ordering information of the input tokens because of the limitation of its self-attention mechanism. Therefore, incorporating explicit position representations is crucial to improve the performance of transformer-based models (Devlin et al., 2018; Dosovitskiy et al., 2020).

Generally, there are two mainstream position encoding methods in transformer-based NLP and CV models, *i.e.*, absolute position embedding (APE) and relative position embedding (RPE). APE methods (Vaswani et al., 2017; Devlin et al., 2018; Dosovitskiy et al., 2020) encode absolute positions of the input tokens with either trainable (Devlin et al., 2018) or fixed embedding (Vaswani et al., 2017). These position embeddings are added with the token embeddings, which are then passed to the self-attention layer to calculate the token relationship considering their positional information. It has been verified effective in a variety of NLP (Wang et al., 2020; Devlin et al., 2018) and CV (Wu et al., 2021) tasks. On the other hand, RPE methods (Chu et al., 2021; Shaw et al., 2018) encode the pairwise distances of every two tokens. Commonly, it directly interacts with the calculation of attention mechanism in different ways (Wu et al., 2021; Chu et al., 2021). Compared with APE, RPE methods are superior to modeling the positional information of extremely long or variant-length sequences. As a result, in some span prediction tasks of NLP, RPE methods are shown to achieve more performance gains than APE ones (Wang et al., 2020).

Despite the success of the position embedding methods in unimodal tasks, its exploration in the field of cross-modal modeling is still limited. Recent works on cross-modal tasks (Cho et al., 2021; Li et al., 2021) could be classified into two frameworks, 1) One-stage methods (Fig. 1(a)) which extract the cross-modal representation with a unified cross-modal encoder; 2) Two-stage methods (Fig. 1(b)), which have additional text encoder and image encoder. Both of them adopt the position embeddings in a separate way, where the text and image

[*] : equal contribution

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 5401 - 5413
December 7-11, 2022

position representations are embedded individually. In this way, the models can only learn position embedding in each modality separately while ignoring positional information between two tokens from different modalities. However, it is challenging to raise a unified method for cross-modal position embedding. Firstly, the inputs from two modalities are embedded into different spaces, making the input embedding not comparable. Secondly, since the text tokens and image patches are not aligned, the relative positions between two units from different modalities are meaningless.

In this paper, we advocate a new perspective for effective cross-modal position encoding (shown in Fig. 1(c)), called AnChor-basEd Relative Position Embedding (ACE-RPE). It first computes alignment between text and image tokens to locate aligned pieces, which are called *anchors* in this paper. Subsequently, the token-to-token (t2t) and patch-to-patch (p2p) relative position is calculated for unimodal ordering information. The relative position searching of arbitrary text token and image patch is then considered as a shortest path problem, containing three steps: 1) routing from given token and its nearby anchors; 2) routing from anchors and their located image patches, and 3) routing from the located patches to the given image patch. As illustrated in Fig. 2, the relative position of "A" and the image patch of the man is derived from three terms: the t2t relative position between "A" and the anchor "man", the relative position from anchor "man" to the image patch matching "man", and the relative position from the located image patch to the patch of human (obviously, 0 in this case). Finally, we embed the anchor-based relative position to the self-attention calculation. Further, we conduct extensive experiments to verify the effectiveness of the proposed ACE-RPE compared to many strong baselines. The results demonstrate that our method can boost the performance of cross-modal transformers with a large margin.

The main contributions of this work can be summarized as follows,

- We propose the ACE-RPE method to incorporate positional information into cross-modal transformers and bridge the gap of different modalities. As we know, it is the first work to model relative position in cross-modal tasks.
- We give an anchor-based RPE method to get relative positions according to the located anchors between two modalities. Extensive experiments compared with strong baselines reveals the effectiveness of this method.
- Our method achieves new SOTA in 5 cross-modal benchmarks, including Flickr30K (Plummer et al., 2015), MS-COCO (Lin et al., 2014), SNLI-VE (Xie et al., 2019), NLVR2 (Suhr et al., 2018) and RefCOCO+ (Yu et al., 2016). In addition, it also surpasses baseline methods significantly on VQA (Goyal et al., 2017).

2 Related Work

2.1 Position Embedding for NLP

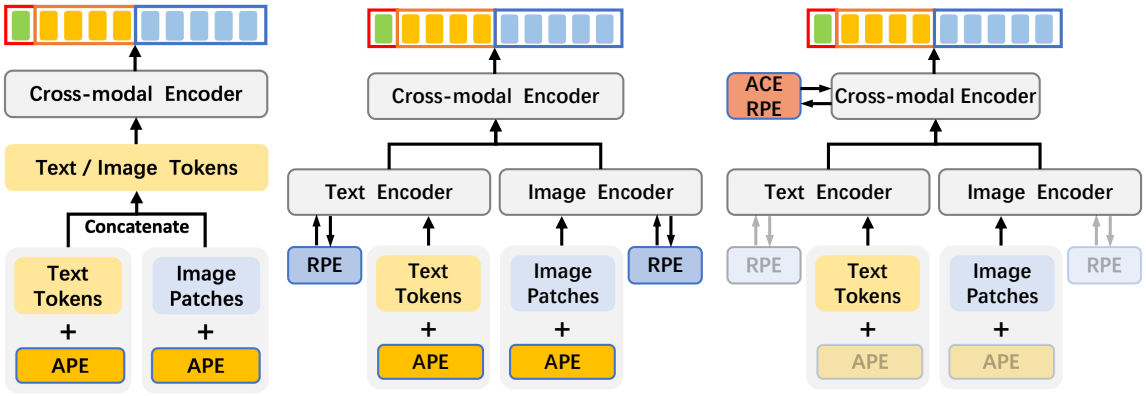
Currently, Transformer (Vaswani et al., 2017) plays a major role in the field of NLP. It shows superiority in many real-world tasks, such as natural language inference (Devlin et al., 2018) and question answering (Devlin et al., 2018; Rajpurkar et al., 2016). However, the self-attention of transformer lacks the ability to capture ordering information of input tokens in a sequence. Such that, additional explicit representations for token positions are crucial to the performance of the transformer.

The position embedding in NLP could be categorized into two classes: APE and RPE. APE encodes the absolute position of tokens in a sequence. Each position has its individual embedding, which are generated with specific functions, like sinusoidal operator (Vaswani et al., 2017) or learnable encoding (Devlin et al., 2018). Usually, the generated APE is added with the input text tokens for an explicit perspective view of token positions. Therefore, the same token in different positions will have different embedding. Currently, various works on APE are proposed to further boost the performance of transformer-based methods.

