Face Recognition method survey:2

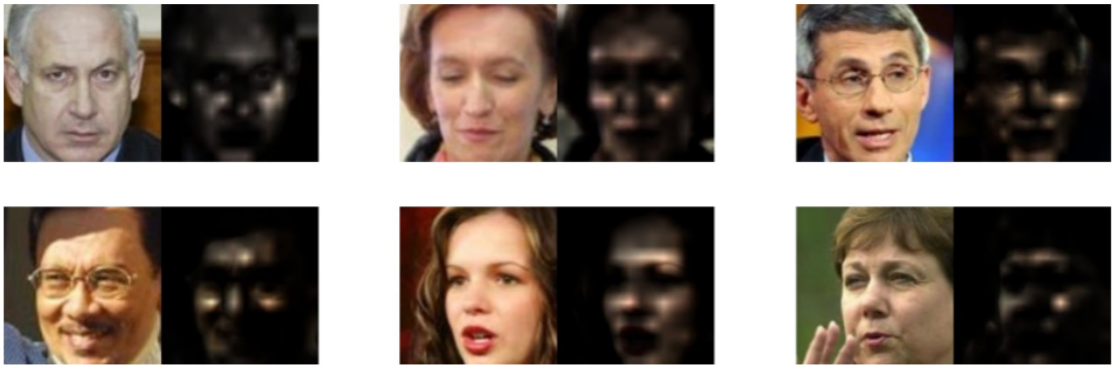
paper:Face Transformer for Recognition [pdf]

前言：Transformer在NLP领域目前已经打败RNN成了主流方法，因其全局性和并行性的优点，人们开始尝试将Tranformer引入CV领域。开山之作为2020年提出的ViT，当时在各大数据集上达到了和CNN不相上下的结果。本篇论文是将Transformer结构引入人脸识别的第一篇论文，证明了Tranformer对人脸数据的可行性，但是对于网络结构没有太多的探讨。笔记文末会给出近几年Transformer结构的改进方向。

论文动机：

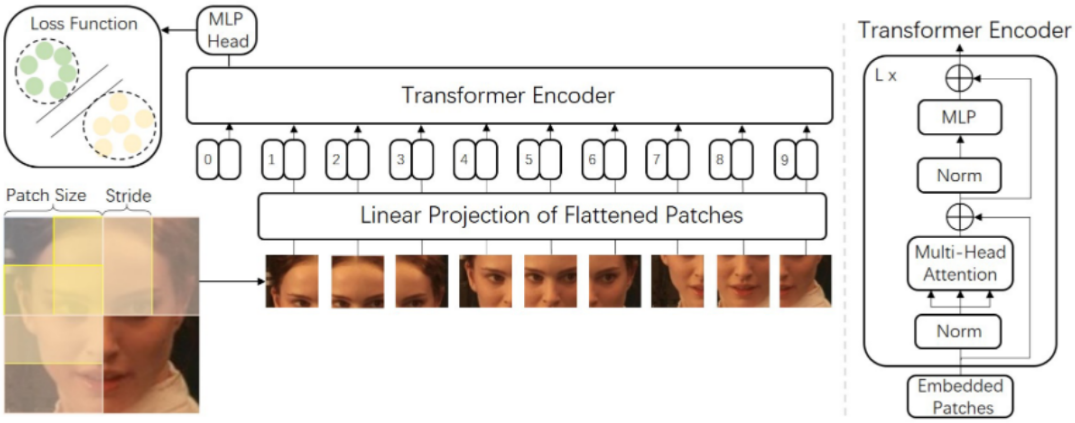
确定人脸识别在应用Transformer模型的可行性，忽略效率问题。文章证明了在大规模数据库上训练的 Transformer 模型获得了与具有相近参数量 CNN 相当的性能。此外Transformer 模型关注的面部区域是合理的。

