



PCATNet: Position-Class Awareness Transformer for Image Captioning

Ziwei Tang¹, Yaohua Yi^{2,*}, Changhui Yu² and Aiguo Yin³

¹Research Center of Graphic Communication, Printing and Packaging, Wuhan University, Wuhan, 430072, China

²School of Remote Sensing and Information Engineering, Wuhan University, Wuhan, 430072, China

³Zhuhai Pantum Electronics Co., Ltd., Zhuhai, 519060, China

*Corresponding Author: Yaohua Yi. Email: whudcil@whu.edu.cn

Received: 18 November 2023; Accepted: 07 March 2023

Abstract: Existing image captioning models usually build the relation between visual information and words to generate captions, which lack spatial information and object classes. To address the issue, we propose a novel Position-Class Awareness Transformer (PCAT) network which can serve as a bridge between the visual features and captions by embedding spatial information and awareness of object classes. In our proposal, we construct our PCAT network by proposing a novel Grid Mapping Position Encoding (GMPE) method and refining the encoder-decoder framework. First, GMPE includes mapping the regions of objects to grids, calculating the relative distance among objects and quantization. Meanwhile, we also improve the Self-attention to adapt the GMPE. Then, we propose a Classes Semantic Quantization strategy to extract semantic information from the object classes, which is employed to facilitate embedding features and refining the encoder-decoder framework. To capture the interaction between multi-modal features, we propose Object Classes Awareness (OCA) to refine the encoder and decoder, namely OCA_E and OCA_D , respectively. Finally, we apply GMPE, OCA_E and OCA_D to form various combinations and to complete the entire PCAT. We utilize the MSCOCO dataset to evaluate the performance of our method. The results demonstrate that PCAT outperforms the other competitive methods.

Keywords: Image captioning; relative position encoding; object classes awareness

1 Introduction

Image captioning is the research to generate human descriptions for images [1–3]. Recently, image captioning makes great progress because of improved classification [4–6], object detection [7,8] and machine translation [9]. Inspired by these, many researchers propose their methods based on the encoder-decoder framework, in which the images are encoded to features by pre-trained Convolutional Neural Network (CNN) and then decoded to sentences by Recurrent Neural Network (RNN) [10,11], Transformer [12] or Bert [13] models. In addition, the attention mechanism has been proposed to help the model build relevance between image regions and the generated sentence [14–17]. Therefore,



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

the concentration of improving image caption can be summarized as two aspects: (1) optimizing the image representation [14,17–19], including the visual feature, position and classes information, and (2) improving the process of image representation by modifying the structure [14,20].

Objectively, when a man tries to generate a sentence for an image, he will implement three steps: (1) get the region and classes of objects, (2) build the relationship among them, and (3) search the appropriate words to complete the whole caption. However, the researches on image captioning tend to overlook the first step and focus on the latter steps to construct directly the relationship between the visual features and words [12,20–23]. According to the latest research, the image can be represented by object-attribute region-based features [14] or grid features [19] whose classes and position information are dropped. Recently, Wu et al. [24] revisit the position encoding for visual Transformer and demonstrate that excellent relative/absolute position encoding can improve the performance of visual features for object recognition & detection. Nevertheless, object positions often fall into disuse for image captioning, which inevitably results in the loss of spatial information. Besides, Li et al. [25] propose the feature pairs to solve the problem between the image features and language features, and then apply the big-data pre-training to generate a corpus which is so time-consuming with the Bert model.

With the enlightenment from the aforementioned works, we propose the Position-Class Awareness Transformer (PCAT) network for image captioning, where the network is transformer-based with distinctive position encoding and structure of class feature embedding. On the one hand, we propose a relative position encoding method to quantize the spatial information to vectors for CNN-based visual features. Then, we embed these quantized vectors into the Self-attention [12] (SA) module to ameliorate the relation among objects for the encoder phase. On the other hand, we embed class names as the language vectors and reconstruct the Transformer, which can build the semantics relationship among the objects and narrow the gap from the vision to captions, to adopt the class information from detected objects.

In the paper, we exploit Transformer to construct our framework. In the encoder, a novel relative position encoding method is proposed to model the relationships among the objects and update it to the Self-attention modules. Simultaneously, we construct an extra feature processing module to obtain the semantic association of classes in an image. In the decoder, we improve the block units by adding an independent attention unit, which can bridge the gap from caption features to visual features. We employ the MSCOCO dataset and perform quantitative and qualitative analyses to evaluate our method. The experiment results demonstrate that our method achieves competitive performance with 138.3% CIDEr scores.

The contributions include:

1. We propose the Position-Class Awareness Transformer (PCAT) network to boost image captioning by the spatial information and detected object classes.
2. We propose a relative position encoding method, namely Grid Mapping Position Encoding (GMPE), intuitively measuring the distances of the objects for the Self-attention module, to strengthen the correlation and subordination.
3. We propose a Classes Semantic Quantization strategy to improve the representation of class names and refine the encoder-decoder framework by Object Classes Awareness (OCA) to model the interaction between vision and language.

2 Related Work

2.1 Image Captioning

Solutions for image captioning are proposed upon the encoder-decoder in recent years. For example, Vinyals et al. [26] propose the CNN-LSTM architecture to encode the image into features and decode them into a caption. Anderson et al. apply the two-layer Long Short-Term Memory (LSTM) network to concentrate on the weighting stage. These methods all employ the RNN-based decoder, which may lose relevance if the two generating words have a large step interval [23]. Until Google proposes the Transformer [12] which applies the Self-attention to calculate the similarity matrix between vision and language, image captioning is trapped in this issue.

