

# $StyTr^2$ : Image Style Transfer with Transformers

## 1 摘要

图像风格迁移的目标是在保持原始内容的条件下，在参考风格的指导下渲染出艺术化的图像。由于卷积神经网络 (CNN) 的局部性，提取和保持输入图像的全局信息很困难。因此，传统的神经网络风格迁移方法在内容表示上会存在有偏的问题。为了解决这个关键问题，我们将输入图像的远程依赖性考虑到图像风格转换中，提出了一种称为  $StyTr^2$  的基于transformer的方法。和用于其他视觉任务的视觉transformer相比， $StyTr^2$  包含两个不同的transformer编码器，分别为内容和样式生成特定领域的序列。在编码器之后，采用多层transformer解码器根据样式序列对内容序列进行样式化。我们还分析了现有位置编码方法的不足，并提出了内容感知位置编码 (CAPE)，它是尺度不变的，更适合图像风格迁移任务。定性和定量实验证明了所提出的  $StyTr^2$  与最先进的基于 CNN 和基于流的方法相比的有效性。

## 2 贡献

- 1) 提出了一个称为  $StyTr^2$  的基于transformer的风格迁移框架，用于生成具有保存完好的输入内容图像结构和细节的风格化结果；
- 2) 提出了一种内容感知的位置编码方案，具有尺度不变性，适用于图像风格转换任务；
- 3) 综合实验表明， $StyTr^2$  优于基线方法，并以理想的内容结构和风格模式取得了出色的结果。

## 3 代码主体部分标注

论文的算法框架如图 1所示：

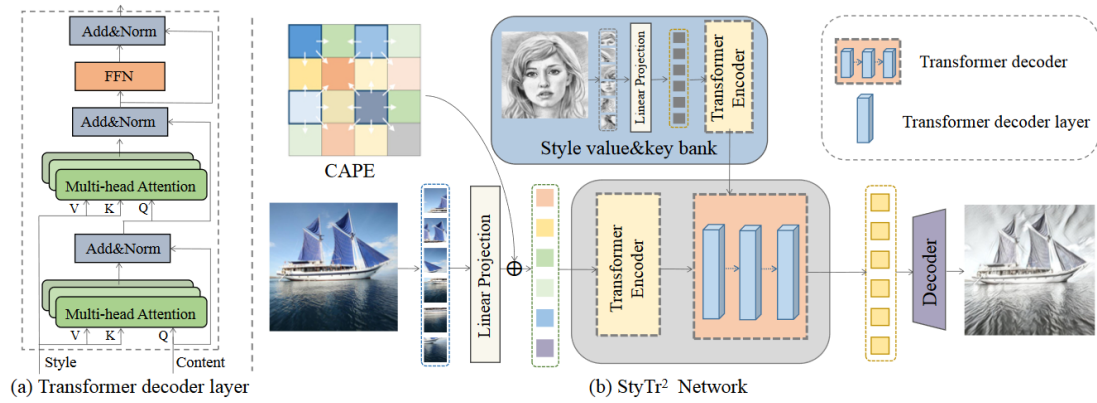


图 1: Frame of CycleGAN

如上图，先将内容和风格图像分割成补丁，并使用线性投影来获得补丁序列。然后，添加了 CAPE 的内容序列被送入内容transformer编码器，而风格序列被送入风格transformer编码器。在两个转换器编码器之后，采用多层transformer解码器根据风格序列对内容序列进行风格化。最后，我们使用渐进式上采样解码器来获得最终输出。代码主体部分及标注如图 2:

```
def forward(self, style, mask_, content, pos_embed_c, pos_embed_s):

    # 内容感知位置嵌入
    content_pool = self.averagepooling(content)
    pos_c = self.new_ps(content_pool)
    pos_embed_c = F.interpolate(pos_c, mode='bilinear', size=style.shape[-2:])

    # flatten NxHxW to HWxC
    style = style.flatten(2).permute(2, 0, 1)
    if pos_embed_s is not None:
        pos_embed_s = pos_embed_s.flatten(2).permute(2, 0, 1)

    content = content.flatten(2).permute(2, 0, 1)
    if pos_embed_c is not None:
        pos_embed_c = pos_embed_c.flatten(2).permute(2, 0, 1)

    # 风格编码器对风格编码
    style = self.encoder_s(style, src_key_padding_mask=mask, pos=pos_embed_s)

    # 内容编码器对内容编码
    content = self.encoder_c(content, src_key_padding_mask=mask, pos=pos_embed_c)

    # 多层transformer解码器根据风格序列对内容序列风格化，进行解码
    hs = self.decoder(content, style, memory_key_padding_mask=mask,
                      pos=pos_embed_s, query_pos=pos_embed_c)[0]

    ### HWxC to NxHxW to
    N, B, C = hs.shape
    H = int(np.sqrt(N))
    hs = hs.permute(1, 2, 0)
    hs = hs.view(B, C, -1, H)
```