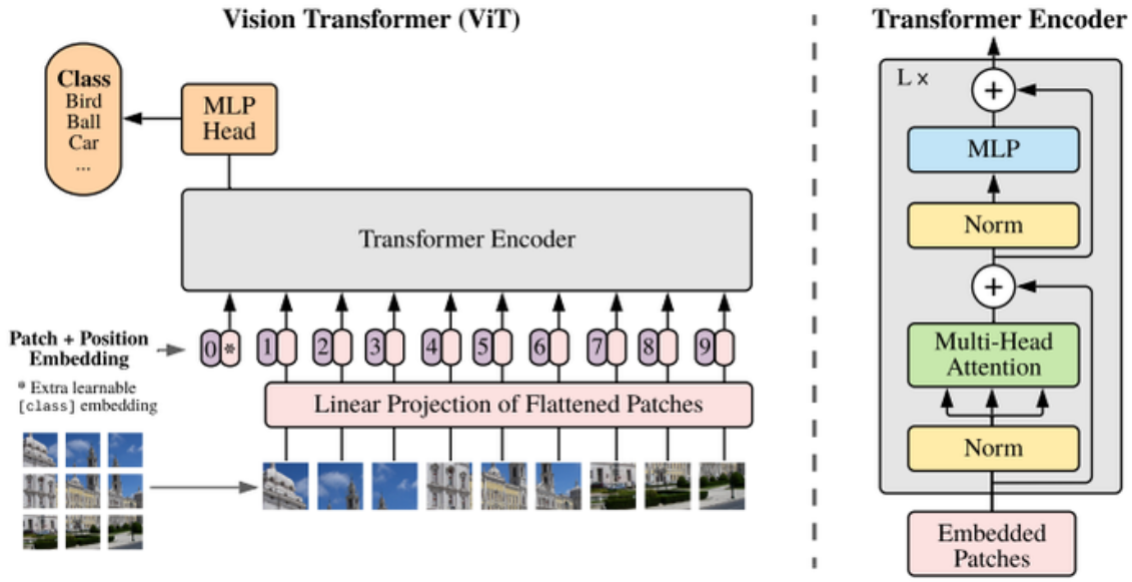
下图为论文可视化出的人脸权重较高的部分，这些权重较高的像素值点其实也就是区分不同人脸时的局部特征



网络模型

笔记中罗列了face transformer和ViT的网络结构图，可以看到ViT中的图像是直接将图像切分成3x3的patch，但是face transformer通过 一个滑动窗口具体来说，一个patch的大小为原图像的一半，将其看作一个滑动窗口从图像左上角开始，以二分之一的Patch size进行滑动，之后可以得到9个patch，这样就在没增加任何运算量和存储量的前提下，增加了patch之间的信息。





实验数据分析说明

从文中的数据可以看出在CASIA-WebFace数据集上，ViT的表现要比ResNet差很多，这很可能是因为CASIA数据集的体量不够大的原因，在ViT原文中提到Transformer 在大型数据集上的表现要比CNN网络好，但是在一些小型数据集上却表现不佳，这主要是因为Transformer缺少CNN所拥有的归纳偏执，所以Transfomer的解空间的个数或维度要比CNN多很多，需要更多的epoch去收敛。此外实验数据也说明了使用重叠的patch有助于网络的学习。

Training Data	Models	LFW	SLLFW	CALFW	CPLFW	TALFW	CFP-FP	AgeDB-30
CASIA-WebFace	ResNet-100 [12]	99.55	98.65	94.13	90.93	53.17	96.30	95.50
	ViT-P8S8 [1]	97.32	90.78	86.78	80.78	83.05	86.60	81.48
	ViT-P12S8	97.42	90.07	87.35	81.60	84.00	85.56	81.48
MS-Celeb-1M	ResNet-100 [12]	99.82	99.67	96.27	93.43	64.88	96.93	98.27
	ViT-P8S8 [1]	99.83	99.53	95.92	92.55	74.87	96.19	97.82
	T2T-ViT [5]	99.82	99.63	95.85	93.00	71.93	96.59	98.07
	ViT-P10S8	99.77	99.63	95.95	92.93	72.95	96.43	97.83
	ViT-P12S8	99.80	99.55	96.18	93.08	70.13	96.77	98.05

计算机视觉在使用时有很多落地痛点，遮挡就是最常遇见的一个，文章通过给图像加mask来测试网络的识别性能，与ResNet相比，Face Transformer在遮挡数据集上的鲁棒性并不比CNN好。

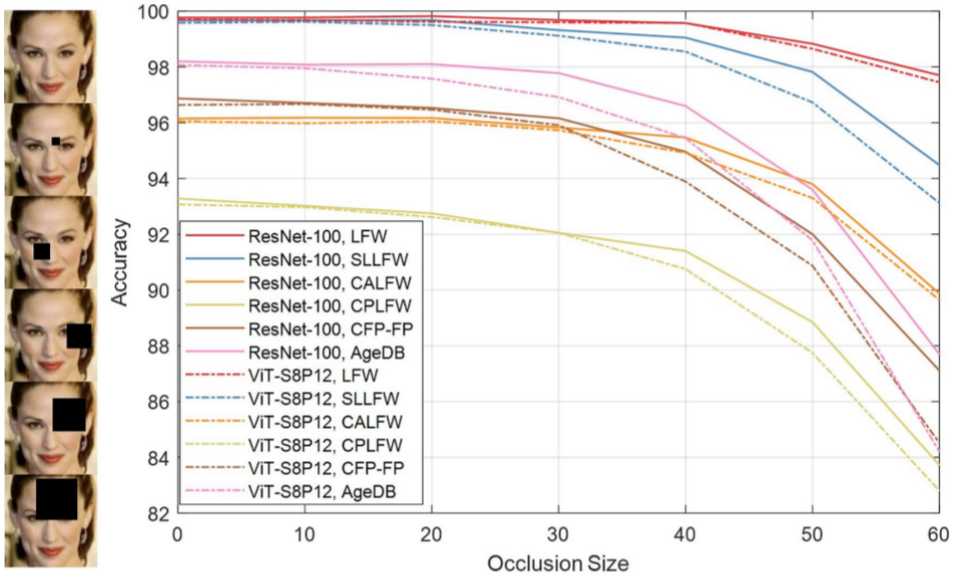


Fig. 4. The recognition performance of Face Transformer model and ResNet-100 as the occlusion area increases.