Transformer is still an encoder-decoder framework consisting of the attention and Feed Forward Network. Upon this, some optimized Transformers are proposed to obtain better features by improving the structure of the model. M²-Transformer [22] encodes image regions and their relationships into a multi-layer structure to fuse both shallow and deep relationships. Then, the generation of sentences adopts a multi-level structure by low- and high-level visual relations, which is better than the application of single semantic features. However, M²-Transformer is an optimization of Transformer, it still researches the feature and can't solve the splitting problem of cross-modal feature conversion for image captioning. To address this issue, X-Transformer [20] focuses on the interaction between image and language by spatial and channel bilinear attention distribution. According to this improvement, X-Transformer achieves excellent performance in 2020. Furthermore, Zhang et al. [19] propose RSTNet to count the contribution of visual features and context features while generating fine-grained captions by novel adaptive attention. Meanwhile, RSTNet is the first to apply the grid features of the image for image captioning and obtain excellent performance. Since 2021, many pre-training methods for image captioning are proposed. For example, Zhang et al. [18] and Li et al. [25] research the visual representation of an image and propose the grid features and pre-training strategy of visual objects features respectively. The pre-training methods apply big data to construct relationships between the visual features and language features and achieve powerful performance for image captioning.

2.2 Self-Attention and Position Encoding in Transformer

Self-attention is the sub-unit of Transformer, which maps the query, key and value to the output. Moreover, for each input token $v_i \in \mathbb{R}^d$, the Self-attention can output a corresponding sequence $z_i \in \mathbb{R}^d$ which can be computed as follows:

$$z_i = \sum_{j=1}^n \partial_{i,j} (v_j W^V) \quad (1)$$

$$\partial_{i,j} = \frac{\exp(\delta_{i,j})}{\sum_{k=1}^n \exp(\delta_{i,k})} \quad (2)$$

$$\delta_{i,j} = \frac{(v_i W^Q)(v_j W^K)^T}{\sqrt{d}} \quad (3)$$

where the projections $W^Q, W^K, W^V \in \mathbb{R}$ are trainable matrixes, ∂ is the SoftMax function and δ is the scaled dot-product attention.

The position encoding methods that we discuss are the absence of the image captioning encoder. As we know, the position encoding is initially designed to generate the order of sequence for the embedding token [12] named absolute position encoding, which can be formulated as:

$$v_i = v_i + p_i \quad (4)$$

where the p_i is the positional encodings and $v_i \in \mathbb{R}^d$. There are several methods to accomplish the encoding such as the sine and cosine functions and the learnable parameters [12,27].

Besides the absolute position encoding, researchers recently reconsider the pairwise relationships between the tokens. Relative position encoding is significant for the tasks that request distance or sequence to measure the association [24,28]. The relative position between tokens v_i and v_j is encoded to $p_{ij}^Q, p_{ij}^K, p_{ij}^V \in \mathbb{R}^d$ and embedded into the Self-attention, which can be defined as:

$$v_i^Q = v_i \mathbf{W}^Q + p_{ij}^Q \quad (5)$$

$$v_j^K = v_j \mathbf{W}^K + p_{ij}^K \quad (6)$$

$$v_j^V = v_j \mathbf{W}^V + p_{ij}^V \quad (7)$$

Although relative position encoding has been widely applied in object detection, it is hardly employed in image captioning. Considering the semantic information of the relative distance, we believe it can advance the interaction of vision and language.

3 The Proposed Method

The architecture of the PCAT network is presented in Fig. 1. We use the transformer-based framework. The encoder includes the N refining blocks which are in charge of embedding position and objects classes information to capture the relationship among detected objects. The decoder applies the image features and reconstructs the blocks to embed the object classes information between the captions and visual features to bridge them.

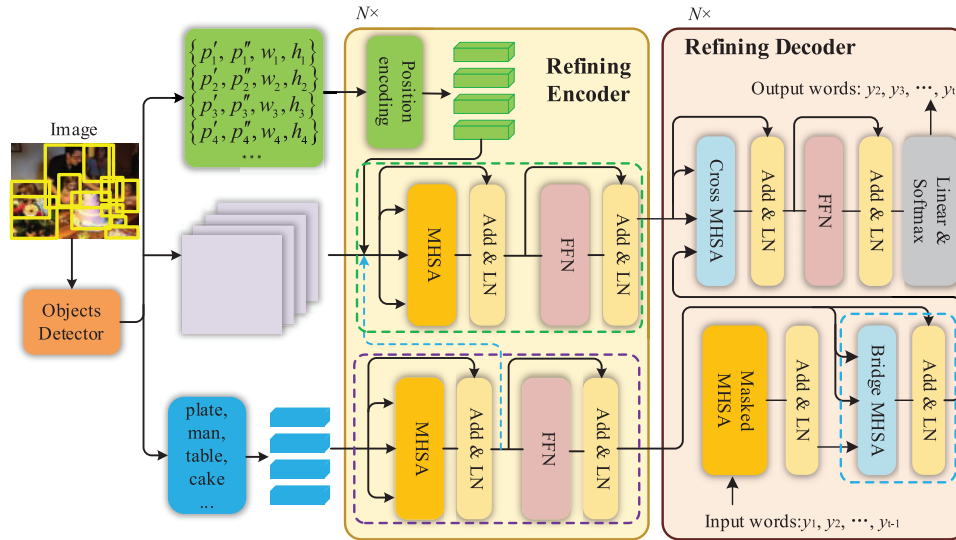


Figure 1: Overview of our proposed PCAT network

3.1 Grid Mapping Position Encoding

As the extra information of the objects in an image, position encoding is always ignored for image captioning. To re-weigh the attention by the spatial information, we design a novel learnable spatial grid feature map to improve position encoding, as well as update the Self-attention to adopt it.

Given the objects detected in an image I , we first extract their center point position, height and width of regions as $P_i = \{p'_i, p''_i, h_i, w_i\}$. Then, we design a learnable spatial grid feature map M and set its size to $m \times m$, as shown in Fig. 2, and apply the data process θ to work out the absolute distance encoding d_i^A via the objects' positions P_i and the M , which can be formulated as:

$$d_i^A = M_i(\theta(p'_i), \theta(p''_i)) + M_{bias}(p'_i, p''_i, h_i, w_i) \quad (8)$$

where $M_i(\cdot)$ denotes the position grid feature with indexes (the blue box in Fig. 2) and M_{bias} is the compensation of the extra region which can't be covered by a grid (the shaded area in Fig. 2). Finally, we apply the Euclidean distance and linearization to compute relative distance and obtain the relative distance encoding $p_{i,j}$.

