Couplformer: Rethinking Vision Transformer with Coupling Attention

Hai Lan*
FJIRSM[1]
lanhai09@mails.ucas.ac.cn

Xihao Wang*
Technical University of Munich
xihaowang2016@gmail.com

Hao Shen
fortiss GmbH
shen@fortiss.org

Peidong Liang
Fujian (Quanzhou) HIT
zqllpd@hotmail.com

Xian Wei†
FJIRSM[1]
FJOEL[2]
xian.wei@tum.de

Abstract

With the development of the self-attention mechanism, the Transformer model has demonstrated its outstanding performance in the computer vision domain. However, the massive computation brought from the full attention mechanism became a heavy burden for memory consumption. Sequentially, the limitation of memory consumption hinders the deployment of the Transformer model on the embedded system where the computing resources are limited. To remedy this problem, we propose a novel memory economy attention mechanism named Couplformer, which decouples the attention map into two sub-matrices and generates the alignment scores from spatial information. Our method enables the Transformer model to improve time and memory efficiency while maintaining expressive power. A series of different scale image classification tasks are applied to evaluate the effectiveness of our model. The result of experiments shows that on the ImageNet-1K classification task, the Couplformer can significantly decrease 42% *memory consumption compared with the regular Transformer. Meanwhile, it accesses sufficient accuracy requirements, which outperforms* 0.56% *on Top-1 accuracy and occupies the same memory footprint. Besides, the Couplformer achieves state-of-art performance in MS COCO 2017 object detection and instance segmentation tasks. As a result, the Couplformer can serve as an efficient backbone in visual tasks and provide a novel perspective on deploying attention mechanisms for researchers.*

1. INTRODUCTION

In recent years, the attention mechanism has generated considerable interest in the domain of deep learning. The main breakthroughs of attention modules first appeared in Natural Language Processing (NLP) [38] and then was transferred to the computer vision domain [10]. Different from Convolutional Neural Network(CNN), the attention module tends to exert a learnable weight on features to distinguish importance from various perspectives. Due to the outstanding extraction ability of information, the attention mechanism has immediately been leveraged in several deep learning applications and has become one of the indispensable concepts in the deep learning field [24]. With the development of the neural network, the attention mechanism has been introduced in many specific tasks, such as image caption generation [45, 23], action recognition [32, 34], image-based analysis [31], and graph [39].

As a particular form of attention, self-attention is applied as the core mechanism of the neural network, named Transformer. Recent efforts have shown that Transformer and attention-based models have become ubiquitous in modern vision tasks [10]. However, there are a few drawbacks of the Transformer model. Coming with outstanding performance, it also exposed the severe demand for computation of memory. Because the size of full attention matrix is proportional to the square of the sequence length, the time and space complexity extend considerably [3]. Some challenges result from this problem. First of all, compared with the CNN, the more significant memory consumption of the Transformer restricts its deployment possibility on terminal equipment, such as mobile robots and unmanned aerial vehicles. Secondly, due to the limitation of memory consumption, the selection of patch size tends to be larger, which adversely affects the patch representation of the Visual Transformer [35]. Moreover, with the growth of the image size, the problem of memory limitation will be

[1]Fujian Institute of Research on the Structure of Matter, Chinese Academy of Sciences
[2]Fujian Science & Technology Innovation Laboratory for Optoelectronic Information of China
*Equal technical contribution
†Corresponding author

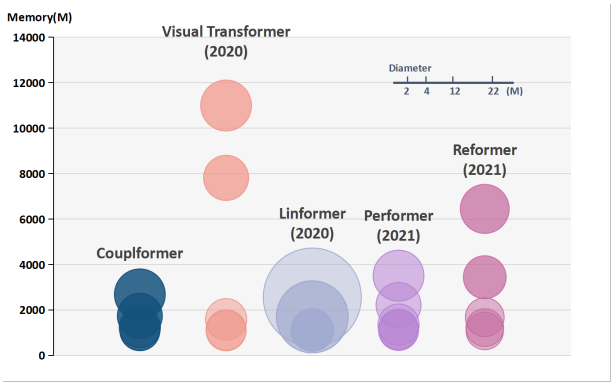


Figure 1: The proposed Couplformer achieves memory economy in the visual task. When the patch size is unvaried, the memory consumption increases with the growth of the image size(32*32, 64*64,128*128,256*256). Compared with the original Transformer, efficient Transformer models present a lower memory consumption to a relative degree. Our Couplformer keeps the most economy memory increment with the minimum training parameters. (the scatter area means the number of training parameters).

much more conspicuous. Therefore, memory consumption inhibits the Transformer model scalability in quite a lot of settings. Several attempts have been made to improve the efficiency of the Vision Transformer(see Section 2 for details). Despite their gratifying result, these methods have mainly relied on the following approaches. Depending on the low-rank prior, some works [41, 4, 17] utilize the kernel method to discover a relatively low-rank structure to reduce the memory. By limiting the attention map, some methods [26, 25] predefine the field of view to save the computation cost. Moreover, other works [8, 28] leverage extensional network architecture to improve the Transformer's efficiency.

In this paper, we introduce a novel method named Couplformer, which involves the coupling attention mechanism to alleviate the memory limitation in Vision Transformers. Specifically, we decouple the attention map into two submatrices and the alignment scores of each sub-matrix are generated with the vector along the height and width axis respectively, which is inspired by the human vision patterns [30] that human visual perception tend to capture the difference between lines during reading. From the Fig. 1, Couplformer merely requires 31.6% memory consumption of original Vision Transformer(ViT) while the image size is $256 * 256$. The experiment results show Couplformer could be applied to different scales of datasets and obtain competitive performance by training from scratch.