RPE (Dai et al., 2019; Devlin et al., 2018; Raffel et al., 2019) encodes the pairwise relative token position via interacting with the query, key or value in self-attention modules (Shaw et al., 2018). Compared to APE, RPE is translation-invariant and could encode variable lengths of input sequences. Therefore, it is shown to surpass APE on some long-sequence tasks (Wang et al., 2020).

2.2 Position Embedding for CV

With the great success of Visual Transformer (ViT) (Dosovitskiy et al., 2020) on large-scale dataset, the transformer-based methods have also become



(a) One-stage (b) Two-stage (c) Ours

Figure 1: Conceptual comparison of three position embedding methods. The output blocks in green, orange and blue present the [CLS] token, text and image embedding. (a) The One-stage method (Tsai et al., 2019), which has a unified cross-modal encoder. Only APE is utilized in this method. (b) Two-stage method (Li et al., 2021), containing extra text and image encoders. Both AFE and RPE are injected in the backbone, but they are embedded modality-separately. (c) Our ACE-RPE method. Except for unimodal AFE and RPE, ACE-RPE is proposed to leverage the cross-modal encoder with the relative position information from different modalities.

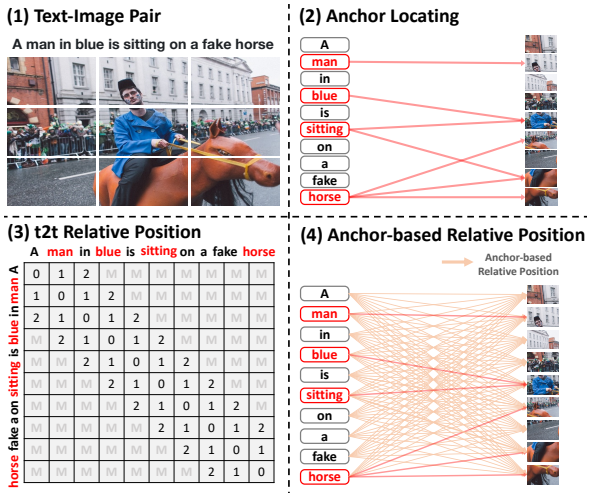


Figure 2: A case from MS-COCO (Lin et al., 2014) to illustrate ACE-RPE. The proposed Anchor-based Relative Position is calculated with the located anchors (words in red) and t2t relative position. "M" is the masked relative position.

an important paradigm in the area of CV (Dosovitskiy et al., 2020; Wu et al., 2021). Following the transformer-based methods in NLP tasks, the position embedding is also considered as a key component to obtain better performance on CV tasks. Though common RPE on images could outperform APE methods in some tasks (Dosovitskiy et al., 2020), it is demonstrated by some works (Dosovitskiy et al., 2020; Srinivas et al., 2021) that the superiority of RPE is not solid. To handle this issue, some follow-up works (Chu et al., 2021; Wu et al., 2021; Zhang and Yang, 2021) present significant improvement on RPE methods, which could overpass APE counterparts by more robust margins.

In summary, position embedding has been proved to have a significant effect on the performance of transformer-based models in both NLP and CV. However, the exploration on cross-modal tasks is still vacant. One of the most important reasons is that it is challenging to find a meaningful "position" between different modalities. For example, it is not available for us to define the position of the word "are" in a text and the corresponding patches in an image. To this end, we propose an anchor-based method, which bridges the gap between the text and image modalities and makes it possible to calculate position embeddings of different modalities.

3 Methods

The overview of our backbone network is presented in Fig. 3, which contains a 6-layer visual transformer (Dosovitskiy et al., 2020) as the image encoder, a 6-layer linguistic transformer (Devlin et al., 2018) as the text encoder and a 6-layer cross-modal transformer. The AnChor-basEd Position Embedding (ACE-RPE) is proposed to leverage the cross-modal encoder with cross-modal positional information. It involves two key procedures: 1) learning the locating of cross-modal anchors; 2) ACE-RPE calculation by incorporating anchor locating and t2t/i2i relative position. In this section, we first present the above procedures in detail (Sec. 3.1 and Sec. 3.2). Then, we present the overall pre-training objectives of our method.