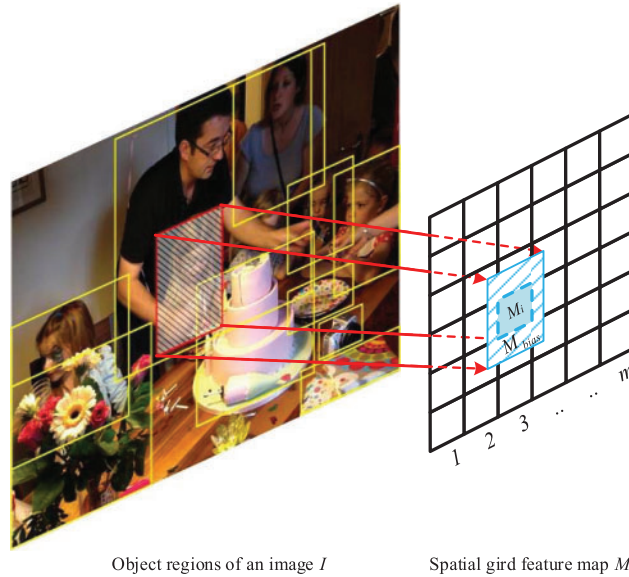


Figure 2: Illustration of Grid Mapping

As the spatial information of objects, we refine the Self-attention of the encoder (which is inspired by the contextual mode in [24]) to embed the relative position encodings $p_{i,j}$, as shown in the encoder in the green box of Fig. 1 and the optimized Self-attention in Fig. 3.

Considering the interaction of visual features of objects v , we regard the relative position encoding $p_{i,j}$ as the bias for the similarity matrix $v_i^Q (v_j^K)^T$ of query and key. Therefore, δ in Eq. (3) can be refined as:

$$\delta_{i,j} = \frac{(v_i W_p^Q) (v_j W_p^K)^T + (v_i W_p^Q) (p_{i,j})^T}{\sqrt{d}} \quad (9)$$

where the projection $W_p^Q \in \mathbb{R}$ is trainable matrixes.

Grid Mapping. The reason that Transformer for image captioning abandons the position encoding while encoding is that image is 2D and the regions of objects are not a sequence. To calculate the

position on a 2D image and define the absolute distance encoding d_i^A , we propose an undirected mapping method, for the process θ and feature map M_i & M_{bias} .

Considering that each grid of feature map M possesses a fixed index which can be represented as a 2D sequence, what we should concentrate on is that an object region covers how many grids. Unfortunately, we can hardly map an object region to only a grid region entirely. Thus, we define a parameter M_{bias} to represent the partial feature map of neighbor grids. According to this issue, we first calculate the corresponding indexes of the covered grids for each of the detected regions and find out the center of the covered grids (the blue box in Fig. 2), which can be formulated as:

$$\theta(p'_i) = \left\lceil \frac{p'_i}{\frac{h_I}{m}} \right\rceil \quad (10)$$

$$\theta(p''_i) = \left\lceil \frac{p''_i}{\frac{w_I}{m}} \right\rceil \quad (11)$$

where h_I and w_I refer to the height and width of I . Then, we collect the 8-neighbor grids of the computed M_i and the process can be followed as:

$$M_{bias} = \sum_{n=1}^8 (\lambda_n M_i^n) \quad (12)$$

where λ_n is the intersection between the n th-neighbor grid and M_{bias} , calculated by $P_i = \{p'_i, p''_i, h_i, w_i\}$. Note that if the 8-neighbor grids may not cover the target entirely, we can calculate the all covered grid features for M_{bias} , except the center grid, by the approach of concentric circles with different weights. The 8-neighbor is the normal situation of concentric circles. Therefore, we can obtain the absolute position encoding d_i^A .

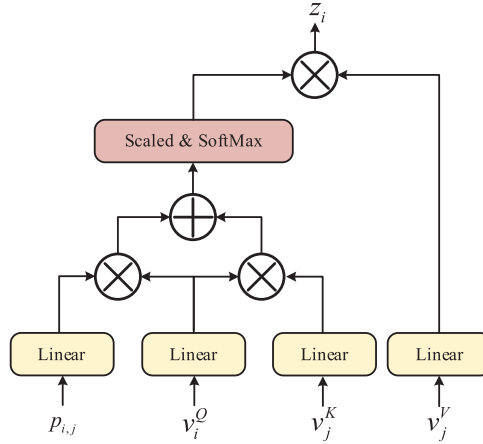


Figure 3: Illustration of optimized Self-attention with position encoding

Relative Position Encoding. The relative position encoding $p_{i,j}$ is determined by the relative distance calculation and the correlation between objects. Note that anyone relative distance isn't mapped into an integer because the semantic distance between two objects can't altogether be replaced with position distance. Therefore, we follow Eq. (8) to obtain the two mapped absolute distance encodings d_i^A and d_j^A , and measure their relative distance encoding $d_{i,j}^R$. Considering that the order of the words responding

objects can't be predicted while generating, we regard mathematics vectors d_i^A and d_j^A as p_i^A and p_j^A to measure their Euclidean distance and linearize them into the encoding $p_{i,j}$ influenced by captions:

$$d_{i,j}^R = \sqrt{(p_i^A - p_j^A)^2} \quad (13)$$

$$p_{i,j} = d_{i,j}^R \mathbf{W}_R \quad (14)$$

where $\mathbf{W}_R \in \mathbb{R}$ is a trainable matrix.

3.2 Object Classes Awareness Methods

In contrast to the positional encodings, classes are the semantic information of objects. Classes are explained as the two different forms of token: (a) they are the attributes of the visual objects; (b) they are the sources of words in the captions. According to these two characteristics, we propose the Classes Semantic Quantization strategy to quantize the objects-classes word, as well as the Object Classes Awareness network (OCA) to refine the encoder-decoder framework.