图 2: 论文代码主体部分标注

## 4 风格迁移测试示例

采用网上爬取的内容图片和不同的风格图片，对论文算法进行测试，结果如图 3所示。

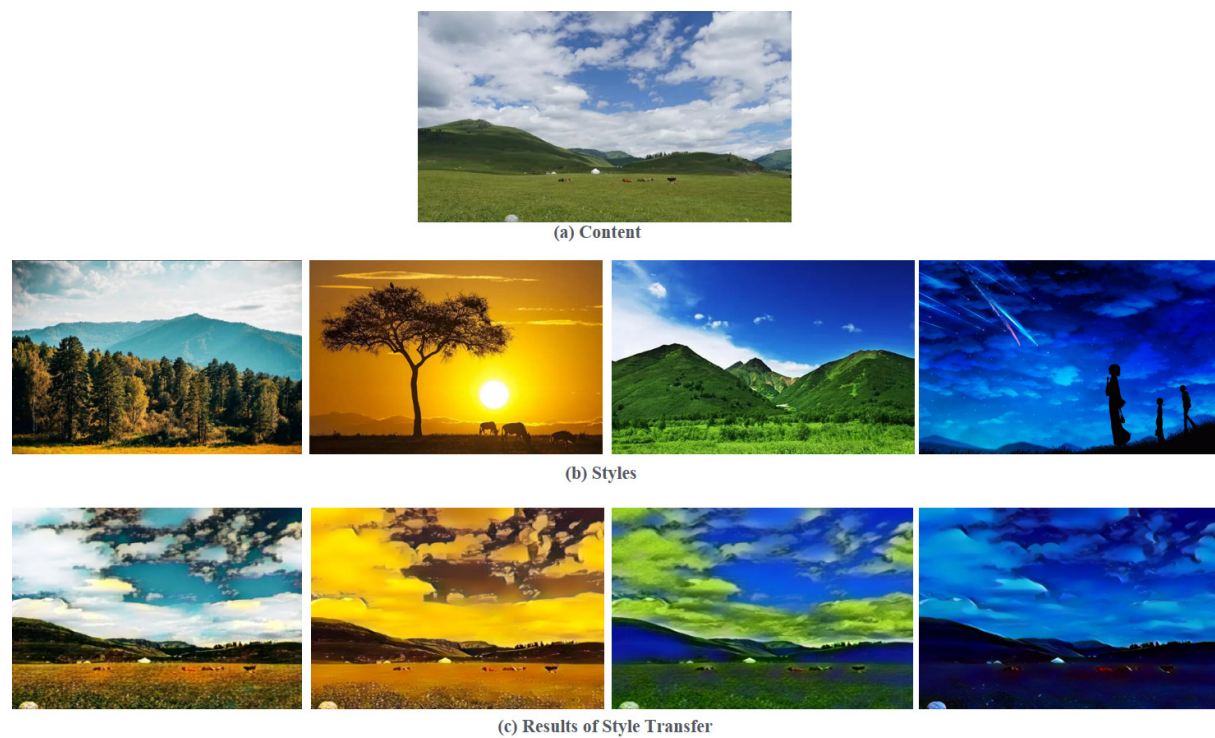


图 3:  $StyTr^2$  风格迁移测试结果示例

Note: 代码运行截图见附件