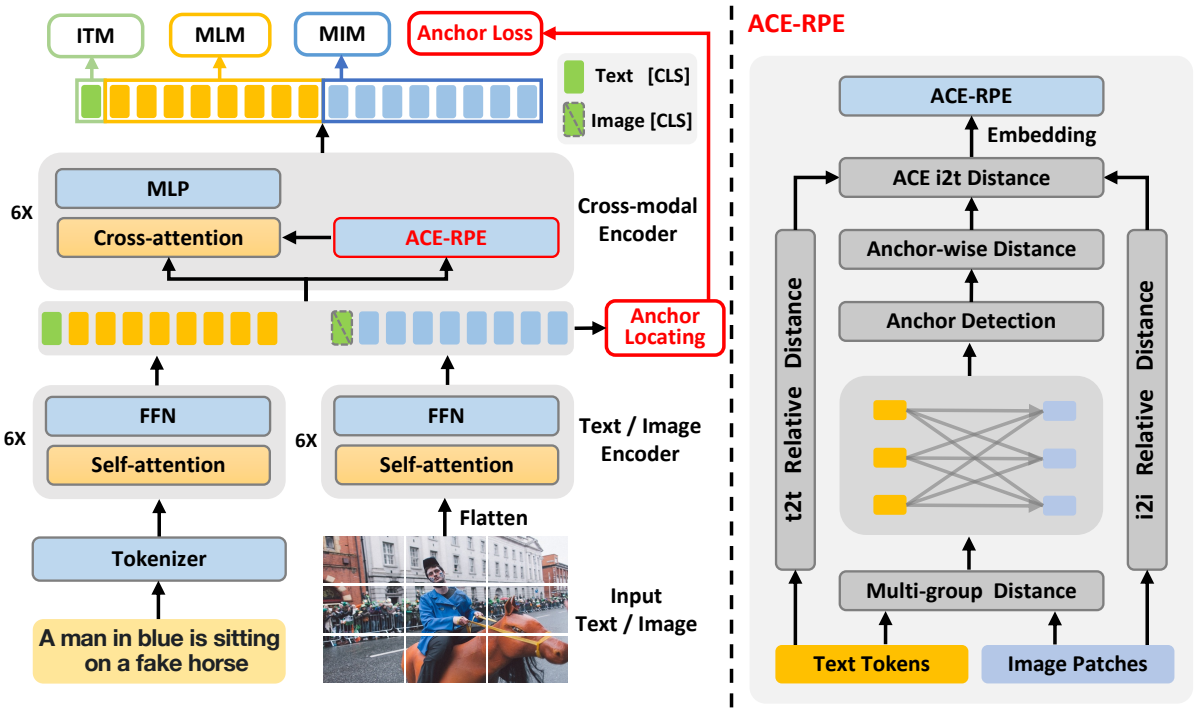


Figure 3: The overall architecture of our ACE-RPE method. It contains a text encoder, an image encoder and an extra cross-modal encoder to extract cross-modal features. Firstly, it learns the cross-modal locating of anchors in an unsupervised manner. Then, the cross-modal position embedding is calculated by interacting with the input embedding of text tokens and image patches (detailed in the right part), which serves as the RPE of the following cross-modal encoder. The model pre-training follows four objectives: Image-Text Matching (ITM), Masked Language Modeling (MLM), Masked Image Modeling (MIM) and Anchor Loss.

3.1 Cross-modal Locating of Anchors

Considering an image x and its corresponding text y, the "anchor" in this paper refers to the prominent tokens of y, which can be located to some patches of x. An illustration of cross-modal anchors is depicted in Fig. 2. Naturally, the word "man" is associated with the image patch containing the human, and "blue" can be located to the blue patches. Then, the words "tie" and "cat" are called ***anchors*** in this paper.

In this part, we propose an unsupervised method to figure out the cross-modal anchors effectively. It uses a token-wise loss to search for anchors without any additional annotations. Formally, the raw image x is segmented into $M+1$ image patches (Dosovitskiy et al., 2020), i.e., $x = \{c_x, x_1, x_2, \cdots, x_m, \cdots, x_M\}$, where each of them is embedded with a normalized D-dimensional vectors, c_x is an image [CLS] token. Similarly, the text y is tokenized to $N+1$ text tokens, $y = \{c_y, y_1, y_2, \cdots, y_n, \cdots, y_N\}$, where c_y is a text [CLS] token. The token-wise similarity between the image patch x_m and text token y_n is computed by a specific similarity function (cosine similarity in this paper) f. We then introduce an ***anchor loss*** to maximize the similarity of the anchors and their matching image patches, without changing the similarity of unmatched pairs, e.g., "blue" and patches of the "horse" in Fig. 2. Accordingly, the proposed anchor loss is formulated based on contrastive learning and log-sum-exp trick[1]:

$$\mathcal{L}_{ace} = \frac{1}{2}\mathbb{E}_{(x,y)}\Big[H_{i2t}(x, \mathcal{O}_y) + H_{t2i}(y, \mathcal{O}_x) - \frac{1}{\lambda}\log\sum_{m,n} e^{\lambda f(x_m, y_n)}\Big] \quad (1)$$

where λ is a scale parameter. $\mathcal{O}_y$ and $\mathcal{O}_x$ indicate the dynamic dictionaries (He et al., 2020), containing one positive sample y and $K-1$ negative samples, that is only text y in $\mathcal{O}_y$ matches image x. K is 65, 536 in this paper, following (Li et al., 2021). f presents the similarity function (cosine similarity in this paper). $H_{i2t}(X, \mathcal{O}_y)$ and $H_{t2i}(Y, \mathcal{O}_x)$ denote the image-to-text and text-to-image con-

[1]Inspired by (Nielsen and Sun, 2016), log-sum-exp is a soft-smoothing version of maximum operation. It is used to output some maximum values (the number is adjusted by a scale parameter), while small values tend to zero

trastive losses based on K-pairs, respectively

$$H_{i2t}(x, \mathcal{O}_y) = -min\Big\{0,\ f(x,y) - \delta - \frac{1}{\lambda} log \sum_{z \in \mathcal{O}_y, z \neq y} e^{\lambda f(x,z)}\Big\} \quad (2)$$

here δ is the margin between positive and negative samples, which is empirically set to 0.05 in our experiments. $H_{t2i}(y, \mathcal{O}_x)$ is defined accordingly.

3.2 Calculation of ACE-RPE

The calculation of ACE-RPE refers to three major components: 1) the locating of anchors with multi-group relative position; 2) the computation of anchor-based cross-modal relative position between text tokens and image patches; 3) cross-modal relative position embedding. Each step is elaborated as follows.

3.2.1 Locating of Anchors

The relative position between anchors and their relative image patches is dynamically generated with a proposed ***multi-group cross-modal similarity***,

$$S_G(x_m, y_n) = \Big[f(\hat{x}_m^1, \hat{y}_n^1),\ f(\hat{x}_m^2, \hat{y}_n^2),\ \cdots,\ f(\hat{x}_m^G, \hat{y}_n^G)\Big] \quad (3)$$

where G is the number of groups, $\hat{x}_m \in \mathbb{R}^{G\frac{D}{G}}, \hat{y}_n \in \mathbb{R}^{G\frac{D}{G}}$ are the reshaped versions of x_m and $\hat{x}_m^j \in \mathbb{R}^{\frac{D}{G}}, \hat{y}_n^j \in \mathbb{R}^{\frac{D}{G}}$, Note that our proposed multi-group cross-modal similarity is not a scalar but a vector of length G.

Shown in Eqn. 3, the multi-group cross-modal similarity functions on all text tokens and image patches. We then introduce a post-locating for anchors with a soft shrinking operator,

$$\widehat{S}_G(x_m, y_n) = \begin{cases} S_G(x_m, y_n), & S_G(x_m, y_n) \geq \delta \\ \delta e^{\tau(S_G(x_m, y_n) - \delta)}, & S_G(x_m, y_n) < \delta \end{cases} \quad (4)$$

where δ is a hyper-parameter. τ is a large enough scalar, set to 10^4 is this paper.

The set of anchors is then defined as

$$\mathcal{A}_G(x, y) = \{x_m \mid \exists\, y_n, s.t.\ \widehat{S}_G(x_m, y_n) \geq \delta\} \quad (5)$$

where "$\geq$" is calculated element-wisely by each group of $\widehat{S}_G$. Hence, the $\mathcal{A}_G(x, y)$ is a collection of G anchor sets, which may be different in different groups. As indicated in Eqn. 8 and analyzed in Sec. A.2, the multi-group anchor sets instead of a single one can enhance the flexibility of position embeddings.