The main contributions of this paper include:

1. We introduced coupling attention mechanism which decouple the self-attention map into two sub-matrices to reduce the space complexity from $\mathcal{O}((hw)^2)$ to $\mathcal{O}(h^2+w^2)$. And we designed a novel way to construct the sub-matrix by calculating the alignment score between the vectors along the height and width axis rather than the channel axis.

2. We designed an efficiently algorithm for coupling attention by leveraging the vectorization trick. This algorithm can realize the process of self-attention without calculating the full attention matrix explicitly and provide a substantial speedup for both model training and inference.

3. Based on the coupling attention mechanism, we elaborated a framework named Couplformer and evaluated our model by conducting experiments on different scales of CV tasks without bells and whistles. According to the experiments, our model achieves competitive performance with relatively low memory consumption.

2. RELATED WORK

Vision Transformer Before *Dosovitskiy et al.* [10] proposed the Vision Transformer, CNNs dominated the area of visual recognition. Depending on the weight sharing, scale separation, and shift equivariant, CNNs possess the powerful and efficient ability to extract the feature from the image [12]. Although the Transformer network did not present the strong equivariance representation as CNNs, its unique structure endows its permutation equivariance to obtain inductive bias [11]. In detail, the standard Visual Transformer's structure includes the following parts: Token Embedding, Positional Embedding, Transformer Encoder, and Classification Head [18]. With the exploration of the Transformer network, various Transformer-based models were proposed to solve the vision tasks efficiently. *Touvron et al.* [36] proposed the DeiT, which uses distillation learning to overcome the drawback of ViT that it could only present the outstanding performance in large-scale datasets. CPVT [5] model applied the different position embedding methods to improve the efficiency and flexibility of the ViT model. According to the CvT [42] and CeiT [46] model, they try to hybridize the Transformer and CNN network to derive desirable properties from each one. *Liu et al.* [22] proposed the Swin Transformer, which adopts a series of approaches in terms of visual tasks, such as patch partition, linear embedding, pyramid structure, and window-based MSA.

Efficient Transformers Depending on the self-attention mechanism, the Transformer model has already become prevalent in many fields. As described above, the standard self-attention operation relies on dot-production multiplication. Sequentially, this leads to the problem that

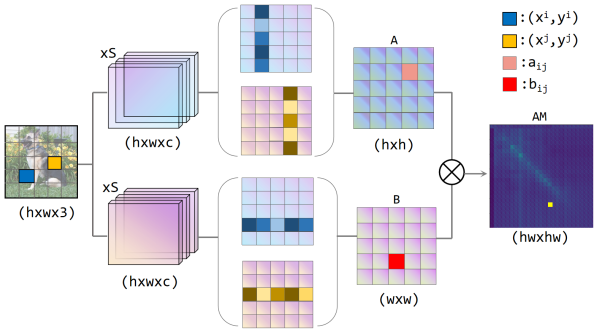


Figure 2: Coupling attention mechanism, using the product of two alignment scores of vectors along the height and width axis to obtain the attention map(the s denotes the number of heads, and other notations are introduced in Section 3).

self-attention is the quadratic time and memory complexity [33]. The reason is that the dot product between the feature matrix Q and matrix K generates a massive matrix to present the token-token interaction. In such a situation, it is unavoidable that there is exhaustive and redundant computation in the standard self-attention operation. In order to address this problem, the researchers proposed several novel Transformer architectures to improve the original self-attention mechanism, which named "efficient Transformers" [33].

The first solution is to sparsify the self-attention layers. According to the sparse attention, the attention map can only be computed by limited pairs in a particular predefined manner. For example, by limiting and fixing the field of view, *Qiu et al.* [26] employed the fixed block local attention patterns to constrain the pair for the score. Similarly, Sparse Transformer [3] and Longformer [1] leverage fixed strided attention patterns to achieve the cost reduction. *Kitaev et al.* [19] proposed a learnable approach by using hash-based similarity to replace the token-to-token interaction.

Secondly, depending on the low rank prior, employing the kernel-based method to approximate the attention matrix could also reduce the complexity. In terms of approximation solution, Linformer [41], Performer [4] and Linear Transformer [17] utilized the kernel method to avoid explicitly implementing the dot production. They attempted to find a relatively low-rank structure to reduce the memory and computational complexity.

Lastly, there are some of the other efficient Transformer architectures different from the solutions mentioned above. Depending on the segment-based recurrence, Transformer XL [8] applies a hidden state to connect adjacent blocks with a recurrent mechanism [33]. Different from the Transformer XL [8], *Rae et al.* [28] utilizes a dual memory system to maintain a fine-grained memory of past segment activations. As one kind of efficient Transformer, our model could be classified into the approximation solution to reduce the computation complexity and memory consumption.

3. COUPLFORMER

In this section, a brief review of self-attention mechanism is given firstly, then we describe the main idea of coupling attention and its efficient calculation. Finally, we introduce our elaboration of Couplformer model for image classification.

3.1. Standard Attention Mechanism

According to the standard visual attention mechanism in ViT[10], the input feature $x \in \mathbb{R}^{h\times w\times d}$ of attention module is reshaped into the flattened token $x_p \in \mathbb{R}^{L\times d}$. L denotes the length of input tokens sequence which is the product of height h and width w of the input feature. Then three linear transformations are applied on the input tokens x_p to generate the $Q, K, V \in \mathbb{R}^{L\times d}$ respectively. The output of self-attention module $O \in \mathbb{R}^{L\times d}$ can be calculated as below:

$$AM = \frac{QK^\top}{\sqrt{d}}, O = (SM(AM))V \tag{1}$$

Here $SM(\cdot)$ denotes the softmax operation along the matrix's rows. $AM \in \mathbb{R}^{L\times L}$ denotes the attention matrix, which has $O(L^2)$ space complexity. Intuitively, the quadratic dependency of the attention matrix leads to high memory consumption and limits the application of the large feature maps in the computer vision scenario.