Classes Semantic Quantization

The objects-classes are essentially the words w_{cls} . To quantize them and to ensure they hold the same semantics field as the captions, we utilize the words dictionary (Section 4.1) from the dataset to quantize the w_{cls} to word embedding vectors v^{cls} :

$$v^{cls} = w_{cls} \mathbf{W}_e \quad (15)$$

where \mathbf{W}_e is the word embedding matrix. Note that some classes are word groups $g_{cls} = \{w_{cls}^1, w_{cls}^2, \dots, w_{cls}^k | k \in N^*\}$, such as “hot dog”, “traffic lights” and “fire hydrant”. If the word group can be represented by the core word, for example the “fire hydrant” is almost equivalent to the word “hydrant”, we will crop the auxiliary word. Besides, if the word group retains the new semantics different from any word, we will add their vectors together. The process can be defined as a piecewise function:

$$v^{cls} = \begin{cases} w_{cls}^n \mathbf{W}_e, & \forall w_{cls}^n \approx g_{cls} \\ \sum_{n=1}^k (w_{cls}^n \mathbf{W}_e), & \exists w_{cls}^n \neq g_{cls} \end{cases} \quad (16)$$

where g_{cls} represents the word group and $n \in \{1, 2, \dots, k\}$.

Encoder

For the encoder, we quantize the class words to the vectors v^{cls} and accept them as the tokens which are homologous with the visual features v of detected objects. We refine the encoder by improving the encoder block. The OCA module (the purple box in Fig. 1) is proposed to build the relationship among classes, which is identical to the Self-attention for the v . We apply the extra multi-head Self-attention (MHSA), residual structure and layer-normalization (LayerNorm) to model the semantic relationship of objects, which can be defined as:

$$z_{n_{head}}^{cls} = SA_{n_{head}}(v^{cls}, v^{cls}, v^{cls}) \quad (17)$$

$$Z_{head}^{cls} = [z_1^{cls}, z_2^{cls}, \dots, z_{n_{head}}^{cls}] \mathbf{W}_E^0 \quad (18)$$

$$Z^{cls} = \text{LayerNorm}(v^{cls} + \text{ReLU}(Z_{head}^{cls})) \quad (19)$$

where n_{head} means the n -th head of the MHSA, the projection $\mathbf{W}_E^0 \in \mathbb{R}$ is a learnable matrix, the $[\cdot]$ is the concatenation of the vectors, ReLU is the activation function and $Z_{head}^{cls}, Z^{cls} \in \mathbb{R}^d$ are the results

of MHSA for classes vectors and LayerNorm respectively. Besides, we have to calculate them without sharing layers because v^{Cls} possess a different modality from the v .

In this phase, we propose a fusion strategy to embed class information for the vision encoder, namely OCA_E . We provide Z_{head}^{Cls} for the visual features v to add semantic information and help the model optimize relationships among objectives. As the blue dashed in Fig. 1, we input Z_{head}^{Cls} and v simultaneously. Considering the improved Self-attention with position encoding, we can just renovate the v with Z_{head}^{Cls} , defined as:

$$v = v + Z_{head}^{Cls} \quad (20)$$

Decoder

The decoder aims to generate the final captions with the visual feature and class information from the encoder. As shown in the blue box of Fig. 1, we refine the decoder with an additional feature processing module that can embed class information between language features and visual features, namely OCA_D . The refining decoder consists of N blocks each of which can be divided into three modules: (a) Language Masked MHSA Module, which can achieve the interaction in the generated words; (b) Bridge MHSA Module (words-to-classes), which includes an MHSA, the residual connection and a LayerNorm and can be regarded as the interaction between caption words and detected objects names; (c) Cross MHSA Module (classes-to-vision), which contains an MHSA, a feed-forward Network (FFN), the residual connections, the LayerNorms, a linear and a SoftMax function and generates the caption word by word eventually.

Language Masked MHSA Module. We apply this module to build the relations (words-to-words) among the words $y_{1:t-1}$ that can be represented as:

$$\tilde{y}_{t-1} = \text{LayerNorm} \left(y_{t-1} + \text{MHSA} (y_{t-1} W_t^Q, y_{t-1} W_t^K, y_{t-1} W_t^V) \right) \quad (21)$$

where $W_t^Q, W_t^K, W_t^V \in \mathbb{R}$ are learnable matrixes and y_{t-1} indicates the vectors of the word at $(t-1)$ -th step.

Bridge MHSA Module (words-to-classes). Because a detected object itself corresponds to a region and the class of this object is semantic, we propose the structure of words-to-classes-to-vision. This module aims to model the relationship between words \tilde{y}_{t-1} and class features Z^{Cls} . Therefore, we construct bridge attention to capture the class context information, which denotes the primary multi-modal interaction to bridge language and vision by classes and can be formulated as:

$$\kappa_{t-1} = \text{MHSA} \left(\tilde{y}_{t-1} W_{we}^Q, Z^{Cls} W_{we}^K, Z^{Cls} W_{we}^V \right) \quad (22)$$

$$\tilde{Z}_{t-1}^{Cls} = \text{LayerNorm} \left(Z^{Cls} + \kappa_{t-1} W_{wc}^0 \right) \quad (23)$$

where $W_{we}^Q, W_{we}^K, W_{we}^V, W_{wc}^0 \in \mathbb{R}$ are learnable matrixes, $\tilde{Z}_{t-1}^{Cls} \in \mathbb{R}^{(t-1) \times d}$ denotes the output of the Bridge MHSA with the \tilde{y}_{t-1} and is exploited as the input of Cross MHSA Module (classes-to-vision).