Finally, the distance between anchors and their relative image patches is,

$$D_G(x_m, y_n) = \frac{1}{\widehat{S}_G(x_m, y_n)} \quad (6)$$

3.2.2 Anchor-based Cross-modal Relative Position Calculation

Given an arbitrary text token and an image patch, we consider the calculation of their relative position as a shortest path problem, where the path is split into three steps: 1) route from the given text token to nearby anchors; 2) route from anchors to their located image patches and 3) route from the located image patches to the given image patch. Formally, the anchor-based relative distance is,

$$P_{ace}(x_m, y_n) = \min_{i,j} \Big\{D_{p2p}(x_m, x_i) \oplus D_G(x_i, y_j) \oplus D_{t2t}(y_j, y_n)\Big\} \quad (7)$$

where "$\oplus$" is the broadcasting addition of scalars and vectors. "$min(\cdot)$" is executed in an inner-group manner, i.e., the values are compared in each group. Therefore, the output $P_{ace}(x_m, y_n)$ keeps a vector of length G. Here D_{p2p} and D_{t2t} are the common image patch-to-patch and text token-to-token physical distance, respectively. For efficiency, we only consider neighborhood of B_p tokens in D_{t2t} and a square neighborhood of B_t image paths in D_{p2p}. It should be noted that, the matrix of all text tokens and image patches $P_{ace}(x, y)$ can be implemented efficiently by Pointwise Convolution (Howard et al., 2017), reducing the computation complexity to $O(MNB_pB_tG)$, which can be omitted since B_p, B_t and G are small enough.

3.2.3 Cross-modal Relative Position Embedding

Sec. 3.2.2 provides the multi-group relative position of each text token and image patch. The pairwise anchor-based relative position is then embedded with a learnable matrix $W \in \mathbb{R}^{G \times D}$,

$$E_{ace}(x_m, y_n) = P_{ace}(x_m, y_n)W \quad (8)$$

Which is called ACE-RPE in this paper. Obviously, the proposed ACE-RPE is a specific case of RPE, where the distance of the images and texts is calculated with an anchor strategy and represented by a G-dimensional vector. Then, the distances are projected to learnable position embedding and the same distance enforces the same position embedding. Consequently, the t2t RPE in NLP, p2p RPE

in CV and t2p/p2t RPE in cross-modal tasks are united in a unified form, as formulated in Eqn. 7.

Detailedly presented in Sec. A.3, we propose two different cross-attention modes interacting with ACE-RPE, i.e., the bias mode and the contextual mode. By default, we use the contextual mode in this paper.

3.3 Pre-training Objectives

The pre-training of our models involves optimizing four objectives jointly, *i.e.*, the proposed anchor loss for anchor locating, Masked Language Modeling (MLM) for text embedding, Masked Image Modeling (MIM) for image embedding, Image-Text Matching (ITM) for cross-modal matching, as shown in Fig. 3.

Anchor Loss is optimized during pre-training for better anchor locating. Noted in Eqn. 1, it enhances the similarity of anchors and their matching image patches by token-wise contrastive learning, exclusively ignores unmatched pairs through log-sum-exp trick.

Masked Language Modeling (MLM) predicts the masked words with both contextual text tokens and image patches. It aims to learn better text embedding by injecting extra contextual information in image patches. In this part, we conduct the MLM with a masking probability of 15% and take the output text embedding of cross-encoder to predict the masked tokens.

Masked Image Modeling (MIM) predicts raw pixel values of the randomly masked image patches by a lightweight one-layer head. Following (Xie et al., 2021), we implement this task by optimizing the ℓ_1 loss between raw pixel values and the output of the prediction head.

Image-Text Matching (ITM) is to predict whether an image-text pair is positive (matched) or negative (unmatched), and further capture the contextual correlation between vision and language. It is a binary classification task while taking the embedding of the [CLS] token as a joint representation of the image-text pair.

4 Experiments

In this section, we first provide numerical analyses of the proposed ACE-RPE method compared with widely used baselines on 5 cross-modal tasks, including 6 benchmarks. Then, we make a detailed ablation study to analyze the contribution of each component of the proposed ACE-RPE method.

4.1 Pre-training Setup

Pre-training Datasets Following ALBEF (Li et al., 2021), the pre-training datasets are constructed with four public-released datasets, including two web datasets (Conceptual Captions (Sharma et al., 2018), SBU Captions (Ordonez et al., 2011)), and two in-domain datasets (MS-COCO (Lin et al., 2014) and Visual Genome (Krishna et al., 2017)). The entire pre-training dataset contains about 4.0M unique images and 5.1M image-text pairs.

Implementation Details Our ACE-RPE method contains 163.7M parameters, including a text encoder of 66.6M linguistic transformer (Devlin et al., 2018), an image encoder of 43.8M ViT-B/16 (Dosovitskiy et al., 2020) and a cross-modal encoder of 53.3M transformer (Devlin et al., 2018). It is notable that, the text encoder is constructed with the first 6 layers of the original $\text{BERT}_{\text{base}}$. Presented in Fig. 3, the pre-trained objectives are composed of three tasks: Masked Language Modeling (MLM) (Li et al., 2021) for text embedding, Masked Image Modeling (MIM) (Xie et al., 2021) for image embedding (Li et al., 2021), and Image-Text Matching (ITM) for cross-modal modeling. Our model is pre-trained for 30 epochs with a batch size of 512 on 8 NVIDIA A100 GPUs. We use AdamW (Loshchilov and Hutter, 2017) setting the weight decay as 0.02. The initial learning rate is 10^{-4} and decayed to 10^{-6}, using a cosine schedule (Loshchilov and Hutter, 2016). We use RandAugment (Cubuk et al., 2020) as the image augmentation strategy, and then scale the augmented image to the resolution of 256×256. We also utilize the momentum distillation proposed in ALBEF (Li et al., 2021) and the queue size is 65, 536. By default, the hyper-parameters are set as $B_t = 5$, $B_p = 9$, $\lambda = 2$, $\delta = 0.05$ and $G = 8$, respectively.

4.2 Downstream Cross-modal Tasks

We conduct comprehensive experimental comparison on 5 cross-modal tasks, including: 1) Image-Text Retrieval on MS-COCO (Lin et al., 2014) and Flickr30K (Plummer et al., 2015); 2) Visual Entailment on SNLI-VE (Xie et al., 2019); 3) Visual Reasoning on NLVR2 (Suhr et al., 2018); 4) Visual Question Answering on VQA (Goyal et al., 2017) and 5) Weakly-supervised Visual Grounding on RefCOCO+ (Yu et al., 2016).

