Deep video

深度学习时代使用卷积神经网络去处理视频理解的最早期的工作之一
将卷积神经网络从图片识别应用到视频识别里面
视频和图片的区别就是多了一个时间轴，有更多的视频帧而不是单个的图片，所以自然是有几个变体是需要尝试的,如下：

- single frame：只用单帧做理解，和图像分类没区别
- late fusion：在视频中随机选几帧，每一帧单独通过一个神经网络，这两个神经网络是权值共享的，然后把得到的特征合并一下，通过FC层最后做一下输出，这个做法的本质还是单帧经过神经网络得到一个特征，像图片分类，但是最后把特征合并起来了，所以稍微有一点时序上的信息在里面
- early fusion：在输入层面做了融合，具体做法就是把五个视频帧在RGB的channel上合起来，变成15个channel，这意味着网络的结构需要有一点改变了，第一个卷积层接收输入的通道数要变为15，之后的网络跟之前保持不变，这种做法，可以在网络的刚开始输入的层面感受到时序上的改变，希望能学到一些全局的运动时间信息
- slow fusion:希望能够在网络学习的构成中的特征层面做一些合并会更好一些，具体做法是每次选择10个视频帧的视频段，然后每4个视频帧经过一个神经网络抽取特征，刚开始的层全局共享，抽取最开始的特征之后，由最开始的四个输入片段合并成两个片段，再做一些卷积操作获得更深层的特征，然后把特征交给FC做最后的分类

结果四种方法的结果都差别不大，即使是在100万个视频上做了预训练之后，在UCF101上做迁移学习时还比不上之前的手工特征。

另外一条思路，多分辨率卷积神经网络结构：
把输入分成两个部分，一个是原图，另外一个从原图的正中间抠出一部分变成一个输入，
因为对于图片或者视频来说最有用的或者物体都会出现在正中间，所以把上面的分支交fovea（人脸视网膜里最中心的东西，对外界变化最敏感的区域） stream，
下面分支叫contex stream(图片整体信息)，作者想通过这个操作既获得图片中间最有用的信息，
又能学习到图片整体的理解，看看这样能不能提升对视频的理解，两个有两个网络权值共享，可看成早期对注意力使用的方式

双流网络

- 手工特征中有一种叫光流的特征，有效表达了物体运动信息。（Two-Stream Convolutional Networks for Action Recognition in Videos）论文将光流作为深度学习的输入，结合原始视频帧。做视频理解任务，效果很好
- 空间流和时间流分别经过softmax后做class score fusion
- optical flow(光流)：提取出物体运动的方向和速度,颜色表示方向，亮度表示速度。

缺点

- 不是完全end-to-end的视频分析，需要离线计算光流，计算光流比较耗时，没法达到实时
- 解决的是short-term video分析，没法有效的解决long-term video分析。

I3D网络

- 使用了新的数据集Kinetics重新评估了当前最新的模型架构，Kinetics数集有400个人体行为类别，每个类别有400多个clips。替换了数据集HMDB-51和UCF-101，这些数据集，如UCF-101和HMDB-51的视频数量都比较少，很多模型因此都获得了比较接近的效果，没法有效的对模型性能进行评价
- 这些数据来自真实有挑战的YouTube视频。作者提出的双流膨胀3D卷积网络(I3D)，该网络是对一个非常深的图像分类网络中的卷积和池化kernel从2D扩展到了3D，来无缝的学习时空特征。并且模型I3D在Kinetics预训之后，I3D在基准数据集HMDB-51和UCF-101达到了80.9%和98.0%的准确率。
- 作者通过实验重新实现了许多有代表性的神经网络结构，之后通过对这些网络在Kinetics上预训练，之后对这些网络在HMDB-51和UCF-101数据集上进行微调来分析他们的迁移行为
- 作者基于在ImageNet上预训练的带BN的InceptionV1为骨干，搭建了五种行为分类网络，其中四种为基于以前论文搭建的网络，最后一种作者提出网络I3D

