本周主要看了 R-C3D 和 SSN 两篇文章，前者思路和 faster rcnn 类似，后者采用金字塔结构对视频序列进行处理，对两篇文章一些不明白的点进行了记录。

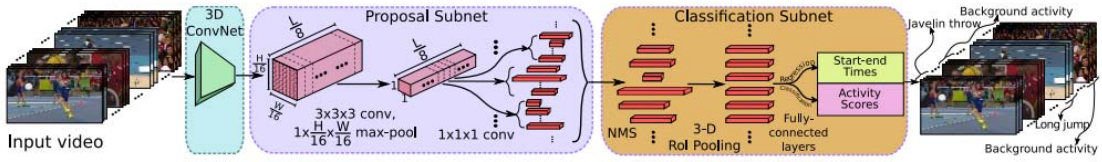
R-C3D:

NMS: 通过某种方式对一个类别所有 box 进行按照一个 score 从排序（例如每个 box 的置信度由高到低排序），选取排序中第一名的 box，然后在序列中去除和这个 box 的 IoU 大于 threshold 的 box，再在剩下的 box 中找排名第一的 box 重复此步骤直至遍历完所有 box。为了改进在 NMS 算法中两个 ground truth 的 IoU 过大而出现误判，提出了 soft-NMS 算法，将和排名第一的 box 的 IoU 大于 threshold 的 box 不直接删除，而是将他的 score 改成原 score 与 IoU 构成的一个高斯函数。

Subnet training: Proposal Subnet 部分 IoU 大于 0.7 为正样本，小于 0.3 为负样本，其他 proposal 不进行训练，正负样本比例为 1:1。Classification Subnet 部分取与 ground truth 的 IoU 大于 0.5 且最高的 proposal 作为正样本，与所有 ground truth 的 IoU 低于 0.5 的 proposal 作为负样本，正负样本比例为 1:3。两部分用类似的损失函数进行训练。

THUMOS14: 数据集用来进行 action recognition 和 temporal action recognition 两个任务。Action recognition 的数据集分为四部分：training（UCF101 数据集，经过修整不含有无意义序列段），validation（和 UCF101 类别一样的 1000 个视频，每个视频的标签有一个主要类别和一个次要类别，且视频没有经过修整（untrimmed）），background（2500 个视频，类别和 UCF101 类别不同但有一定关系（如空的篮球场和扣篮）），test（1500 个视频，含有一个或多个或不含有给定类别）。Temporal action recognition 的数据集用了 action recognition 数据集的部分，但只取了 20 类视频，validation 中给定了 temporal ground truth，只有 training 视频是 trimmed。

C3D 网络结构：所有卷积层都用 3*3*3（D*H*W）的 kernel，pool1 使用 1*2*2，后面的 pool 为 2*2*2



Pipeline: 途中 3D ConvNet 为 C3D 网络中 conv1a 到 conv5b 的部分，原视频(size 为 112*112 长度只受到 GPU 内存约束)序列经过 C3D 网络得到 512*L/8*H/16*W/16 的 feature map，先通过一个 3*3*3 的 filter 增加感受野，再经过一个 1*H/16*W/16 的 max-pool 得到 L/8 个 512 维的特征向量（downsample）。将 L/8 的每一层作为 anchor segment 的中心点，通过对每个中心点生成 scale 不同的 K 个长度，总共会生成(8/L)*K 个 anchor segment。在 512*L/8*1*1 的 feature 后面通过 1*1*1 的卷积来生成 proposal offsets 和 scores 来确定哪些视频段是正样本段。Classification Subnet 部分先通过一个阈值为 0.7 的 greedy NMS 对 proposal 样本进行清理，之后通过 3D ROI pooling 对不同长度的 feature 进行 maxpool 最后得到 size 为 512*1*4*4 的 feature map。最后通过两个全连接层分别得到 label 以及对应 temporal 区间。

SSN(structured segment networks):