Cross MHSA Module (classes-to-vision). This module aims at modeling the relationship between the attended classes \tilde{Z}_{t-1}^{Cls} and visual features Z , which refers to another multi-modal crossing to bridge language and vision by MHSA with classes and vision. The process can be given by:

$$\lambda_{t-1} = \text{MHSA} \left(\tilde{Z}_{t-1}^{Cls} W_{cv}^Q, Z W_{cv}^K, Z W_{cv}^V \right) \quad (24)$$

$$\tilde{Z}_{t-1} = \text{LayerNorm} (Z + \lambda_{t-1} W_{cv}^0) \quad (25)$$

$$y_{t-1}^{cv} = \text{LayerNorm} \left(\tilde{Z}_{t-1} + \text{FFN} \left(\tilde{Z}_{t-1} \right) \right) \quad (26)$$

where $W_{cv}^Q, W_{cv}^K, W_{cv}^V, W_{cv}^0 \in \mathbb{R}$ are learned parameters, \tilde{Z}_{t-1}^{Cls} from the former module is input into MHSA as query, and visual features Z which are composed of position information from the encoder are fed into MHSA as key and value.

The distribution of the vocabulary is as follows:

$$p(y_t | y_{1:t-1}) = \text{Softmax} (y_{t-1}^{cv} W^y) \quad (27)$$

where $W^y \in \mathbb{R}$ is a learnable matrix.

3.3 Training and Objectives

Train by Cross-Entropy Loss. First, we train our model by the Cross-Entropy Loss L_{XE} :

$$L_{XE}(\theta) = - \sum_{t=1}^T \log (p_{\theta} (y_t^* | y_{1:t-1}^*)) \quad (28)$$

where $y_{1:T}^*$ represents the ground truth.

Optimize by CIDEr Score. Then, we employ Self-Critical Sequence Training (SCST) [29] to optimize:

$$L_{RL}(\theta) = -\mathbf{E}_{y_{1:T} \sim p_{\theta}} [r(y_{1:T})] \quad (29)$$

where the reward $r(\cdot)$ is computed by CIDEr (Consensus-based Image Description Evaluation) [30]. The gradient can be defined as:

$$\nabla_{\theta} L_{RL}(\theta) \approx - (r(y_{1:T}^s) - r(\hat{y}_{1:T})) \nabla_{\theta} \log p_{\theta} (y_{1:T}^s) \quad (30)$$

where y^s refers to the result of sampled probability and the \hat{y} means the result of the greedy algorithm.

4 Experiment

4.1 Dataset and Implementation Details

We apply the MSCOCO dataset [31] to conduct experiments. The dataset has 123287 images (82783 for training and 40775 for validation) with 5 captions for each. We adopt the Karpathy split [32] to obtain the training set, the validation set and the testing set. Besides, we collect the words that occur more than 4 times in the training set and get a dictionary containing 10369 words. The metrics of BLEU (Bilingual Evaluation Understudy) [33], CIDEr [30], METEOR (Metric for Evaluation of Translation with Explicit ORdering) [34], SPICE (Semantic Propositional Image Caption Evaluation) [35] and ROUGE-L (Recall-Oriented Understudy for Gisting Evaluation) [36] are applied to evaluate our method. We compute these metrics with the public code from the MSCOCO dataset.

Differing from the image grid features, we demand accurate object classes and position information for our framework. Therefore, we exploit the Objects365 [37], MSCOCO [31], OpenImages [38] and Visual Genome [39] datasets to train the Faster-Rcnn model [7] for extracting objects features, and merge their classes to obtain a label list with more than 1800 classes, which is similar to VinVL [18]. These objects' visual vectors are extracted in 2048-dimension and transformed into 512-dimension vectors to match the embedding size. The number of block N is set to 6. With Cross-Entropy Loss,

we adopt the learning rate of $4e-4$ decayed 0.8 every 2 epochs and ADAM during the total 20 epochs. While training with CIDEr Score Optimization in another 30 epochs, we set the learning rate to $4e-5$ and decay it by 50%. Furthermore, the batch size is 10 and the beam size is 2.

4.2 Comparisons with Other Models

We report the performances of the other methods and our method in Table 1. The compared methods include Show&tell (LSTM) [26], SCST [29], RFNet [11], UpDown [14], AoANet [40], Pos-aware [41], M²-Transformer [22], X-Transformer [20], RSTNet [19] and PureT [42]. These methods are operated with LSTM or Transformer.

We adopt the strategy of pre-training for the visual feature in VinVL [18] and Transformer [12] as our baseline. Therefore, our baseline achieves good scores because of the great pre-training of detection which also provides accurate position and class information for the proposed method.

For stability, we first present the results of a single model in Table 1. Our models with XE Loss and SCST training are both superior to others. With the XE Loss training, our single model with different terms (OCA_E and OCA_D) achieves the highest scores in all metrics, especially the CIDEr score which obtains advancement of over 1% to the X-Transformer and AoANet. With the SCST training, our models also achieve the best comprehensive performance. While comparing with the strong competitors M²-Transformer, X-Transformer and RSTNet, our two models are superior to them in all terms of metrics, especially the CIDEr score improved by over 2%. Besides, the BLEU-4 score of our methods reach 41.2% and 41.6% which achieve improvements of 0.3% and 0.7% to the latest PureT, respectively. Meanwhile, our methods surpass PrueT in terms of all metrics except METEOR.