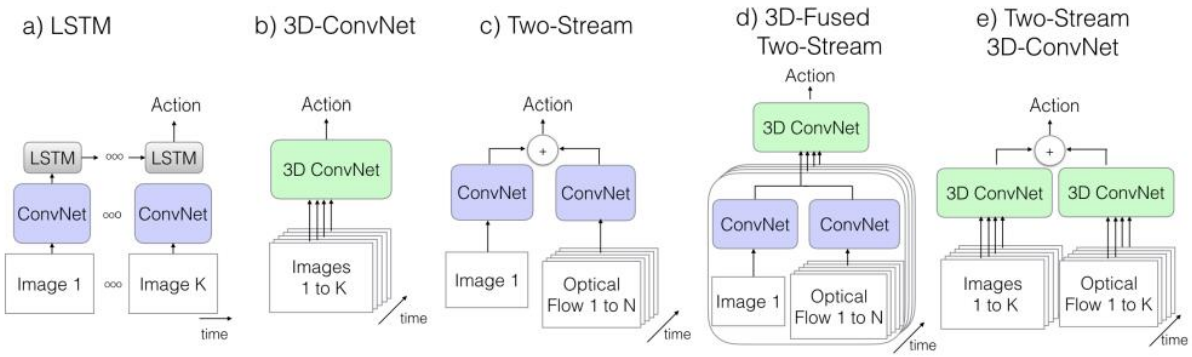
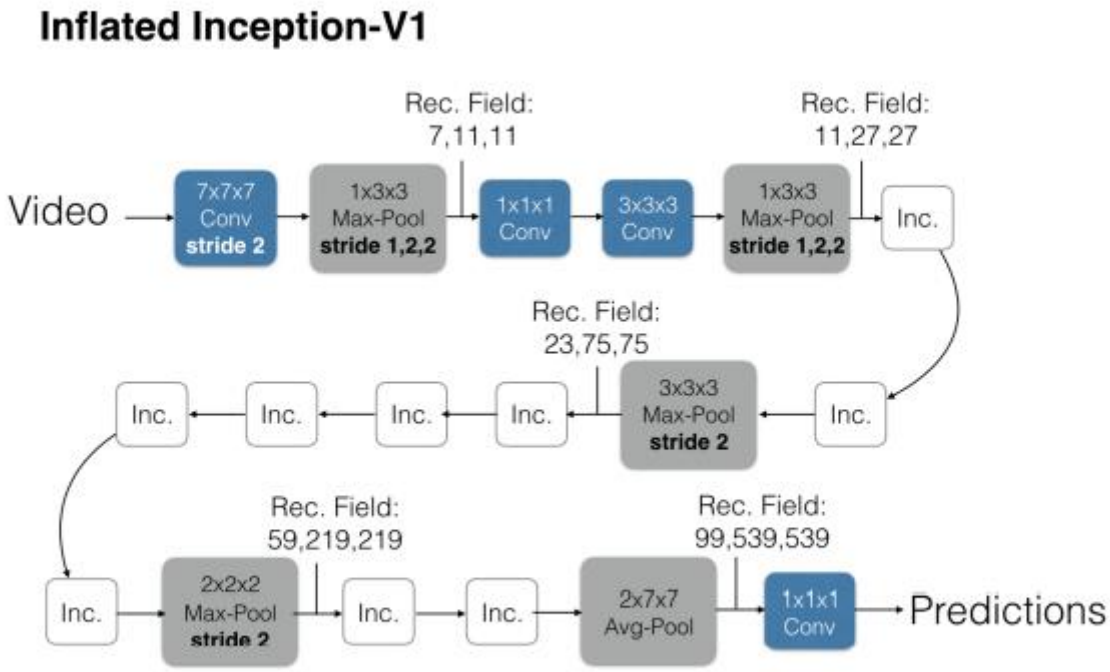
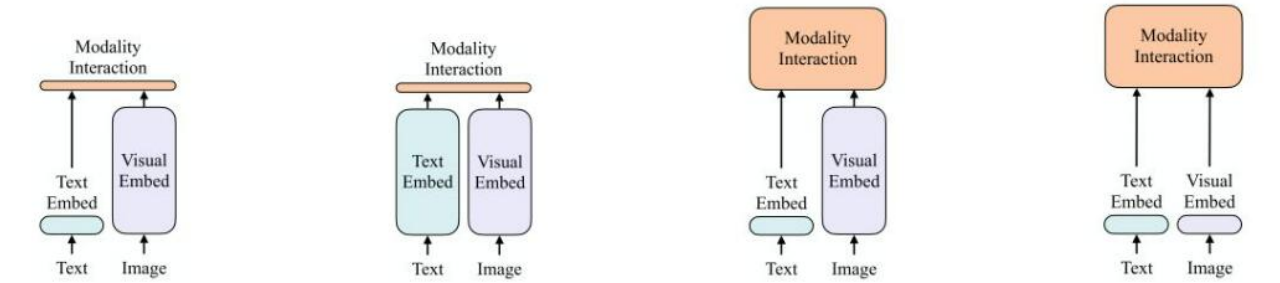