According to the Eq. (1), that the element $am_{i,j}$ in the attention matrix AM is the dot product of $i-th$ row vector $\mathbf{q}_i$ of query matrix Q and $j-th$ row vector $\mathbf{k}_j$ of key matrix K, here i and j denote the indexes of flattened token sequence, and i and j can be calculated by the 2D coordinates $(x^{(i)}, y^{(i)}), (x^{(j)}, y^{(j)})$ of the specified tokens on the 2D feature map as below:

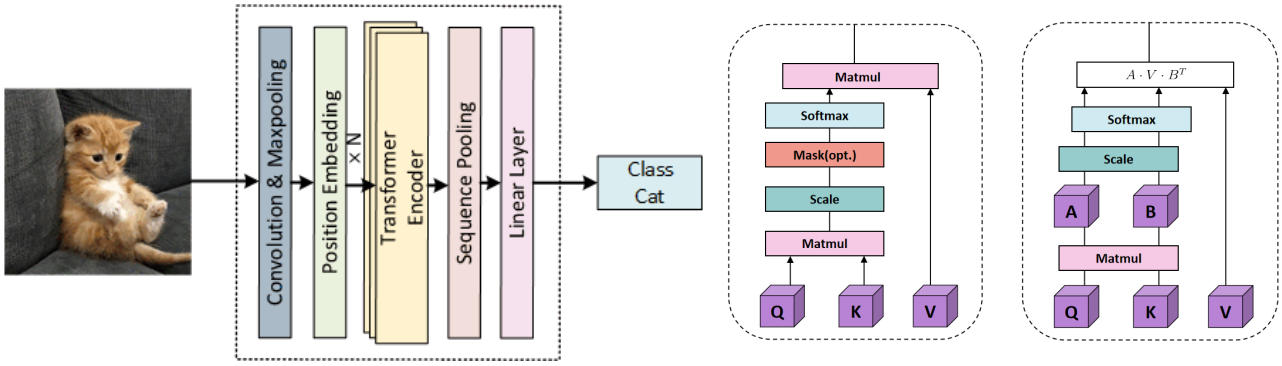
$$\begin{cases} i = x^{(i)} + y^{(i)} * w \\ j = x^{(j)} + y^{(j)} * w \end{cases} \tag{2}$$

3.2. Coupling Attention Mechanism

Inspired by the idea of applying decomposition or dimension reduction to attention matrix to reduce the complexity of attention mechanism [44, 41, 19], we assume the attention map $\mathbf{AM} \in \mathbb{R}^{hw\times hw}$ can be approximately decoupled into two sub-matrix $\mathbf{A} \in \mathbb{R}^{h\times h}$ and $\mathbf{B} \in \mathbb{R}^{w\times w}$.

Hypothesis 1

$$\boldsymbol{AM} \approx \widehat{\boldsymbol{AM}} = \boldsymbol{A} \otimes \boldsymbol{B}$$



(a) Architecture (b) Dot-Product Attention (c) Couplformer Attention

Figure 3: Illustrating the Couplformer architecture.

Here $\otimes$ denotes the tensor product operator, which can be implemented by Kronecker product while $\mathbf{A}$ and $\mathbf{B}$ are two matrices, explicitly the Kronecker product of $\mathbf{A}$ and $\mathbf{B}$ is defined by:

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{0,0}\mathbf{B} & \cdots & a_{0,h-1}\mathbf{B} \\ \vdots & \ddots & \vdots \\ a_{h-1,0}\mathbf{B} & \cdots & a_{h-1,h-1}\mathbf{B} \end{bmatrix} \tag{3}$$

The lowercase $a_{*,*}$ denotes the element in the matrix $\mathbf{A}$, and the indexes of matrix elements are all starting from zero. Because the sub-matrix $B \in \mathbb{R}^{w \times w}$, by the definition in Eq. (3), the element in the coupling attention matrix $\widehat{\mathbf{AM}}$ can be written as:

$$\widehat{am_{i,j}} = (\mathbf{A} \otimes \mathbf{B})_{i,j} = a_{[i//w, j//w]} \times b_{[i\%w, j\%w]} \tag{4}$$

Here $\widehat{am_{i,j}}$ denotes the element in the coupling attention matrix $\widehat{\mathbf{AM}}$, which represents the alignment score of any two tokens from $\mathbf{Q}$ and $\mathbf{K}$. The subscript of am_{ij} denotes the 1D indexes i and j of given tokens. The operator $//$ denotes exact division, and $\%$ is modulo operation in Eq. (4). Considering the $am_{i,j}$ in the standard attention mechanism is the inner product of $\mathbf{q}_i$ and $\mathbf{k}_j$, we substitute the i, j in the 2 into Eq. (4), hence the element of coupling attention matrix can be represented by the spatial coordinates on the 2D feature map as:

$$\widehat{am_{i,j}} = a_{[y^{(i)}, y^{(j)}]} \times b_{[x^{(i)}, x^{(j)}]} \tag{5}$$