Table 1: The results of our method and other methods. The B, M, R, C and S represent the metrics of BLEU, METEOR, ROUGE-L, CIDEr and SPICE. *indicates the results that we reproduce based on VinVL

Method	B-1	B-2	B-3	B-4	M	R	C	S
Trained by cross-entropy loss								
Baseline [18]*	76.7	61.3	47.1	36.7	28.2	57.1	118.5	20.9
LSTM [26]	—	—	—	29.6	25.2	52.6	94.0	—
SCST [29]	—	—	—	30.0	25.9	53.4	99.4	—
Adaptive-Attention [15]	73.4	56.6	41.8	30.4	25.7	—	102.9	—
RFNet [11]	76.4	60.4	46.6	35.8	27.4	56.5	112.5	20.5
UpDown [14]	77.2	—	—	36.2	27.0	56.4	113.5	20.3
AoANet [40]	77.4	—	—	37.2	28.4	57.5	119.8	21.3
X-Transformer [20]	77.3	61.5	47.8	37.0	28.7	57.5	120.0	21.8
PCATNet (w/ OCA_E)	77.7	61.8	47.8	37.4	28.8	57.6	122.3	22.1
PCATNet (w/ OCA_D)	77.8	61.7	47.8	37.2	28.7	57.5	121.2	21.9
Optimized by CIDEr Score Optimization								
Baseline [18]*	81.9	66.9	52.1	40.3	29.8	59.6	135.5	23.2
LSTM [26]	—	—	—	31.9	25.5	54.3	106.3	—
SCST [29]	—	—	—	34.2	26.7	55.7	114.0	—

(Continued)

Table 1: Continued

Method	B-1	B-2	B-3	B-4	M	R	C	S
UpDown [14]	79.8	–	–	36.3	27.7	56.9	120.1	21.4
AoANet [40]	80.2	–	–	38.9	29.2	58.8	129.8	22.4
Pos-aware [41]	80.8	65.1	50.6	39.3	29.0	59.2	128.9	22.8
M ² -Transformer [22]	80.8	–	–	39.1	29.2	58.6	131.2	22.6
X-Transformer [20]	80.9	65.8	51.5	39.7	29.5	59.1	132.8	23.4
RSTNet [19]	81.8	–	–	40.1	29.8	59.5	135.6	23.3
PureT [42]	82.1	–	–	40.9	30.2	60.1	138.2	24.2
PCATNet (w/OCA_E)	82.6	67.7	53.2	41.2	29.9	60.2	137.8	24.0
PCATNet (w/OCA_D)	82.6	67.7	53.3	41.6	30.0	60.3	138.3	24.3

In addition, we report the results of the ensemble of four models with SCST in Table 2. Our method also achieves excellent performance and advances the M²-Transformer and RSTNet by more than 6% in terms of CIDEr. Furthermore, our method and PureT are about equal in performance, as outlined in the case of the single model. We also present some generated captions in Table 3 to demonstrate the performance of our approach.

Table 2: The ensemble results of four models

Method	B-4	M	R	C	S
SCST [29]	35.4	27.1	56.6	117.5	–
RFNet [11]	37.9	28.3	58.3	125.7	21.7
M ² [22]	40.5	29.7	59.5	134.5	23.5
PureT [42]	42.1	30.4	60.8	141.0	24.3
PCATNet (w/OCA_E)	42.3	30.2	61.0	140.6	24.2
PCATNet (w/OCA_D)	42.4	30.2	61.2	140.9	24.4

4.3 Ablative Studies

We conduct ablative experiments to understand the influence of each module in our model.

Influence of GMPE. To quantify the influence of GMPE in the refined encoder, we conduct experiments with different modules. We adopt 6 blocks of encoder and decoder and set the size of grid feature $m \times m$ to 16×16 . Note that we adopt the Transformer as our baseline in row 1. In Table 4, we evaluate the performance of GMPE in three combinations including the baseline with GMPE (row 2), OCA_E with GMPE (row 5) and OCA_D with GMPE (row 6). As we can see, combining baseline and GMPE can achieve improvements of 1.4% in terms of the CIDEr score to the pure baseline. Furthermore, GMPE can increase the CIDEr score of pure OCA_D from 136.7% to 138.3% and improve the performance of OCA_E from 137.1% to 137.8%.

Table 3: Visualization for generated captions of GroundTruth, Baseline and our method


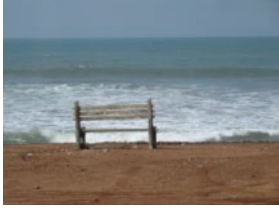

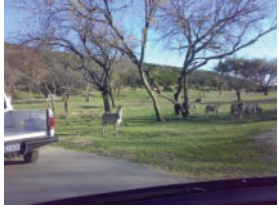
			
GT: three people sit at a table holding lollipops	GT: a wooden bench sitting on a beach next to the ocean	GT: a man on a snowboard standing at the bottom of the mountain	GT: several zebras are on the grass by a truck
Baseline: a group of people sitting at a table with a birthday cake	Baseline: a bench on a beach near the ocean	Baseline: a man holding a snowboard in the snow	Baseline: a herd of zebras standing in a field next to a car
Ours (w/OCA_E): a group of people sitting at a table with lollipops	Ours (w/OCA_E): a wooden bench sitting on the beach next to the water	Ours (w/OCA_E): a man standing on a snowboard in the snow	Ours (w/OCA_E): a herd of zebras standing on the side of a truck
Ours (w/OCA_D): three people sitting at a table	Ours (w/OCA_D): a wooden bench sitting on a beach next to the ocean	Ours (w/OCA_D): a man standing on a snowboard on the slopes	Ours (w/OCA_D): a group of zebras grazing in the grass next to a truck

Table 4: The results of ablation studies, which are obtained after CIDEr Score Optimization

GMPE	OCA _E	OCA _D	B-4	R	C
×	×	×	39.8	59.6	135.5
✓	×	×	40.9	60.0	136.9
×	✓	×	40.8	60.1	137.1
×	×	✓	41.0	59.9	136.7
✓	✓	×	41.2	60.2	137.8
✓	×	✓	41.6	60.3	138.3
✓	✓	✓	40.6	59.7	136.4

Influence of OCA_E and OCA_D. To better understand the influence of OCA_E and OCA_D in encoder and decoder respectively, we conduct several experiments to evaluate them. Note that the number of block N is set to 6 and $m \times m$ is set to 16×16 . In Table 4, we present the results of OCA_E in rows 3, 5 and 7, as well as the performance of OCA_D in rows 4, 6 and 7. It can be seen that the baseline with only OCA_E or OCA_D (row 3 and row 4) can achieve improvements of 1.6% and 1.2% in terms of CIDEr score, respectively. Besides, the CIDEr score of OCA_E and OCA_D combined with GMPE (row 5 and row 6) can reach 137.8% and 138.3%, which achieve improvements of 0.9% and 1.4% to the baseline

with GMPE (row 2), respectively. However, while combining OCA_E and OCA_D , we obtain a poor record (row 7) resulting from too much specific class information which can fragment the generated captions.

