

# 频率 + Transformer介绍 --视觉方向

主要是介绍频率层 + transformer的应用，本文中主要考虑的是视觉方向的研究，并在不断更新中。

傅里叶域在提取基于频率的图像信息分析中起着重要作用，在频率域中，图像的一些特征（如边缘、纹理等）可能更加明显或者更易于处理。

- **Fnet**的开创性研究支持了这一点，把transformer架构里attention层全部换成傅里叶层，即：将嵌入维度\*输入数量的2-D矩阵进行二维傅里叶分解，取实部得到y。最后的结果证明，一个无attention的transformer结构，对于图像处理的结果依然很好，但是复杂度大大降低。发在NAACL 2021上

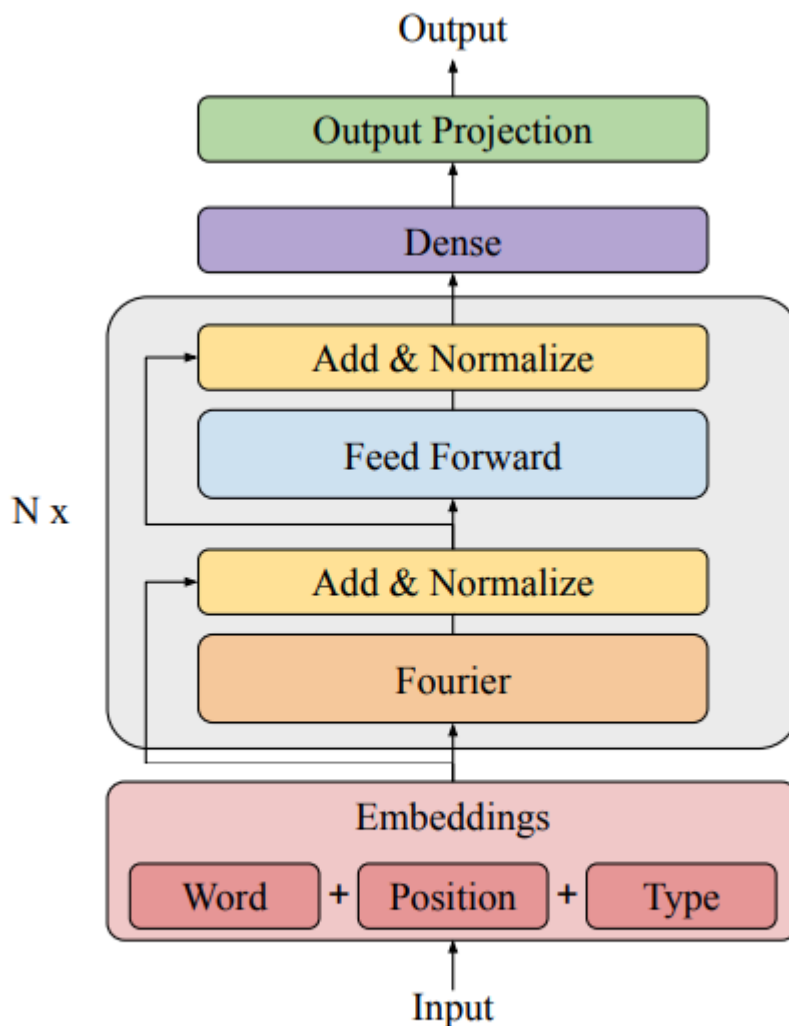


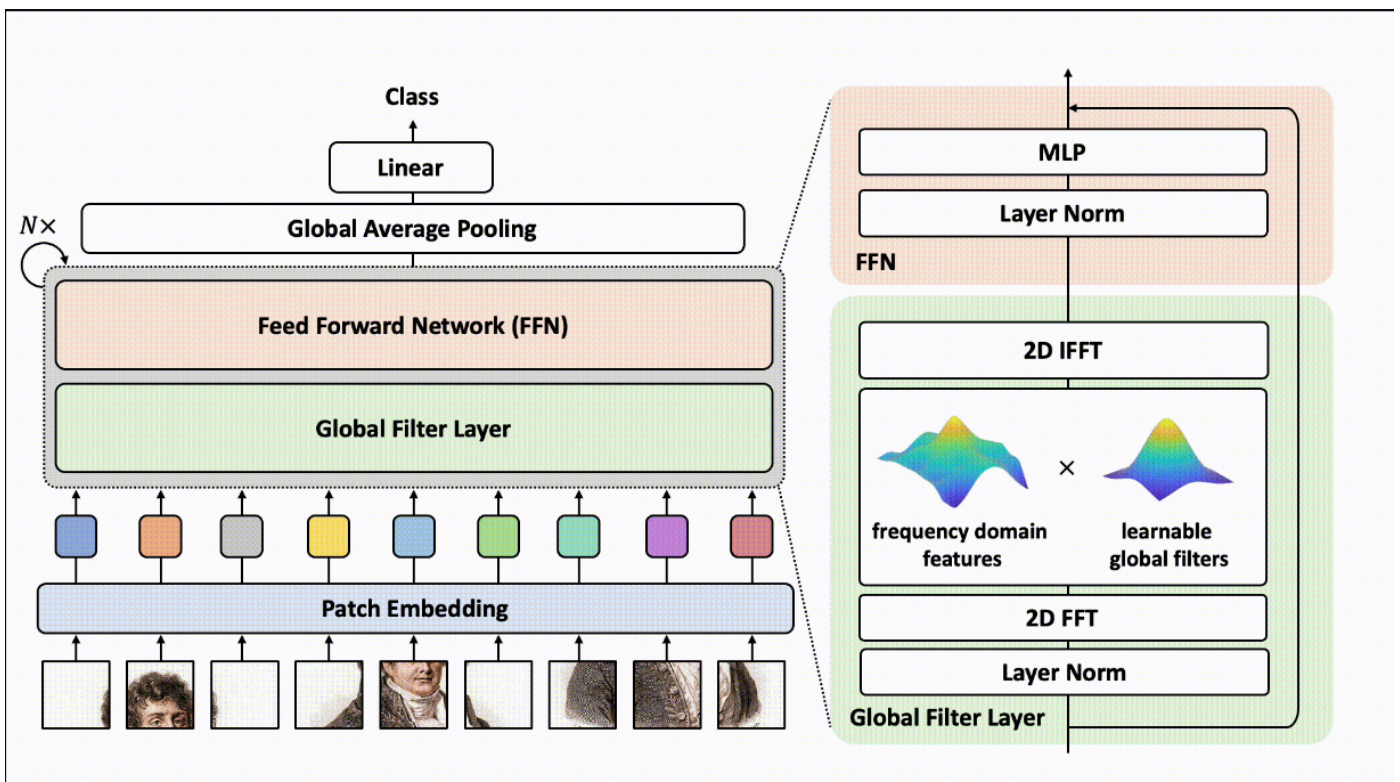
Figure 1: FNet architecture with  $N$  encoder blocks.