3.3. Spatial Attention and Channel Merge

In the previous section, we decouple the original attention matrix into two sub-matrix $\mathbf{A}$ and $\mathbf{B}$, then establish the connection between the coupling attention matrix and spatial coordinate of given tokens on 2D feature map. Here comes the question, how can we construct the sub-matrix $\mathbf{A} \in \mathbb{R}^{h \times h}$ and $\mathbf{B} \in \mathbb{R}^{w \times w}$ while retaining the attention mechanism to capture query-key correlation? To solve this problem, we elaborate a novel approach to generate the sub-matrix by exploiting the spatial information of 2D features. Specifically, for the sub-matrix $\mathbf{A}$ which is given the fixed shape $h \times h$, and the element $a_{i,j}$ of $\mathbf{A}$ is corresponding to the height position $(y^{(i)}, y^{(j)})$, it is natural to consider $a_{i,j}$ as the inner product of $y^{(i)} - th$ and $y^{(j)} - th$ row vector of 2D feature map, with the same idea, the element $b_{i,j}$ can be obtained by the inner product of the column vector of 2D feature map. Therefore, as shown in Fig. 2, given two points $\mathbf{i}$ and $\mathbf{j}$ on the 2D feature map with spatial position $(x^{(i)}, y^{(i)})$ and $(x^{(j)}, y^{(j)})$, the elements in sub-matrix $\mathbf{A}$ and $\mathbf{B}$ can be calculated by:

$$a_{ij}^{n} = \sum_{ch=1}^{c} q_{ch}^{y^{(i)}} \cdot k_{ch}^{y^{(j)}}, \qquad b_{ij}^{n} = \sum_{ch=1}^{c} q_{ch}^{x^{(i)}} \cdot k_{ch}^{x^{(j)}} \tag{6}$$

Here $q_{ch}^{y^{(i)}}$ denotes the $y^{(i)}$-th row vector 2D feature map $\mathbf{Q}_s$ while $k_{ch}^{x^{(j)}}$ denotes the $x^{(j)}$-th column in $\mathbf{K}_s$, the subscript ch denotes the channel number and the superscript n denotes the n-th head. To keep the same form with multi-head attention, we use c as the channel dimension in each head and s is the number of heads, therefore the total channel dimension d equals $c \times s$. Comparing with the standard multi-head attention, the only difference is all the value matrices $\mathbf{V}$ in the same head are multiplied by a shared attention matrix $\widehat{\mathbf{AM}}$ which is obtained by sum up all the dot-product results among the channel. Coupling attention mechanism implies a closer form with the human natural eye movement patterns [30], because humans prefer to compare the difference between row by row and column by column in the spatial field rather than the channel direction.

3.4. Efficient Calculation

Based on the hypothesis 1 and Eq. (5), we construct the sub-matrix $\mathbf{A}$ and $\mathbf{B}$ from a novel perspective of attention

mechanism. Thus, the original attention matrix can be decouple into two sub-matrix $\mathbf{A} \in \mathbb{R}^{h\times h}$ and $\mathbf{B} \in \mathbb{R}^{w\times w}$. The main advantage of coupling attention is that it can implement the procedure of attention mechanism without the explicit calculation of attention matrix **AM**, which can significantly mitigate the burden of memory consumption.

In the standard attention mechanism, the alignment score is the production in channel dimension from $\mathbf{Q}, \mathbf{K} \in \mathbb{R}^{h\times w,c}$. According the Eq. (5), then we can replace the original alignment score am_{ij} with the product of $a_{[y^{(i)},y^{(j)}]}$ and $b_{[x^{(i)},x^{(j)}]}$. In detail, $a_{[y^{(i)},y^{(j)}]}$ presents the alignment scores of height dimension with $\mathbf{Q}_a, \mathbf{K}_a \in \mathbb{R}^{h,w\times c}$, and $b_{[x^{(i)},x^{(j)}]}$ presents the alignment scores of width dimension with $\mathbf{Q}_b, \mathbf{K}_b \in \mathbb{R}^{w,h\times c}$. Therefore, the coupling attention mechanism can be regarded as capturing the similarity on the 2D feature map with the spatial information rather than the channel-wise information. Furthermore, coupling attention mechanism can be efficiently calculated by the vec operator trick.

The calculating procedure of coupling attention mechanism can be simplified to significantly reduce the time and space complexity of algorithm, the output O in the Eq. (1) can be obtained without explicitly calculating the $\widehat{\mathbf{AM}}$ by vectorization trick.

Lemma 1 *The Kronecker product can be used to get a convenient representation for some matrix equations:*

$$(\boldsymbol{A} \otimes \boldsymbol{B}) \cdot row(\boldsymbol{X}) = row(\boldsymbol{A} \cdot \boldsymbol{X} \cdot \boldsymbol{B}^\top)$$

Here $row(\cdot)$ denotes the vectorization of the matrix $\boldsymbol{X}$, which stack the rows of a matrix $\boldsymbol{X} \in \mathbb{R}^{m\times n}$ one underneath the other to obtain a single vector $row(\boldsymbol{X}) \in \mathbb{R}^{mn}$.

In the equation 1, the output of original attention mechanism O is the dot product between the softmax of attention matrix and the **V**. With the lemma1, the equation 1 can be written as below:

$$SM(\widehat{AM}) \cdot row(V) \approx row((SM(A))V(SM(B))^\top) \quad (7)$$

In Eq. (7), **V** is reshaped into $\mathbb{R}^{c\times h\times w}$ and is applied the matrix multiplication with **A** and $\mathbf{B}^\top$ with shape $\mathbb{R}^{c\times h\times h}$ and $\mathbb{R}^{c\times w\times w}$ in succession. Moreover, we empirically split out the softmax operation on sub-matrix **A** and **B**, respectively. In the implementation, we maintain the concept of multi-head structure to keep the same form with standard Transformer. And we find that the performance of model is sensitive with the number of heads, more detail will be discussed in the Section 4.

