Note: MixFormer: Mixing Features across Windows and Dimensions[1]

sgc

April 23, 2022

1 Abstract

Local-window self-attention suffers from non-overlapped windows and shares weights on channel dimension. So authors proposed MixFormer combining local-window self-attention and depth-wise convolution in a parallel design. And they also designed a bi-directional interaction across the two branches.

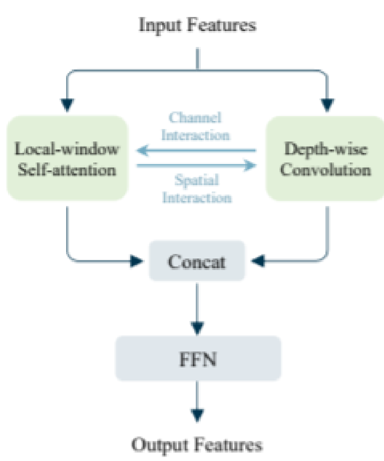

Figure 1. **The Mixing Block.** We combine local-window self-attention with depth-wise convolution in a parallel design. The captured relations within and across windows in parallel branches are concatenated and sent to the Feed-Forward Network (FFN) for output features. In the figure, the blue arrows marked with *Channel Interaction* and *Spatial Interaction* are the proposed bi-directional interactions, which provide complementary clues for better representation learning in both branches. Other details in the block, such as module design, normalization layers, and shortcuts, are omitted for a neat presentation.

2 Method

2.1 The Mixing Block

2 main design: (1) adopt a parallel design to combine local-window self-attention and depth-wise convolution, (2) introduce bi-directional interactions across branches.

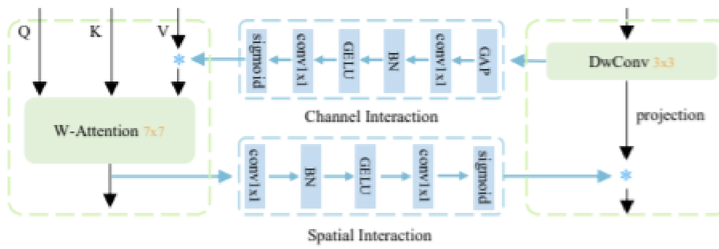

Figure 2. **Detailed design of the Bi-directional Interactions.** The channel/spatial interaction provides channel/spatial context extracted by depth-wise convolution/local-window self-attention to the other path.

The Parallel Design The parallel design benefits two-folds: First, combining local-window self-attention with depth-wise convolution across branches models connections across windows, addressing the limited receptive fields issue. Second, parallel design models intra-window and cross-window relations simultaneously, providing opportunities for feature interweaving across branches and achieving better feature representation learning.

Bi-directional Interactions This paper proposed bi-directional interaction to enhance modeling ability in channel and spatial dimension for local-window self-attention and dwconv respectively.

The Mixing Block

Overall Architecture The projection layer increases the features channels to 1280 with a linear layer followed by an activation layer, aiming to preserve more details in the channel before the classification head.

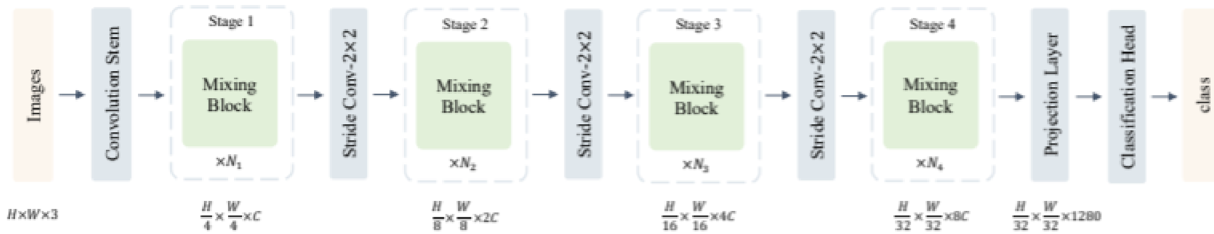

Figure 3. **Overall Architecture of MixFormer.** There are four parts in MixFormer: Convolution Stem, Stages, Projection Layer, and Classification Head. In Convolution Stem, we apply three successive convolutions to increase the channel from 3 to C. In Stages, we stack our Mixing Block in each stage and use stride convolution ($stride = 2$) to downsample the feature map. For Projection Layer, we use a linear layer with activation to increase the channels to 1280. The Classification Head is for the classification task.

References

[1] Q. Chen, Q. Wu, J. Wang, Q. Hu, T. Hu, E. Ding, J. Cheng, and J. Wang, "Mixformer: Mixing features across windows and dimensions," *arXiv preprint arXiv:2204.02557*, 2022. (document)