Transformer改进方向

缺点	原因	解决方案
数据需求量大	1. Self-Attention归纳偏置能力较CNN弱 2. 原始Patch Embedding为一个大卷积，参数量大，难以获得底层信息	1. 使用CNN作为Teacher Model，引入蒸馏损失辅助训练； 2. 使用多个卷积与池化结合的方法代替原始Patch Embeddding，或使用前CNN后Transformer的网络结构。
计算量大	1. SA的计算复杂度与token数的平方相关 2. Transformer计算过程中token数以及hidden size保持不变	1. 使用金字塔结构； 2. 使用局部窗口SA； 3. 引入Conv2D代替Linear； 4. 在SA过程的X->(Q,K,V)中对K，V的特征图进行池化。
堆叠层数数量受限	不同的Block之间的相似性随着模型的加深而增加；不同token之间的相似性随着随着模型的加深而增加(过度平滑，over-smoothing)	1. 增大hidden size; 2. 在注意力图softmax前后，对其在head维度进行线性变换以增加信息交互； 3. 深层Dropout rate增大； 4. 增加相似度惩罚损失项.
模型本身无法编码位置	进行位置编码	1. 使用固定位置编码 2. 使用可学习绝对位置编码 3. 使用可学习相对位置编码 4. 使用卷积的空间不变性编码位置信息

缺点	原因	解决方案
数据需求量大	1. Self-Attention归纳偏置能力较CNN弱 2. 原始Patch Embeddding为一个大卷积，参数量大，难以获得底层信息	DeiT:引入Distillation token，训练过程中引入蒸馏损失; CeiT:使用多个卷积与池化代替ViT中的一个大卷积；在FFN中间引入DWConv2d，融合不同Patch的信息； LV-ViT:使用Re-labeling技术给予每个Patch一个软标签进行辅助训练; Early Convolutions Help Transformers See Better:探讨如何使用多个卷积与池化代替ViT对性能的影响； CoAtNet:结合DWConv与SA为新算子，使用前MBConv后Transformer Block的方式构造网络。
计算量大	1. SA的计算复杂度与token数的平方相关 2. Transformer计算过程中token数以及hidden size保持不变	CvT:引入金字塔结构，在stage之间通过Conv2d缩小token数量，同时扩大hidden size；在X->(Q,K,V)使用Conv2d/DWconv2d代替Linear； PVT:引入金字塔结构，在stage之间通过Conv2d缩小token数量；在X->(Q,K,V)中K，V使用额外的Conv2d来缩小特征图大小; Swin Transformer:引入金字塔结构，在stage之间通过拼接2*2范围内的像素点，再通过线性变换缩小token数量；使用局部窗口SA降低计算复杂度；使用窗口移动的方式使得前后两层不同窗口存在信息交流； Twins-PCPVT-S：相比PVT，在每一层后面引入DWConv2d作为相对位置编码；使用局部窗口SA与池化全局SA交替的方式减低计算量; LeViT:在stage之间通过一个对Q进行sample的Block缩小token数量；使用多个卷积与池化进行Patch Embedding；在网络中引入Conv2d+BN代替Linear。

缺点	原因	解决方案
堆叠层数数量受限	不同的Block之间的相似性随着模型的加深而增加；不同token之间的相似性随着随着模型的加深而增加(过度平滑，over-smoothing)	增大hidden size DeepViT、Talking-Heads Attention：在注意力图softmax前(前后)，对其在head维度上进行线性变换 CaiT：使用LayerScale，使用可学习参数对FFN/SA的直接输出在hidden size维度方向做不同的缩放；深层Dropout加大；使用Talking-Heads Attention。 DiversePatch：增加相似度惩罚项；增加对比损失使深层特征与浅层对应patch特征接近；引入使用CutMix后每个Patch对应的类别信息辅助训练。 Refiner：在经过softmax的注意力图使用线性变换在head维度升维，经过DWConv2d增强局部信息交互，再通过线性变换映射回原维度。
模型本身无法编码位置	进行位置编码	DETR：使用固定位置编码 ViT：使用可学习绝对位置编码 Swin Transformer：使用可学习相对位置编码 CPVT、CvT：使用卷积的空间不变性编码位置信息

总结

本篇论文是将Transformer结构引入人脸识别的第一篇论文，证明了Tranformer对人脸数据的可行性，但是对于网络结构没有太多的探讨，大多都是围绕着跑出来的结果好不好，哪里不好，但是对原因可解释这块并没有做太多的说明。实际上针对网络的改进还有很多的任务可以做，比如位置编码、模型层数、与CNN、RNN结合，都是一些不错的方向。