Complexity Analysis To better illustrate the efficiency of our model, we provide some complexity analysis in this section. As described in the Section 3.4, $L = h \times w$. According to the Eq. (7), the computation complexity of the coupled attention block is $\mathcal{O}(h^2 + w^2)$, which is less than the $\mathcal{O}((hw)^2)$ in standard self-attention. In the aspect of the number of training parameters, our model has $4d^2$ training parameters, which is the same as standard self-attention.

3.5. Model Structure

To adopt the Couplformer for CV problem, we are inspired by [13, 14] and elaborate an architecture based on the ViT [10]. In this section, we would like to introduce the structures specifically. The total framework of our model is presented in the Fig. 3. The architecture hyper-parameters of Couplformer are:

- Couplformer-Micro: layer numbers = 6, embedding dimension=128
- Couplformer-Tiny: layer numbers = 8, embedding dimension=256
- Couplformer-Small: layer numbers = 14, embedding dimension=256
- Couplformer-Base: layer numbers = 14, embedding dimension=384

Position Embedding In a standard Transformer, position embedding is an essential way regarding encoding spatial information [10]. Following the mainstream of the Transformer researcher, we also apply the learnable and sine positional embedding to capture the relative distance between patches. However, due to the spatial information capture ability of Couplformer, we infer that the position embedding plays a less important role in Couplformer than standard Transformer, and the experiment results in Section 4 demonstrate this inference.

Encoder with Coupling Attention The block of the encoder consists of two parts: the multi-head attention block and the feed-forward network. In our architecture, we replace the original multi-head attention block with coupling attention which can capture spatial information in visual tasks and reduce the memory consumption. Moreover, the other components, such as Layer Normalization(LN) and Feedforward Network(FFN), are kept the same with the regular encoder.

Classification Token According to the standard approach of the Transformer framework, an extra learnable classification token is added to the sequence. However, in our model, we want to keep the shape of input feature maps in the Transformer encoder. Therefore, we apply the sequence pooling structure [14] to avoid the usage of extra classification token by which the tensor decomposition is destroyed. In this structure, the model could distribute more weight to

the patch, which contains more information relevant to the classifier.

4. EXPERIMENTS

In this section, we investigate the model capabilities of couplformer, Convolutional related Neural Networks(ConvNets), and other Vision Transformers on different scales of image classification tasks. Then, we conduct several ablation studies to present the unique performance of our method.

4.1. Training Setup

In order to explore the scalability of Couplformer, we conduct our experiment on CIFAR-10 [20], CIFAR-100 [20], ImageNet-1K [9]. The benchmark dataset CIFAR-10 has 10 classes of images, including 50000 images for training and 10000 images for testing. CIFAR-100 has 100 classes of images, and each class includes 500 images for training and 100 images for testing. ImageNet-1K dataset has 1000 classes images, including 1,281,167 images for training and 50,000 images for validation. In Section 4.4, we conduct object detection and instance segmentation tasks on MS COCO 2017 [21] dataset to evaluate the generalization of our method. The experiments' implementation details are listed in the corresponding section and the supplemental material.

4.2. Image Classification

Evaluation in Small Datasets As described in Table 1, we conduct the image classification task in CIFAR-10, and CIFAR100 datasets. We compare Couplformer with the standard Visual Transformer [10] and several popular efficient Transformers [41, 19, 4] on the same backbone [14] We also list two ConvNets baseline: ResNet [15] and MobileNet [29] for comparison.

From the results in Table 1, most transformer-based models present a similar performance to the convolution-based models. As argued in [14], standard ViT architecture is not adapted to the small-size datasets. However, efficient transformer models and hybrid transformer-based models present adaptiveness in small-size datasets. In terms of calculation volume, convolution-based models have fast and smaller computation complexity. Nevertheless, with the growth of the size of the dataset, the convolution-based models reveal a lower accuracy than efficient transformers. For example, the most efficient transformer models achieve above 70% accuracy in the CIFAR-100 dataset. According to the result of Couplformer-T, it achieves the promised accuracy under the same parameter volume and MACs. Although the Couplformer-S does not obtain the best accuracy under the evaluation of small-size datasets, it has the most economical memory usage while maintaining slightly lower accuracy than other efficient transformer models. Additionally, Couplformer still outperforms the ConvNets without a significant increase in MACs. These experiments prove that our model could keep remarkable outcomes on small-scale datasets.

Model	CIFAR -10	CIFAR -100	Params	MACs
Convolutional Networks				
ResNet34 [15]	89.45%	64.67%	21.80M	0.09G
ResNet50 [15]	89.30%	61.25%	25.56M	0.09G
MobileNet [29]	90.55%	67.12%	8.72M	**0.03G**
Vision Transformer				
ViT-Base [10]	76.42%	46.61%	85.63M	0.43G
ViT-Lite [14]	91.38%	69.74%	3.72M	0.24G
Swin-T [22]	89.67%	65.79%	29.0M	0.45G
Swin-S [22]	89.74%	62.75%	50.0M	0.87G
CVT-7/4 [14]	92.43%	73.01%	3.72M	0.24G
CCT-4/3 [14]	91.45%	70.46%	0.48M	0.05G
Efficient Transformer				
Linformer [41]	92.45%	70.87%	3.96M	0.28G
Performer [4]	91.58%	73.11%	3.85M	0.28G
Reformer [19]	90.58%	73.02%	3.39M	0.25G
Couplformer-M	90.81%	69.19%	**0.48M**	0.10G
Couplformer-T	**93.44%**	**74.53%**	3.85M	0.28G
Couplformer-S	92.15%	67.22%	20.94M	1.38G

Table 1: **CIFAR-10 and CIFAR-100 results**. MACs denotes the Multiply-Accumulate operations. The image resolution is 32×32 in the training process of CIFAR-10 and CIFAR-100 datasets. All models are trained from scratch using AdamW optimizer for 200 epochs with cosine learning rate decay, and batch size is 128. We train our models with a single GeForce RTX 2080Ti (11GB). Data augmentation includes random crop and random horizontal flip.