Figure 2. Video architectures considered in this paper. K stands for the total number of frames in a video, whereas N stands for a subset of neighboring frames of the video.



- 通过在时间维度上重复2D卷积的权重N次，并且通过除以N来重新缩放它们来初始化模型参数。把2D卷积的权重，扩展到3D卷积上

ViLT



VSE、VSE++和SCAN属于(a)类型。对图像和文本独立使用encoder，图像的更重，文本的更轻，使用简单的点积或者浅层attention层来表示两种模态特征的相似性。

CLIP属于(b)类型。每个模态单独使用重的transformer encoder，使用池化后的图像特征点积计算特征相似性。

ViLBERT、UNTER和Pixel-BERT属于(c)类型。这些方法使用深层transformer进行交互作用，但是由于VE仍然使用重的卷积网络进行特征抽取，导致计算量依然很大。

作者提出的ViLT属于(d)类型。ViLT是首个将VE设计的如TE一样轻量的方法，该方法的主要计算量都集中在模态交互上。

Modality Interaction Schema
模态交互部分可以分成两种方式：一种是single-stream(如BERT和UNITER)，另一种是dual-stream(如ViLBERT和LXMERT)。其中single-stream是对图像和文本concate然后进行交互操作，而dual-stream是不对图像和文本concate然后进行交互操作。ViLT延用single-stream的交互方式，因为dual-stream会引入额外的计算量。

模态交互部分可以分成两种方式：一种是single-stream(如BERT和UNITER)，另一种是dual-stream(如ViLBERT和LXMERT)。其中single-stream是对图像和文本concate然后进行交互操作，而dual-stream是不对图像和文本concate然后进行交互操作。ViLT延用single-stream的交互方式，因为dual-stream会引入额外的计算量。

现有的VLP模型的text embedding基本上都使用类BERT结构，但是visual embedding存在着差异。在大多数情况下，visual embedding是现有VLP模型的瓶颈。visual embedding的方法总共有三大类，其中region feature方法通常采用Faster R-CNN二阶段检测器提取region的特征，grid feature方法直接使用CNN提取grid的特征，patch projection方法将输入图片切片投影提取特征。ViLT是首个使用patch projection来做visual embedding的方法。

- 在本文中，作者提出了一种简单的VLP结构——视觉和语言Transformer，极大的降低了多模态预训练模型的复杂度，在embed的时候采用了最简单的结构，并且也达到了不错的性能，最重要的是能够让模型的速度大幅度提升。不过，作者在进行参数初始化的时候还是用到了ViT的预训练参数，这也导致了对于模型结构修改的空间就比较小。因为如果模型改变太大，就不能用预训练好的参数初始化，从而性能也会降低。

TimeSformer

- 将VIT用于视频理解，取多帧，每一帧的图像都分成一个一个的小patch，之后也可以直接送入transformer，输入的patch多了几倍。从原理上来说这样是可行的，但是就如同3D卷积一样，这样的计算量也是难以接受的，特别是对于视频时间相对长一些的数据来说，需要提取的帧数也要随之增加。本文中作者实验了五种不同的方式，最终发现了所谓的divided space-time attention
- divided space-time attention：在time attention 中，每个图像patch仅和其余帧在对应位置提取出的图像patch进行 attention操作。在space attention 中，图像patch和其他帧的提取出的图像对应位置的patch进行attention操作。然后这个图像patch和同一帧的提取出的图像patch进行attention操作。