Influence of the Number of Blocks. We fine-tune the number of refining encoder-decoder blocks N and the size of the grid feature $m \times m$. Note that we adopt the baseline while experimenting on N and the baseline with GMPE for $m \times m$. As shown in Table 5, the baseline has a continuous improvement in the CIDEr score with the gradual increase of the value N . Besides, the baseline model tends to be stable when N is set to 6. Therefore, we set N to 6 as the final configuration. The baseline with GMPE also gets a significant improvement in all metrics and reaches peak performance while 16×16 . Nevertheless, we still believe in the effectiveness of the other sizes and don't suggest setting the size lower than 11×11 , because the small size can result in too many objects in one grid and lose the advantage of GMPE.

Table 5: Experiments about the number of block N and the size of grid feature $m \times m$

N	B-1	B-4	M	R	C
3	81.3	39.7	29.4	58.8	133.8
4	81.6	39.8	29.6	59.2	134.4
5	81.8	40.0	29.6	59.5	135.2
6	81.9	40.3	29.8	59.6	135.5
7	81.9	40.2	29.8	59.5	135.4
$m \times m$	B-1	B-4	M	R	C
9×9	81.9	40.4	29.7	59.6	135.5
11×11	82.1	40.5	29.7	59.8	136.2
14×14	82.2	40.8	29.8	59.8	136.6
16×16	82.2	40.9	29.9	60.0	136.9
18×18	82.2	40.9	29.9	59.9	136.8

5 Conclusion

In this paper, we propose a novel Position-Class Awareness Transformer network, which can embed more information, such as spatial and classes of objects, from an image to relate vision with language. To achieve this purpose, the GMPE module and OCA module are proposed, which are designed by spatial information and object classes respectively. The proposed GMPE, a relative position encoding method for embedding spatial correlations, constructs a grid mapping feature to calculate the relative distance among objects and quantizes them to the vectors. Moreover, we propose the OCA to refine the encoder-decoder framework, which can model the correlation between visual features and language features by the extracted semantic information of object classes. Formally, we also associate the GMPE with the OCA. Experiment results demonstrate that our method can significantly boost captioning, where GMPE supplies the model with spatial information and OCA bridges the visual features and language features. In particular, our method achieves excellent performance against other methods and provides a novel scheme for embedding information.

In the future, we will explore how to generate captions with object classes directly and further develop relative position encoding with direction for image captioning. With the information of object classes, we will attempt at combining generated word and objects classes name, which can provide more semantic information for the next generating word. Furthermore, we plan to improve the proposed GMPE with the directions among objects and the semantics of captions, which can capture more interaction among objects by associating the language module.

Acknowledgement: The numerical calculations in this paper have been done on the supercomputing system in the Supercomputing Center of Wuhan University.

Funding Statement: This work was supported by the National Key Research and Development Program of China [No. 2021YFB2206200].

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] G. Kulkarni, V. Premraj, S. Dhar, S. Li, Y. Choi *et al.*, “Baby talk: Understanding and generating simple image descriptions,” in *Proc. CVPR*, Colorado Springs, CO, USA, pp. 1601–1608, 2011.
- [2] Q. Yang, Z. Ni and P. P. Ren, “Meta captioning: A meta learning based remote sensing image captioning framework,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 186, pp. 190–200, 2022.
- [3] W. H. Jiang, M. W. Zhu, Y. M. Fang, G. M. Shi, X. W. Zhao *et al.*, “Visual cluster grounding for image captioning,” *IEEE Transactions on Image Processing*, vol. 31, pp. 3920–3934, 2022.
- [4] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh *et al.*, “ImageNet large scale visual recognition challenge,” *International Journal of Computer Vision*, vol. 115, pp. 211–252, 2015.
- [5] A. Srinivas, T. Y. Lin, N. Parmar, J. Shlens, P. Abbeel *et al.*, “Bottleneck Transformers for visual recognition,” in *Proc. CVPR*, Nashville, TN, USA, pp. 16514–16524, 2021.
- [6] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai *et al.*, “An image is worth 16×16 words: Transformers for image recognition at scale,” arXiv preprint arXiv:2010.11929, 2010.
- [7] S. Ren, K. He, R. Girshick and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017.
- [8] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov *et al.*, “End-to-end object detection with transformers,” in *Proc. ECCV*, pp. 213–229, 2020.
- [9] I. Sutskever, O. Vinyals and Q. V. Le, “Sequence to sequence learning with neural networks,” in *Proc. NIPS*, Montreal, Canada, pp. 3104–3112, 2014.
- [10] K. Cho, B. V. Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares *et al.*, “Learning phrase representations using RNN Encoder-Decoder for statistical machine translation,” arXiv preprint arXiv:1406.1078, 2014.
- [11] W. Jiang, L. Ma, Y. G. Jiang, W. Liu and T. Zhang, “Recurrent fusion network for image captioning,” in *Proc. ECCV*, Munich, Germany, pp. 510–526, 2018.
- [12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones *et al.*, “Attention is all you need,” in *Proc. NIPS*, Long Beach, California, USA, pp. 6000–6010, 2017.
- [13] J. Devlin, M. W. Chang, K. Lee and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” arXiv preprint arXiv:1810.04805, 2018.
- [14] P. Anderson, X. He, C. Buehler, D. Teney, M. Johnson *et al.*, “Bottom-up and top-down attention for image captioning and visual question answering,” in *Proc. CVPR*, Salt Lake City, USA, pp. 6077–6086, 2018.
- [15] J. Lu, C. Xiong, D. Parikh and R. Socher, “Knowing when to look: Adaptive attention via a visual sentinel for image captioning,” in *Proc. CVPR*, Hawaii, USA, pp. 3242–3250, 2017.