Evaluation in ImageNet Dataset Table 2 summarizes the result of the evaluation on ImageNet-1K dataset. We compare our approach with several popular methods, including convolution-based (ResNet [15], RegNetY [27]) and Transformer-based (Standard ViT [10], DeiT [36], Swin [22], Mobile-Former [2]) methods. All variants are trained from scratch. According to the ConvNets, i.e. RegNet [15, 27], our model still achieves better performance in the Top-1 accuracy. The RegNet inherits the advantage of the ConvNets, which has the more lightweight parameter volume. However, as a model based on Neural Architecture Search, RegNet has an expansive building cost and weak data generalization ability. The results of Top-1 accuracy ($81.50\%/81.75\%$ vs. 82.36%) demonstrate that our special attention design performs efficient representation capability.

In terms of Transformer-based models, DeiT [36] uti-

Model	Image size	Params	MACs	ImageNet-1K top-1 acc.
ResNet50 [15]	224^2	25.6M	8.2G	76.13%
ResNet101 [15]	224^2	44.5M	15.8G	77.37%
RegNetY-3.2G [27]	224^2	21M	6.4G	78.95%
RegNetY-8G [27]	224^2	39M	16.0G	80.03%
ViT-B/16 [10]	384^2	86M	110.8G	77.9%
ViT-L/16 [10]	384^2	307M	381.4G	76.5%
DeiT-S [36]	224^2	22M	9.2G	79.98%
DeiT-B [36]	224^2	86M	35.0G	81.80%
Swin-T [22]	224^2	29M	9.0G	81.3%
Swin-S [22]	224^2	50M	17.4G	83.0%
MobileFormer-T [2]	224^2	11.4M	2.35G	77.9%
MobileFormer-S [2]	224^2	14.0M	4.06G	79.3%
Couplformer-T	224^2	28M	6.4G	80.48%
Couplformer-S	224^2	49M	20.4G	82.36%

Table 2: **ImageNet-1K results**. The image resolution is 224×224 in the ImageNet-1k dataset's training setup. All models are trained using AdamW optimizer for 1024 epochs, and the batch size is 1024. We train our models with 8 GeForce RTX 3090Ti (24GB). We leverage Auto-Augment [6], Rand-Augment [7], and random erasing [47] as data augmentation.

lizes the knowledge distillation method to improve the network's performance. Due to the teacher-student network for token-based distillation, memory usage would notably increase when the model has a larger number of layers. MobileFormer [2] presents the minimum parameter and computation volume. However, it pays the price of its lightweight by a lower accuracy. SwinTrasnformer [22], as one of the most powerful transformer-based models, reaches the 83.0% Top-1 accuracy. Nevertheless, our method also reaches a similar performance and the difference between our model and SwinTransformer is less than 0.82%. These results also indicate the effectiveness of the novel attention mechanism in our model.

4.3. Ablation Study

Number of Heads To reach the best performance, most transformer-based models have the requirement of heads number. In most cases, a large number of heads could achieve better performance [37]. For example, most models' head number is large than 32. Our ablation study, which leverages the concatenate aggregation choice, also supports the abovementioned statement. As shown in Fig. 4, the accuracy has a sharp decrease once the head number is less than 32. However, under the pooling choice, our model reveals the robustness of the change of head number. From 1 to 256 heads, the fluctuation of the accuracy is under 1%. Moreover, we found that employing a large head number could perform better under both aggregation choices. According to the influence of the layer number, we also found that the layer number could also have a slight effect on performance. We list the detail of the layer number's study in the supplemental.

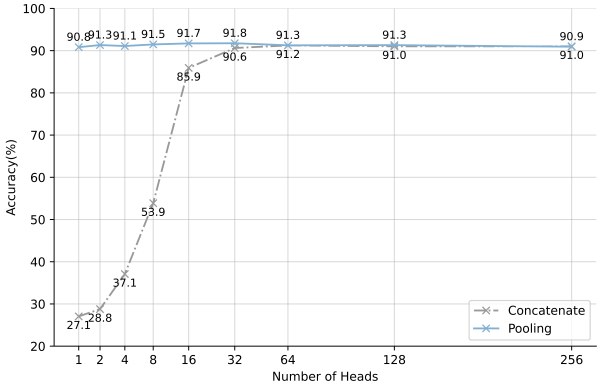


Figure 4: The comparison of Top-1 validation accuracy under different head numbers and aggregation choices. The ablation study is conducted under the Couplformer-T on CIFAR-10 dataset.

Evaluation of Position Embedding We conduct the ablation study about the purpose of position embedding. In the Visual Transformer structure, adding learned positional embeddings to inputs has become mainstream. The position embedding has been proved that it could bring considerable accuracy improvement [10]. However, in our model, the utilization of spatial information to generate the alignment scores in coupling the attention mechanism makes the positional embedding less important.

Table 3 summarizes of the result from the investigation on ViT-12/16 [10], CCT [14], and Couplformer. Among the evaluated three position embedding solutions, sinusoidal takes the best result in ViT and CCT methods. Then, the result of the learnable is slightly lower than sinusoidal. The result without position embedding gets the worst one. As

Model	Pos Emb	CIFAR-10	CIFAR-100
ViT-12/16 [10]	None	73.31%	44.27%
	Sinusoidal	76.42%	46.61%
	Learnable	74.35%	45.89%
CCT-4/3 [14]	None	90.59%	66.25%
	Sinusoidal	91.45%	70.46%
	Learnable	91.42%	70.35%
Couplformer-T	None	93.36%	73.29%
	Sinusoidal	93.44%	74.53%
	Learnable	93.42%	73.79%

Table 3: The comparison of Top-1 validation accuracy under different modes of position embedding.