- **GFNet**证明傅里叶变换层可以取代多头注意层。实际上就是在上一个Fnet的基础上，把各频率那里加了一个逐元素乘法的权重，这个权重是学的。发在了NeurIPS 2021上和T-PAMI上

```

1  import torch
2  import torch.nn as nn
3  import torch.fft
4  class GlobalFilter(nn.Module):
5      def __init__(self, dim, h=14, w=8):
6          super().__init__()
7          self.complex_weight = nn.Parameter(torch.randn(h, w, dim, 2, dtype=torch.float32))
8      def forward(self, x):
9          B, H, W, C = x.shape
10         x = torch.fft.rfft2(x, dim=(1, 2), norm='ortho')
11         weight = torch.view_as_complex(self.complex_weight)
12         x = x * weight
13         x = torch.fft.irfft2(x, s=(H, W), dim=(1, 2), norm='ortho')
14         return x

```



- **SpectFormer**的贡献在上周汇报过了。实际上就是它发现，需要将频谱层和attention层合一起，先用频谱层再用self-attention层堆叠

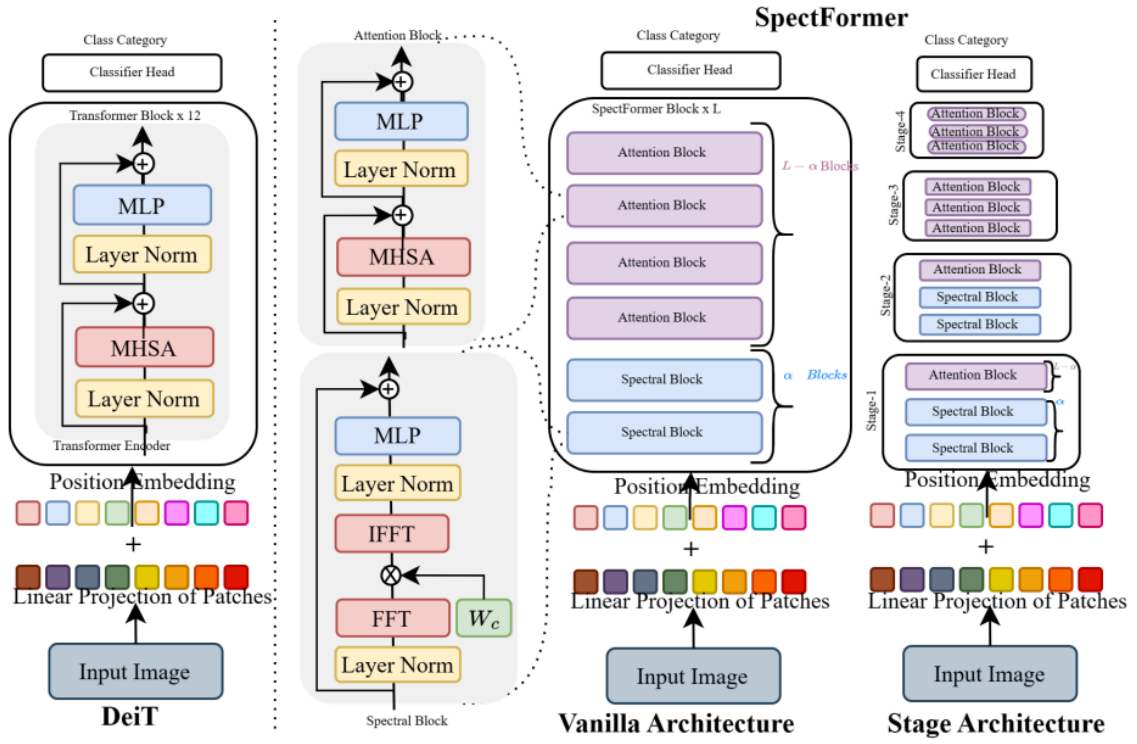


Figure 2. This figure shows Architectural details of SpectFormer. The first part shows the DeiT[53] architecture. The second part shows the vanilla and Stage architecture of the SpectFormer Model. This also shows the layer structure of Spectral and Attention Blocks.