Image-Text Retrieval Image-Text Retrieval refers to retrieving the most relative images given a query text, and vice versa. We evaluate our methods on

Cross-modal Position Embedding	Pre-trained Images	Flickr30K (1K test set)						MS-COCO (5K test set)					
		TR			IR			TR			IR		
		R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
None	4M	94.3	99.5	99.8	83.0	96.8	98.4	72.6	91.2	95.7	56.5	81.3	89.1
APE	4M	94.5	99.6	99.9	83.2	97.0	98.4	73.0	91.3	95.8	56.7	81.5	89.2
RPE	4M	94.4	99.5	99.9	83.2	97.1	98.4	73.2	91.4	95.9	56.7	81.7	89.3
APE + RPE	4M	94.5	99.6	99.9	83.3	97.2	98.5	73.2	91.5	96.0	56.9	81.8	89.3
Uniform†	4M	94.6	99.6	99.9	83.3	97.3	98.5	73.3	91.6	96.0	56.9	81.9	89.4
ACE-RPE	4M	95.2	99.6	99.9	83.5	97.3	98.6	73.9	92.0	96.5	57.6	82.0	90.1
ACE-RPE+$\mathcal{L}_{ace}$	4M	95.4	99.7	99.9	84.0	97.6	98.9	74.2	92.2	96.8	57.9	82.4	90.2
ACE-RPE+ $\mathcal{L}_{ace}$	14M*	**96.7**	**99.9**	**100.0**	**87.0**	**97.8**	**99.1**	**78.9**	**95.2**	**97.7**	**61.4**	**85.3**	**91.0**

† : calculates the distance of all words and patches by a uniform distance without the guidance of "anchor".
* : extended with extra pre-training dataset CC12M (Changpinyo et al., 2021).

Table 1: Comparison in the Image-Text Retrieval task on Flickr30K and MS-COCO. For text retrieval (TR) and image retrieval (IR), we report the Top-1 Recall (R@1), Top-5 Recall (R@5) and Top-10 Recal (R@10). The FLOPs of our ACE-RPE model is 122G, which has just 6.1% computational overhead compared with "None" version (115G FLOPs).

two benchmarks MS-COCO (Lin et al., 2014) and Flickr30K (Plummer et al., 2015). Following ALBEF (Li et al., 2021), the resolution of image crops is increased to 384×384 for more fine-grained retrieval. During finetuning, we employ ITM in Fig. 3 to predict whether the input images and texts are matched.

Visual Entailment Visual Entailment is to predict the relationship of image-text pairs, i.e., entailment, neutral, or contradictory. The SNLI-VE (Xie et al., 2019) dataset is taken as our Visual Entailment benchmark. We follow UNITER (Chen et al., 2020a) and consider Visual Entailment as a three-way classification problem and predict the class probabilities using a multi-layer perceptron on the [CLS] token.

Visual Reasoning The goal of Visual Reasoning is also to predict the relationship of the given texts and images. However, each input pair contains two images and one text, where the text is correlated with both of the images. The model should learn to identify the statement of the text for the given images is right or not. It is conducted on NLVR2 (Suhr et al., 2018) in this paper.

Visual Question Answering Given an image, Visual Question Answering requires the model to predict the answer of a question. For fair comparison with ALBEF (Li et al., 2021), we consider this task as an answer generation task on the VQA (Goyal et al., 2017) benchmark. In detail, an additional 6-layer transformer is applied to generate the answer, while receiving the cross-modal embeddings through the cross-modal encoder in Fig. 3.

Weakly-supervised Visual Grounding Visual Grounding (in RefCOCO+ (Yu et al., 2016)) is to localize the region of an image that corresponding to a given textual description. We follow a weakly-supervised setting (Li et al., 2021), where the model is finetuned with the same strategy as image-text retrieval task, and outputs the heatmaps by Grad-CAM (Selvaraju et al., 2017).

4.3 Comparison with Baseline Methods

In this part, we conduct 4 downstream cross-modal tasks (except for RefCOCO+) to compare the proposed ACE-RPE with the baseline methods, including 1) APE method (Dosovitskiy et al., 2020); 2) RPE method (Dosovitskiy et al., 2020); 3) a unified method combining APE and RPE (Wu et al., 2021). It is remarkable that among all methods, our ACE-RPE is the only cross-modal position embedding. The mentioned APE, RPE and their combined version are all conducted for each modality separately. They are simply concatenated together, and then injected into the cross-modal encoder. Furthermore, we also conduct a uniformed version of our ACE-RPE, where the distances of all words and patches are naively calculated by a uniform distance without the guidance of "anchor".

Cross-modal Position Embedding	VQA		SNLI-VE		NLVR	
	dev	std	dev	test	dev	test
None	73.2	73.6	79.2	79.5	79.9	80.5
APE	73.9	74.1	80.2	80.7	80.6	81.0
RPE	73.8	73.9	79.4	79.6	80.3	80.7
APE+RPE	73.9	74.1	80.1	80.9	80.5	80.8
Uniform†	74.1	74.2	80.3	81.0	80.5	80.9
ACE-RPE	74.9	75.1	81.1	81.4	81.3	81.7
ACE-RPE + $\mathcal{L}_{ace}$	75.4	75.7	81.4	82.0	81.7	81.9
ACE-RPE + $\mathcal{L}_{ace}$*	**76.8**	**76.9**	**82.0**	**82.5**	**83.1**	**83.6**

† : calculates the distance of all words and patches by a uniform distance without the guidance of "anchor".
* : pretrained on CC12M (Changpinyo et al., 2021).

Table 2: Evaluation of the proposed methods on VQA (Goyal et al., 2017), Visual Entailment (SNLI-VE (Xie et al., 2019)) and Visual Reasoning (NLVR (Suhr et al., 2018)) tasks. "dev" and "std" in VQA are the test-dev and test-std datasets.