- [16] L. Chen, H. Zhang, J. Xiao, L. Nie, J. Shao *et al.*, “SCA-CNN: Spatial and channel-wise attention in convolutional networks for image captioning,” in *Proc. CVPR*, Hawaii, USA, pp. 6298–6306, 2017.
- [17] X. Yang, D. Q. Liu, H. W. Zhang, Y. D. Zhang and F. Wu, “Context-aware visual policy network for fine-grained image captioning,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 2, pp. 710–722, 2022.
- [18] P. Zhang, X. Li, X. Hu, J. Yang, L. Zhang *et al.*, “VinVL: Revisiting visual representations in vision-language models,” in *Proc. CVPR*, Nashville, TN, USA, pp. 5575–5584, 2021.
- [19] X. Zhang, X. Sun, Y. Luo, J. Ji, Y. Zhou *et al.*, “RSTNet: Captioning with adaptive attention on visual and non-visual words,” in *Proc. CVPR*, Nashville, TN, USA, pp. 15460–15469, 2021.
- [20] Y. Pan, T. Yao, Y. Li and T. Mei, “X-Linear attention networks for image captioning,” in *Proc. CVPR*, Seattle, USA, pp. 10968–10977, 2020.
- [21] T. Nguyen and B. Fernando, “Effective multimodal encoding for image paragraph captioning,” *IEEE Transactions on Image Processing*, vol. 31, pp. 6381–6395, 2022.
- [22] M. Cornia, M. Stefanini, L. Baraldi and R. Cucchiara, “Meshed-memory Transformer for image captioning,” in *Proc. CVPR*, Seattle, USA, pp. 10575–10584, 2020.
- [23] Z. W. Tang, Y. Yi and H. Sheng, “Attention-guided image captioning through word information,” *Sensors*, vol. 21, no. 23, pp. 7982, 2021.
- [24] K. Wu, H. Peng, M. Chen, J. Fu and H. Chao, “Rethinking and improving relative position encoding for vision Transformer,” in *Proc. ICCV*, Montreal, Canada, pp. 10013–10021, 2021.
- [25] X. Li, X. Yin, C. Li, P. Zhang, X. Hu *et al.*, “Oscar: Object-semantics aligned pre-training for vision-language tasks,” in *Proc. ECCV*, pp. 121–137, 2020.
- [26] O. Vinyals, A. Toshev, S. Bengio and D. Erhan, “Show and tell: A neural image caption generator,” in *Proc. CVPR*, Boston, USA, pp. 3156–3164, 2015.
- [27] J. Gehring, M. Auli, D. Grangier, D. Yarats and Y. N. Dauphin, “Convolutional sequence to sequence learning,” in *Proc. ICML*, Sydney, NSW, Australia, pp. 1243–1252, 2017.
- [28] P. Shaw, J. Uszkoreit and A. Vaswani, “Self-attention with relative position representations,” arXiv preprint arXiv:1803.02155, 2018.
- [29] S. J. Rennie, E. Marcheret, Y. Mroueh, J. Ross and V. Goel, “Self-critical sequence training for image captioning,” in *Proc. CVPR*, Hawaii, USA, pp. 1179–1195, 2017.
- [30] R. Vedantam, C. L. Zitnick and D. Parikh, “CIDER: Consensus-based image description evaluation,” in *Proc. CVPR*, Boston, USA, pp. 4566–4575, 2015.
- [31] T. Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona *et al.*, “Common objects in context,” in *Proc. ECCV*, Zurich, Switzerland, pp. 740–755, 2014.
- [32] A. Karpathy and L. Fei-Fei, “Deep visual-semantic alignments for generating image descriptions,” in *Proc. CVPR*, Boston, USA, pp. 3128–3137, 2015.
- [33] K. Papineni, S. Roukos, T. Ward and W. J. Zhu, “BLEU: A method for automatic evaluation of machine translation,” in *Proc. ACL*, Philadelphia, Pennsylvania, pp. 311–318, 2002.
- [34] B. Satanjeev, “METEOR: An automatic metric for MT evaluation with improved correlation with human judgments,” in *Proc. ACL*, Ann Arbor, Michigan, USA, pp. 228–231, 2005.
- [35] P. Anderson, B. Fernando, M. Johnson and S. Gould, “SPICE: Semantic propositional image caption evaluation,” in *Proc. ACM*, Scottsdale, AZ, USA, pp. 382–398, 2016.
- [36] C. Y. Lin, “ROUGE: A package for automatic evaluation of summaries,” in *Proc. ACL*, Barcelona, Spain, pp. 74–81, 2004.
- [37] S. Shao, Z. Li, T. Zhang, C. Peng, G. Yu *et al.*, “Objects365: A large-scale, high-quality dataset for object detection,” in *Proc. ICCV*, Seoul, Korea(south), pp. 8429–8438, 2019.
- [38] A. Kuznetsova, H. Rom, N. Alldrin, J. Uijlings, I. Krasin *et al.*, “The open images dataset v4,” *International Journal of Computer Vision*, vol. 128, pp. 1956–1981, 2020.
- [39] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata *et al.*, “Visual Genome: Connecting language and vision using crowdsourced dense image annotations,” *International Journal of Computer Vision*, vol. 123, pp. 32–73, 2017.

- [40] L. Huang, W. Wang, J. Chen and X. Wei, "Attention on attention for image captioning," in *Proc. ICCV*, Seoul, Korea(south), pp. 4633–4642, 2019.
- [41] Y. Duan, Z. Wang, J. Wang, Y. K. Wang and C. T. Lin, "Position-aware image captioning with spatial relation," *Neurocomputing*, vol. 497, pp. 28–38, 2022.
- [42] Y. Wang, J. Xu and Y. Sun, "End-to-end Transformer based model for image captioning," in *Proc. AAAI*, online, no.8053, 2022.