Object Detection on MS COCO 2017						
Backbone	AP^b	AP^b_{50}	AP^b_{75}	AP^b_s	AP^b_m	AP^b_l
ResNet-50	42.3	60.5	46.0	23.7	45.7	56.4
ResNet-101	43.3	61.3	47.0	24.4	46.9	58.0
HRNetV2p-W18 [40]	41.9	59.6	45.7	23.8	44.9	55.0
HRNetV2p-W32 [40]	44.5	62.3	48.6	26.1	47.9	58.5
X-101-32x4d	44.7	63.0	48.9	25.9	48.7	58.9
X-101-64x4d	45.7	64.1	50.0	26.2	49.6	60.0
Couplformer-T	45.7	64.5	49.0	29.3	49.0	60.3
Instance Segmentation on MS COCO 2017						
Backbone	AP^m	AP^m_{50}	AP^m_{75}	AP^m_s	AP^m_m	AP^m_l
ResNet-50	36.6	57.6	39.5	19.0	39.4	50.7
ResNet-101	37.6	58.5	40.6	19.7	40.8	52.4
HRNetV2p-W18 [40]	36.4	56.8	39.3	17.0	38.6	52.9
HRNetV2p-W32 [40]	38.5	59.6	41.9	18.9	41.1	56.1
X-101-32x4d	38.6	60.2	41.7	20.9	42.1	52.7
X-101-64x4d	39.4	61.3	42.9	20.8	42.7	54.1
Couplformer-T	39.8	62.1	42.9	23.7	43.0	54.2

Table 4: **Object detection and instance segmentation result** The overall results are obtained via the object detection frameworks: Cascade Mask R-CNN [16]. X-101 denotes ResNeXt-101.AP^b and AP^m denote box mAP and mask mAP respectively.

we see, the regular Transformer in ViT-12/16 and CCT are both benefited from position embedding to obtain a gap of improvement than the baseline without embedding position. In contrast, Couplformer appears unaffected with or without position embedding. This result proves that our model typically does not rely on position embedding. Besides, it also demonstrates that spatial information is implicitly employed, as we declare in Section 3.

4.4. Object Detection

Setting We conduct the object detection and instance segmentation experiment on MS COCO 2017 dataset [21], which contains 118K images for training, 5K for validation, and 20K testing images without providing annotations. As shown in the Table 4, we compare our approach with other backbones: ResNet, ResNeXt [43], and HRNet [40]. For all trained models, we use standard horizontal flipping as data augmentation and resize the input image so that the shorter edge is 800 pixels. The rest of implementation details are listed in the supplementary material.

Evaluation in MS COCO 2017 dataset In terms of object detection, our method and X-101-64-x4d achieve the best AP result. Besides, our method has $+0.4$ and $+1.0$ better in AP^b_{50} and AP^b_{70}, respectively. According to the instance segmentation, Couplformer-T outperforms with rest of the approaches. Moreover, our method has a noticeable improvement in AP_s results, which indicates that the proposed unique attention mechanism enhances the detection capability of small objects. Whereas, for medium and large objects, Couplformer-T does not reveals as appreciable improvement as the small objects.

5. Limitation and Discussions

Although Couplformer presents lightweight and accuracy in several experiments, it can not outperform other methods in the large dataset, i.e., ImageNet-1k. Our model could only match but could not surpass other methods because of the restriction of our hardware. Therefore, in future research, we will improve our model's overall architecture to reach a better result. We believe our coupled attention mechanism still has the potential to be released.

6. CONCLUSIONS

In this paper, we presented a novel memory economy attention mechanism, named Couplformer, which employs spatial information to couple the attention map to replace the traditional self-attention module. Through this novel approach, the shortage of dramatic memory consumption of Visual Transformer is efficiently mitigated. We demonstrate that our model achieves sufficient accuracy requirements with the minimal occupation of GPU. Moreover, we apply spatial information to generate the alignment scores rather than channel-wise as the standard Visual Transformer model, and the experiments confirm the effectiveness of this coupling attention mechanism. Our model makes the Transformer model more flexible for different scales of defined settings. In the days of data explosion, our model helps more researchers who are suffering from limited hardware resources to train on the various sizes of datasets lightheartedly. We believe that our model will bring researchers a fresh perspective of deep learning architectures.

7. Acknowledgements

This work was partially supported by Fujian Science & Technology Innovation Laboratory for Optoelectronic Information of China (No.2021ZZ120) and Fujian Science and Technology Plan(No. 2021T3003)

References

[1] Iz Beltagy, Matthew E Peters, and Arman Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020.

[2] Yinpeng Chen, Xiyang Dai, Dongdong Chen, Mengchen Liu, Xiaoyi Dong, Lu Yuan, and Zicheng Liu. Mobileformer: Bridging mobilenet and transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5270–5279, 2022.

[3] Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating long sequences with sparse transformers. *ArXiv*, abs/1904.10509, 2019.

[4] Krzysztof Marcin Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamás Sarlós, Peter Hawkins, Jared Quincy Davis, Afroz Mohiuddin, Lukasz Kaiser, David Benjamin Belanger, Lucy J. Colwell, and Adrian Weller. Rethinking attention with performers. In *9th International Conference on Learning Representations, Virtual Event, Austria, May 3-7*, 2021.

[5] Xiangxiang Chu, Bo Zhang, Zhi Tian, Xiaolin Wei, and Huaxia Xia. Do we really need explicit position encodings for vision transformers? *arXiv e-prints*, pages arXiv–2102, 2021.