Numerical results are presented in Table 1 and

Methods	Pre-trained Images	Flickr30K (1K test set)						MS-COCO (5K test set)					
		TR			IR			TR			IR		
		R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
UNITER	4M	87.3	98.0	99.2	75.6	94.1	96.8	65.7	88.6	93.8	52.9	79.9	88.0
VILLA	4M	87.9	97.5	98.8	76.3	94.2	96.8	-	-	-	-	-	-
OSCAR	4M	-	-	-	-	-	-	70.0	91.1	95.5	54.0	80.8	88.5
ALIGN	1.2B	95.3	99.8	100.0	84.9	97.4	98.6	77.0	93.5	96.9	59.9	83.3	89.8
ALBEF	4M	94.3	99.4	99.8	82.8	96.7	98.4	73.1	91.4	96.0	56.8	81.5	89.2
ALBEF	14M	95.9	99.8	100.0	85.6	97.5	98.9	77.6	94.3	97.2	60.7	84.3	90.5
Ours	4M	95.4	99.7	99.9	84.0	97.6	98.9	74.2	92.2	96.8	57.9	82.4	90.2
Ours	14M	**96.7**	**99.9**	**100.0**	**87.0**	**97.8**	**99.1**	**78.9**	**95.2**	**97.7**	**61.4**	**85.3**	**91.0**

Table 3: Experimental results of Image-Text Retrieval on Flickr30K and MS-COCO.

Table 2. It is shown that, in the task of Image-Text Retrieval (Table 1), our proposed ACE-RPE could enhance the performance of backbones by large margins. Specifically, compared with baseline cross-modal position embedding, i.e., None position embedding counterparts, our methods improve the performance over 1.1% and 1.0% R@1 in the "TR" and "IR" on Flickr30K. Similar gains in "TR" and "IR" on MS-COCO are up to 1.6% and 1.4%. It is worth noting that, these gains are achieved with the same backbone networks and same pre-training dataset. Meanwhile, while trained on a larger dataset with 14M samples, our model achieves two new SOTA performances on both Flickr30K and MS-COCO.

Method	VQA		SNLI-VE		NLVR	
	dev	std	dev	test	dev	test
VisualBERT (Li et al., 2019)	70.8	71.0	-	-	67.4	67.0
VL-BERT (Su et al., 2020)	71.2	-	-	-	-	-
LXMERT (Tan and Bansal, 2019)	72.4	72.5	-	-	74.9	74.5
12-in-1 (Lu et al., 2020)	73.2	-	-	77.0	-	78.9
UNITER (Chen et al., 2020b)	72.7	72.9	78.6	78.3	77.2	77.9
VL-BART/T5 (Cho et al., 2021)	-	71.3	-	-	-	73.6
ViLT (Kim et al., 2021)	70.9	-	-	-	75.2	76.2
OSCAR (Li et al., 2020)	73.2	73.4	-	-	78.1	78.4
VILLA (Gan et al., 2020)	73.6	73.7	79.4	79.0	78.4	79.3
ALBEF (Li et al., 2021) (4M)	74.5	74.7	80.1	80.3	80.2	80.5
ALBEF (Li et al., 2021) (14M)	75.8	76.0	80.8	80.9	82.6	83.1
ACE-RPE(4M)	74.9	75.1	81.1	81.4	81.3	81.7
ACE-RPE + $\mathcal{L}_{ace}$ (4M)	75.4	75.7	81.4	82.0	81.7	81.9
ACE-RPE + $\mathcal{L}_{ace}$ (14M)	76.8	76.9	**82.0**	**82.5**	**83.1**	**83.6**

Table 4: Comparison with SOTA works on VQA, SNLI-VE and NLVR benchmarks. "dev" and "std" in VQA are the test-dev and test-std datasets.

For the tasks of Visual Question Answering on VQA, Visual Entailment on SNLI-VE and Visual Reasoning on NLVR , the proposed ACE-RPE also outperforms baseline methods robustly, as shown in Table 2. Furthermore, the comparison between "ACE-RPE" and "ACE-RPE + $\mathcal{L}_{ace}$" reveals that the proposed $\mathcal{L}_{ace}$ is key for the performance improvement of ACE-RPE.

4.4 Comparison with SOTA Methods

Table 3, Table 4 and Table 5 report the results of the proposed ACE-RPE and previous SOTA methods. Pretrained on the dataset with 4M images, our methods achieve absolute improvements over ALBEF of 1.1% R@1 in "TR" and 1.2% R@1 in "IR" on Flickr30K. Similar gains in R@1 "TR" and "IR" on MS-COCO are up to 1.1% and 1.1%. For Visual Entailment, Visual Reasoning and Weakly-supervised Visual Grounding tasks, ACE-RPE also outperforms existing methods by substantial margins. With the 14M pre-trained dataset, which is also used in ALBEF, our method achieves 5 new SOTA results on all benchmarks [1], which presents the superiority and robustness of our ACE-RPE.

Method	Val	TestA	TestB
ARN (Liu et al., 2019)	32.8	34.4	32.1
CCL (Zhang et al., 2020)	34.3	36.9	33.6
ALBEF (Li et al., 2021)	58.5	65.9	46.3
ACE-RPE(4M)	59.4	66.6	47.1
ACE-RPE + $\mathcal{L}_{ace}$ (4M)	60.1	67.5	47.9
ACE-RPE + $\mathcal{L}_{ace}$ (14M)	**60.5**	**67.9**	**48.2**

Table 5: Weakly-supervised visual grounding on RefCOCO+ benchmark.

4.5 Visualization of ACE-RPE

In order to reveal the inherent ability of the proposed ACE-RPE to model the cross-modal positional information, we provide Grad-CAM visualization (Selvaraju et al., 2017; Li et al., 2021) of the anchor-based relative position in the last cross-modal transformer. Fig. 4 shows some examples in MS-COCO. The visualization of cross-modal locating is highly correlated with human priors, which indicates the correctness of our ACE-RPE.

5 Conclusion

In this paper, we present a cross-modal position embedding method, called ACE-RPE, in which we first utilize an anchor locating method to learn to match the text words and the image patches.

[1] Except for VQA, where the champion achieved the best score of 82.78 according to https://eval.ai/web/challenges/challenge-page/830/leaderboard/2278. But we think it is not fair to compare the methods in Table 4 with the champion because of different pretrained datasets and great finetuning gap.

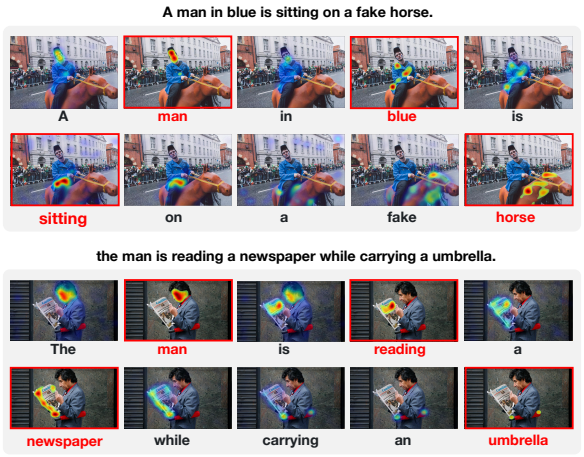


Figure 4: The Grad-CAM (Selvaraju et al., 2017) visualization of cross-modal distance on the last cross-attention layer. The words in red are the anchors.

