# Supplementary Material: Scalable Vision Transformers with Hierarchical Pooling

Zizheng Pan   Bohan Zhuang*   Jing Liu   Haoyu He   Jianfei Cai
Dept of Data Science and AI, Monash University

We organize our supplementary material as follows.

- In Section 1, we elaborate on the components of a Transformer block, including the multi-head self-attention layer (MSA) and the position-wise multi-layer perceptron (MLP).
- In Section 2, we provide details for the FLOPs calculation of a Transformer block.

# 1. Transformer Block

## 1.1. Multi-head Self-Attention

Let $\mathbf{X} \in \mathbb{R}^{N \times D}$ be the input sentence, where $N$ is the sequence length and $D$ the embedding dimension. First, a self-attention layer computes query, key and value matrices from $\mathbf{X}$ using linear transformations

$$[\mathbf{Q}, \mathbf{K}, \mathbf{V}] = \mathbf{X}\mathbf{W}_{qkv}, \tag{A}$$

where $\mathbf{W}_{qkv} \in \mathbb{R}^{D \times 3D_h}$ is a learnable parameter and $D_h$ is the dimension of each self-attention head. Next, the attention map $\mathbf{A}$ can be calculated by scaled inner product from $\mathbf{Q}$ and $\mathbf{K}$ and normalized by a softmax function

$$\mathbf{A} = \mathrm{Softmax}(\mathbf{Q}\mathbf{K}^\top / \sqrt{D_h}), \tag{B}$$

where $\mathbf{A} \in \mathbb{R}^{N \times N}$ and $A_{ij}$ represents for the attention score between the $\mathbf{Q}_i$ and $\mathbf{K}_j$. Then, the self-attention operation is applied on the value vectors to produce an output matrix

$$\mathbf{O} = \mathbf{A}\mathbf{V}, \tag{C}$$

where $\mathbf{O} \in \mathbb{R}^{N \times D_h}$. For a multi-head self-attention layer with $D/D_h$ heads, the outputs can be calculated by a linear projection for the concatenated self-attention outputs

$$\mathbf{X}^{'} = [\mathbf{O}_1; \mathbf{O}_2; ...; \mathbf{O}_{D/D_h}]\mathbf{W}_{proj}, \tag{D}$$

where $\mathbf{W}_{proj} \in \mathbb{R}^{D \times D}$ is a learnable parameter and $[\cdot]$ denotes the concatenation operation.

## 1.2. Position-wise Multi-Layer Perceptron

Let $\mathbf{X}^{'}$ be the output from the MSA layer. An MLP layer which contains two fully-connected layers with a GELU non-linearity can be represented by

$$\mathbf{X} = \mathrm{GELU}(\mathbf{X}^{'}\mathbf{W}_{fc1})\mathbf{W}_{fc2}, \tag{E}$$

where $\mathbf{W}_{fc1} \in \mathbb{R}^{D \times 4D}$ and $\mathbf{W}_{fc2} \in \mathbb{R}^{4D \times D}$ are learnable parameters.

*Corresponding author. Email: bohan.zhuang@monash.edu

## 2. FLOPs of a Transformer Block

We denote $\phi(n, d)$ as a function of FLOPs with respect to the sequence length $n$ and the embedding dimension $d$. For an MSA layer, The FLOPs mainly comes from four parts: (1) The projection of **Q**,**K**,**V** matrices $\phi_{qkv}(n, d) = 3nd^2$. (2) The calculation of the attention map $\phi_A(n, d) = n^2 d$. (3) The self-attention operation $\phi_O(n, d) = n^2 d$. (4) And finally, a linear projection for the concatenated self-attention outputs $\phi_{proj}(n, d) = nd^2$. Therefore, the overall FLOPs for an MSA layer is

$$\begin{aligned} \phi_{MSA}(n, d) &= \phi_{qkv}(n, d) + \phi_A(n, d) + \phi_O(n, d) + \phi_{proj}(n, d) \\ &= 3nd^2 + n^2 d + n^2 d + nd^2 \\ &= 4nd^2 + 2n^2 d. \end{aligned} \tag{F}$$

For an MLP layer, the FLOPs mainly comes from two fully-connected (FC) layers. The first FC layer $fc1$ is used to project each token from $\mathbb{R}^d$ to $\mathbb{R}^{4d}$. The next FC layer $fc2$ projects each token back to $\mathbb{R}^d$. Therefore, the FLOPs for an MLP layer is

$$\phi_{MLP}(n, d) = \phi_{fc1}(n, d) + \phi_{fc2}(n, d) = 4nd^2 + 4nd^2 = 8nd^2. \tag{G}$$

By combining Eq. (F) and Eq. (G), we can get the total FLOPs of one Transformer block

$$\phi_{BLK}(n, d) = \phi_{MSA}(n, d) + \phi_{MLP}(n, d) = 12nd^2 + 2n^2 d. \tag{H}$$