Three-Stage Structures: augmented proposals 是将原给定的长为 d 的 proposals 首尾分别增加 d/2，当 proposal 与 ground truth 匹配时，对应的 augmented proposal 将包含这段 activity 的开头和结尾部分，即可将 augmented proposal 分为 starting, course, ending 三段。

Sparse snippet sampling scheme: 稀疏片段采样，每个视频按取一些片段训练，由此来取消一些冗余信息并克服长期模型所带来的计算困难。文中采取的方法是将 augmented proposals 平均划分为九段，再从每段中随机取一个片段进行后面的处理。

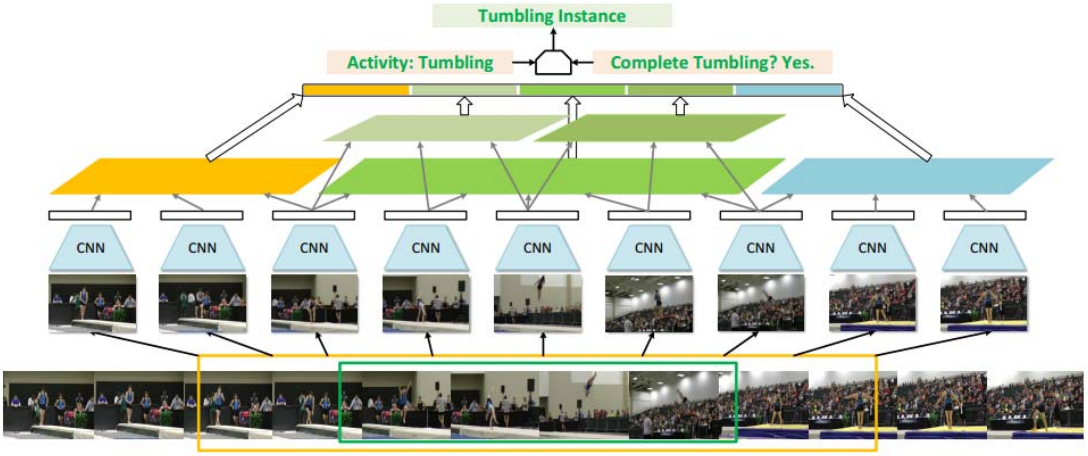
Smooth L1 loss:

$$\text{smooth}_{L_1}(x) = \begin{cases} 0.5x^2 & if\ |x| < 1 \\ |x| - 0.5 & otherwise \end{cases}$$

Inference with reordered computation: 以六帧图片作为一个 snippet，先进行分类和回归的线性响应再做池化，由此来降低对不同 proposal 池化的冗余处理。

Temporal region proposals(TAG): 通过 actionness classifier 可以得到每一个 snippet 序列中每帧属于正样本的概率，通过调整这个概率的阈值可以得到许多小段，从头开始将连续的小段以及之间的空缺视为 proposal，当小段长度占整体 proposal 长度的比例低于一个阈值时重复步骤生成下一个小段，最终得到所有 proposals，文中将这个两个阈值均取为 0-1 区间内步长为 0.05 的相同值（0.05,0.1,0.15……）。

STPP(Structured Temporal Pyramid Pooling): 将 CNN 输出的特征进行平均池化得到特征向量，在 course 段用了两层共三个池化层进行池化，每个 snippet 都会生成一个特征向量，将不同 stage（staring，course，ending）中 snippet 生成的特征向量按照长度做平均，最终可得到包含五段特征向量的集合。



Pipeline: 本文的关键点是将一个行为序列分成 starting，course，ending 三段来确保一个动作的完整性以及其在整个时间序列上的精度。通过稀疏片段采样将 proposal 分成九段分别经过 CNN 网络（此处使用的 two-stream 网络）输出特征，再经过 STPP 结构得到特征向量。特征向量通过 activity classifier 判断类别（通过 course 部分得到的特征向量），再通过这个类别的 completeness classifier 确定这个行为是否完整（通过 staring，course，ending 三部分得到

的整个特征向量）。