Then, we compute physical distances between anchors and tokens from different modalities, which are applied for cross-modal fusion. We conduct comprehensive experiments to analyze the effectiveness of different components of ACE-RPE as well as the performance under different modes and hyper-parameter settings. As we know, this work is the first to present position embeddings for cross-modal tasks, and the experimental results also demonstrate the superiority of our method.

Limitations

Though the proposed ACE-RPE method achieves significant and substantial performance on 6 benchmarks. However, it has two major limitations: 1) the ACE-RPE is injected into backbone model during both pretraining and finetuning procedures. As we know, pretraining is much more time-consuming than finetuning. It will be more efficient to be implemented if it can maintain comparable results by simply initializing our models with a public released pretrained model, and only finetuning our models in downstream tasks. That is to say, the ACE-RPE is only employed in the finetuning model. We think it is worthy of more experimental results to study this kind of implementation. 2) The experiments in this paper are conducted on 8 NVIDIA A100 GPUs, which is expensive for personal researchers.

References

Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. 2021. Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3558–3568.

Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020a. Uniter: Universal image-text representation learning. In *European conference on computer vision*, pages 104–120. Springer.

Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020b. UNITER: universal image-text representation learning. In *ECCV*, volume 12375, pages 104–120.

Jaemin Cho, Jie Lei, Hao Tan, and Mohit Bansal. 2021. Unifying vision-and-language tasks via text generation. *arXiv preprint arXiv:2102.02779*.

Xiangxiang Chu, Zhi Tian, Bo Zhang, Xinlong Wang, Xiaolin Wei, Huaxia Xia, and Chunhua Shen. 2021. Conditional positional encodings for vision transformers. *arXiv preprint arXiv:2102.10882*.

Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. 2020. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 702–703.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov. 2019. Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Zhe Gan, Yen-Chun Chen, Linjie Li, Chen Zhu, Yu Cheng, and Jingjing Liu. 2020. Large-scale adversarial training for vision-and-language representation learning. In *NeurIPS*.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6904–6913.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738.

Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. 2017. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.

Wonjae Kim, Bokyung Son, and Ildoo Kim. 2021. Vilt: Vision-and-language transformer without convolution or region supervision. *arXiv preprint arXiv:2102.03334*.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123(1):32–73.

Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. 2021. Align before fuse: Vision and language representation learning with momentum distillation. *Advances in Neural Information Processing Systems*, 34.

Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. 2019. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*, abs/1908.03557.

Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, Yejin Choi, and Jianfeng Gao. 2020. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *ECCV*, pages 121–137.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer.

Xuejing Liu, Liang Li, Shuhui Wang, Zheng-Jun Zha, Dechao Meng, and Qingming Huang. 2019. Adaptive reconstruction network for weakly supervised referring expression grounding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2611–2620.

Ilya Loshchilov and Frank Hutter. 2016. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Jiasen Lu, Vedanuj Goswami, Marcus Rohrbach, Devi Parikh, and Stefan Lee. 2020. 12-in-1: Multi-task vision and language representation learning. In *CVPR*, pages 10434–10443.

Frank Nielsen and Ke Sun. 2016. Guaranteed bounds on information-theoretic measures of univariate mixtures using piecewise log-sum-exp inequalities. *Entropy*, 18(12):442.

Vicente Ordonez, Girish Kulkarni, and Tamara Berg. 2011. Im2text: Describing images using 1 million captioned photographs. *Advances in neural information processing systems*, 24.

Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. 2015. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Proceedings of the IEEE international conference on computer vision*, pages 2641–2649.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. 2017. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626.

Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. 2018. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565.

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. 2018. Self-attention with relative position representations. *arXiv preprint arXiv:1803.02155*.

Aravind Srinivas, Tsung-Yi Lin, Niki Parmar, Jonathon Shlens, Pieter Abbeel, and Ashish Vaswani. 2021. Bottleneck transformers for visual recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16519–16529.

Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. 2020. Vl-bert: Pre-training of generic visual-linguistic representations. In *ICLR*.

Alane Suhr, Stephanie Zhou, Ally Zhang, Iris Zhang, Huajun Bai, and Yoav Artzi. 2018. A corpus for reasoning about natural language grounded in photographs. *arXiv preprint arXiv:1811.00491*.

Hao Tan and Mohit Bansal. 2019. Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490*.

Yao-Hung Hubert Tsai, Shaojie Bai, Paul Pu Liang, J Zico Kolter, Louis-Philippe Morency, and Ruslan Salakhutdinov. 2019. Multimodal transformer for

unaligned multimodal language sequences. In *Proceedings of the conference. Association for Computational Linguistics. Meeting*, volume 2019, page 6558. NIH Public Access.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Benyou Wang, Lifeng Shang, Christina Lioma, Xin Jiang, Hao Yang, Qun Liu, and Jakob Grue Simonsen. 2020. On position embeddings in bert. In *International Conference on Learning Representations*.

Kan Wu, Houwen Peng, Minghao Chen, Jianlong Fu, and Hongyang Chao. 2021. Rethinking and improving relative position encoding for vision transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10033–10041.

Qi Wu, Damien Teney, Peng Wang, Chunhua Shen, Anthony Dick, and Anton van den Hengel. 2017. Visual question answering: A survey of methods and datasets. *Computer Vision and Image Understanding*, 163:21–40.

Ning Xie, Farley Lai, Derek Doran, and Asim Kadav. 2019. Visual entailment: A novel task for fine-grained image understanding. *arXiv preprint arXiv:1901.06706*.

Zhenda Xie, Zheng Zhang, Yue Cao, Yutong Lin, Jianmin Bao, Zhuliang Yao, Qi Dai, and Han Hu. 2021. Simmim: A simple framework for masked image modeling. *arXiv preprint arXiv:2111.09886*.

Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. 2016. Modeling context in referring expressions. In *European Conference on Computer Vision*, pages 69–85. Springer.

Qinglong Zhang and Yu-Bin Yang. 2021. Rest: An efficient transformer for visual recognition. *Advances in Neural Information Processing Systems*, 34.

Zhu Zhang, Zhou Zhao, Zhijie Lin, Xiuqiang He, et al. 2020. Counterfactual contrastive learning for weakly-supervised vision-language grounding. *Advances in Neural Information Processing Systems*, 33:18123–18134.