[6] Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation policies from data. *arXiv preprint arXiv:1805.09501*, 2018.

[7] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. 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, 2020.

[8] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime G. Carbonell, Quoc Viet Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, Florence, Italy*, 2019.

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *In Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 248–255, 2009.

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning Representations, Virtual Event, Austria, May 3-7*, 2021.

[11] Vijay Prakash Dwivedi and Xavier Bresson. A generalization of transformer networks to graphs. *arXiv preprint arXiv:2012.09699*, 2020.

[12] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[13] Ben Graham, Alaaeldin El-Nouby, Hugo Touvron, Pierre Stock, Armand Joulin, Hervé Jégou, and Matthijs Douze. Levit: a vision transformer in convnet's clothing for faster inference. *arXiv preprint arXiv:2104.01136*, 2021.

[14] Ali Hassani, Steven Walton, Nikhil Shah, Abulikemu Abuduweili, Jiachen Li, and Humphrey Shi. Escaping the big data paradigm with compact transformers. *arXiv preprint arXiv:2104.05704*, 2021.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[17] Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *International Conference on Machine Learning(ICML)*, pages 5156–5165, 2020.

[18] Salman Khan, Muzammal Naseer, Munawar Hayat, Syed Waqas Zamir, Fahad Shahbaz Khan, and Mubarak Shah. Transformers in vision: A survey. *arXiv preprint arXiv:2101.01169*, 2021.

[19] Nikita Kitaev, Lukasz Kaiser, and Anselm Levskaya. Reformer: The efficient transformer. In *8th International Conference on Learning Representations, Addis Ababa, Ethiopia, April 26-30, 2020*, 2020.

[20] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

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

[22] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, October 2021.

[23] Jiasen Lu, Caiming Xiong, Devi Parikh, and Richard Socher. Knowing when to look: Adaptive attention via a visual sentinel for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 375–383, 2017.

[24] Zhaoyang Niu, Guoqiang Zhong, and Hui Yu. A review on the attention mechanism of deep learning. *Neurocomputing*, 452:48–62, 2021.

[25] Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Noam Shazeer, Alexander Ku, and Dustin Tran. Image transformer. In *International Conference on Machine Learning*, pages 4055–4064, 2018.

[26] Jiezhong Qiu, Hao Ma, Omer Levy, Wen-tau Yih, Sinong Wang, and Jie Tang. Blockwise self-attention for long document understanding. In *Findings of the Association for Computational Linguistics: EMNLP Online Event, 16-20 November*, 2020.

[27] Ilija Radosavovic, Raj Prateek Kosaraju, Ross Girshick, Kaiming He, and Piotr Dollár. Designing network design spaces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

[28] Jack W. Rae, Anna Potapenko, Siddhant M. Jayakumar, Chloe Hillier, and Timothy P. Lillicrap. Compressive transformers for long-range sequence modelling. In *8th International Conference on Learning Representations, Addis Ababa, Ethiopia, April 26-30,*, 2020.

[29] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018.

[30] Mads Soegaard. Visual hierarchy: Organizing content to follow natural eye movement patterns. *Interaction Design Foundation [online]. Aarhus: Interaction Design Foundation*, 2, 2020.

[31] Kaikai Song, Ting Yao, Qiang Ling, and Tao Mei. Boosting image sentiment analysis with visual attention. *Neurocomputing*, 312:218–228, 2018.

[32] Sijie Song, Cuiling Lan, Junliang Xing, Wenjun Zeng, and Jiaying Liu. An end-to-end spatio-temporal attention model for human action recognition from skeleton data. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31, 2017.

[33] Yi Tay, Mostafa Dehghani, Dara Bahri, and Donald Metzler. Efficient transformers: A survey. *arXiv preprint arXiv:2009.06732*, 2020.

[34] Yan Tian, Wei Hu, Hangsen Jiang, and Jiachen Wu. Densely connected attentional pyramid residual network for human pose estimation. *Neurocomputing*, 347:13–23, 2019.

[35] Ilya Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, et al. Mlp-mixer: An all-mlp architecture for vision. *arXiv preprint arXiv:2105.01601*, 2021.

[36] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*. PMLR, 2021.

[37] Hugo Touvron, Matthieu Cord, Alexandre Sablayrolles, Gabriel Synnaeve, and Hervé Jégou. Going deeper with image transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 32–42, 2021.

[38] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[39] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

[40] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, et al. Deep high-resolution representation learning for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 43(10):3349–3364, 2020.

[41] Sinong Wang, Belinda Z. Li, Madian Khabsa, Han Fang, and Hao Ma. Linformer: Self-attention with linear complexity. *ArXiv*, abs/2006.04768, 2020.

[42] Haiping Wu, Bin Xiao, Noel Codella, Mengchen Liu, Xiyang Dai, Lu Yuan, and Lei Zhang. Cvt: Introducing convolutions to vision transformers. *arXiv preprint arXiv:2103.15808*, 2021.

[43] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500, 2017.

[44] Yunyang Xiong, Zhanpeng Zeng, Rudrasis Chakraborty, Mingxing Tan, Glenn Fung, Yin Li, and Vikas Singh. Nyströmformer: A nyström-based algorithm for approximating self-attention. In *Thirty-Fifth AAAI Conference on Artificial Intelligence*, pages 14138–14148, 2021.

[45] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057. PMLR, 2015.

[46] Kun Yuan, Shaopeng Guo, Ziwei Liu, Aojun Zhou, Fengwei Yu, and Wei Wu. Incorporating convolution designs into visual transformers. *arXiv preprint arXiv:2103.11816*, 2021.

[47] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 13001–13008, 2020.