A Appendices

A.1 Shared *V.S.* Unshared

ACE-RPE could also be used in a shared mode for fewer parameters. In this part, we conduct experiments with shared ACE-RPE and compared the results with the unshared version. Table 6 shows that shared ACE-RPE would result in a slight performance drop on Image-Text Retrieval and Visual Reasoning task.

Mode	Flickr30K		MS-COCO		NLVR	
	TR	IR	TR	IR	dev	test
Shared	98.1	93.2	87.4	76.7	81.2	81.5
Unshared	98.3	93.5	87.7	76.8	81.7	81.9

Table 6: Ablation study on Image-Text Retrieval and Visual Reasoning task. The average recall on the test set is reported on Flickr30K and MS-COCO.

A.2 Robustness on Hyper-parameters

The default hyper-parameters of the proposed method are: $\lambda = 2$, $\delta = 0.05$ and $G = 8$. Table 7 presents the performance comparison of different choice of these hyper-parameters. Anchor loss with larger λ (Eqn. 1) forces the model to learn more about the most similar anchor, while smaller ones reduce to predict more possible anchors. δ serves as the threshold parameter to select the anchors, and G is the number of groups in the proposed multi-head distance. It is shown that, λ and G influence the performance more significantly compared with δ. It is also indicated that as G is greater than 8, the performance of ACE-RPE maintains almost unchanged.

MS-COCO	λ				δ				G				
	1	2	3	4	0.01	0.05	0.1	0.2	1	4	8	16	32
TR	85.9	87.7	87.6	87.5	87.5	87.7	87.5	87.6	87.1	87.6	87.7	87.7	87.6
IR	75.0	76.8	76.6	76.6	76.5	76.8	76.7	76.7	76.3	76.6	76.8	76.9	76.8

Table 7: Ablation study on Image-Text Retrieval task on MS-COCO. The average recall on the test set is reported.

A.3 Bias *V.S.* Contextual Modes

ACE-RPE presents the position embedding of each text word and image patch. In this part, we propose two different cross-attention modes interacting with ACE-RPE, i.e., the bias mode and the contextual mode.

Bias Mode In this mode, ACE-RPE has no explicit interaction with the query, key or value in the transformer block. Instead, it functions as the bias of the cross-attention block. Formally,

$$\begin{cases} \mathcal{F}_{i2t}(x,y) = \frac{(xW^Q)(yW^K)^T + E_{ace}(x,y)W_E}{\sqrt{D}} \\ \mathcal{F}_{t2i}(y,x) = \frac{(yW^Q)(xW^K)^T + E_{ace}(x,y)W_E}{\sqrt{D}} \end{cases} \quad (9)$$

where $\mathcal{F}_{i2t}$ and $\mathcal{F}_{t2i}$ are the image-to-text and text-to-image cross-attention, respectively. $E_{ace}(x,y) \in \mathbb{R}^{M\times N\times D}$ is a 3-dimensional tensor, denoting the ACE-RPE between all text tokens and image patches. W^Q and W^K are learnable matrices. $W_E \in \mathbb{R}^D$ is a learnable vector, which maps $E_{ace}(x,y)$ into a 2-dimensional matrix.

Contextual Mode ACE-RPE in contextual mode is first flatten into 2-dimension by average pooling, then added with the token/patch embedding,

$$\begin{cases} \overline{x}_i = x_i + \mathbb{E}_{j=1}^{N} E_{ace}(x_i, y_j) \\ \overline{y}_i = y_i + \mathbb{E}_{i=1}^{M} E_{ace}(x_i, y_j) \end{cases} \quad (10)$$

The cross-attention is then,

$$\begin{cases} \mathcal{F}_{i2t}(\overline{x},\overline{y}) = \frac{(\overline{x}W^Q)(\overline{y}W^K)^T}{\sqrt{D}} \\ \mathcal{F}_{t2i}(\overline{y},\overline{x}) = \frac{(\overline{y}W^Q)(\overline{x}W^K)^T}{\sqrt{D}} \end{cases} \quad (11)$$

In this case, ACE-RPE interacts with the queries, keys in a cross-attention block. Besides, it can also be applied to value embeddings,

$$\begin{cases} Z_{i2t}(\overline{x},\overline{y}) = \sigma(\mathcal{F}_{i2t}(\overline{x},\overline{y}))(\overline{y}W^V + E_{ace})^T \\ Z_{t2i}(\overline{y},\overline{x}) = \sigma(\mathcal{F}_{t2i}(\overline{y},\overline{x}))(\overline{x}W^V + E_{ace})^T \end{cases} \quad (12)$$

Here, $\sigma(\cdot)$ presents the softmax function, and W^V is a learnable matrix. E_{ace} is $E_{ace}(x,y)$ for short.

Experimental Result In this part, we compare the performances of two cross-modal modes, i.e., "Bias" and "Contextual" modes. Table 8 illustrates the numerical results in Image-Text Retrieval and Visual Reasoning task. Using the proposed ACE-RPE in contextual mode is demonstrated to be a better way.

Mode	Flickr30K		MS-COCO		NLVR	
	TR	IR	TR	IR	dev	test
Bias	98.1	93.4	87.4	76.6	81.5	81.6
Contextual	98.3	93.5	87.7	76.8	81.7	81.9

Table 8: Ablation study on Image-Text Retrieval and Visual Reasoning task. The average recall on the test set is reported on Flickr30K and MS-COCO.

A.4 Component-wise Analysis

Inspired by (Wu et al., 2021), in the field of image processing, the position embedding interacts with the calculation of the query, key and value in the self-attention layer. Accordingly, we analyze the result of each choice in cross-modal modeling, and the results are shown in Table 9. It is shown

that ACE-RPE calculated on values could only get slight gains over the version without ACE-RPE, but the ones embedded on queries and values would result in significant performance gains.

Position			Flickr30K		MS-COCO		NLVR	
query	key	value	TR	IR	TR	IR	dev	test
×	×	×	97.8	92.7	86.5	75.6	79.9	80.5
✓	×	×	98.1	93.3	87.4	76.6	81.4	81.6
×	✓	×	98.1	93.2	87.5	76.7	81.4	81.5
×	×	✓	97.8	92.8	86.7	76.0	80.7	81.0
✓	✓	×	98.2	93.3	87.5	76.6	81.6	81.8
✓	✓	✓	98.3	93.5	87.7	76.8	81.7	81.9

Table 9: Ablation study on Image-Text Retrieval and Visual Reasoning. The average recall on the test set is reported on Flickr30K and MS-COCO.