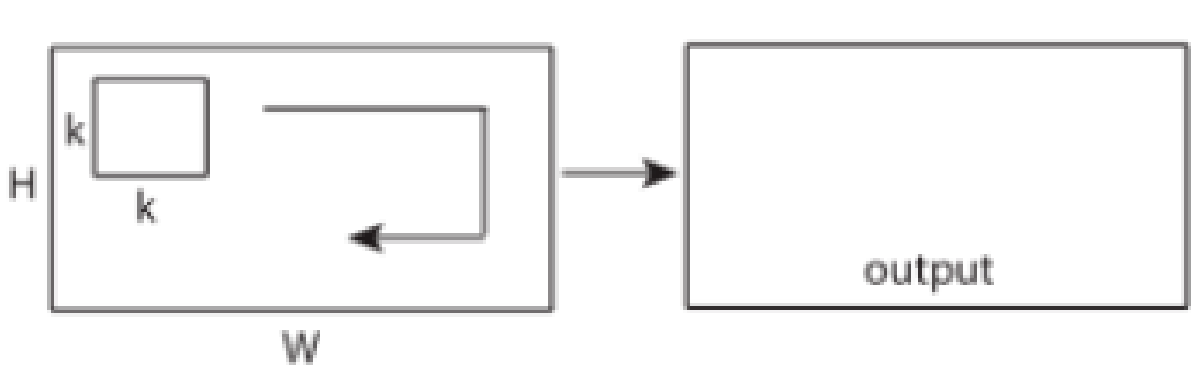
Pseudo-3D residual networks

动作识别

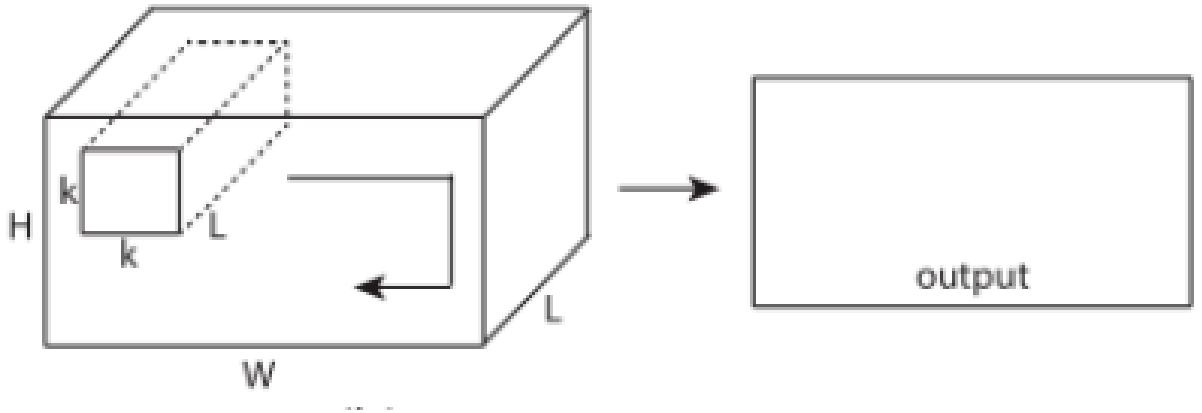
动作识别的主要目标是判断一段视频中人的行为的类别，动作识别任务涉及从视频剪辑（一串二维帧序列）中识别不同的动作，其中的动作可能贯穿整个视频，也可能不会。这有点儿像图像分类任务的一种自然扩展，即在多帧视频中进行图像识别，然后从每一个帧中聚集预测结果。

三维卷积核法

2D卷积网络输入图像会产生图像，输入视频输出的也是图像。

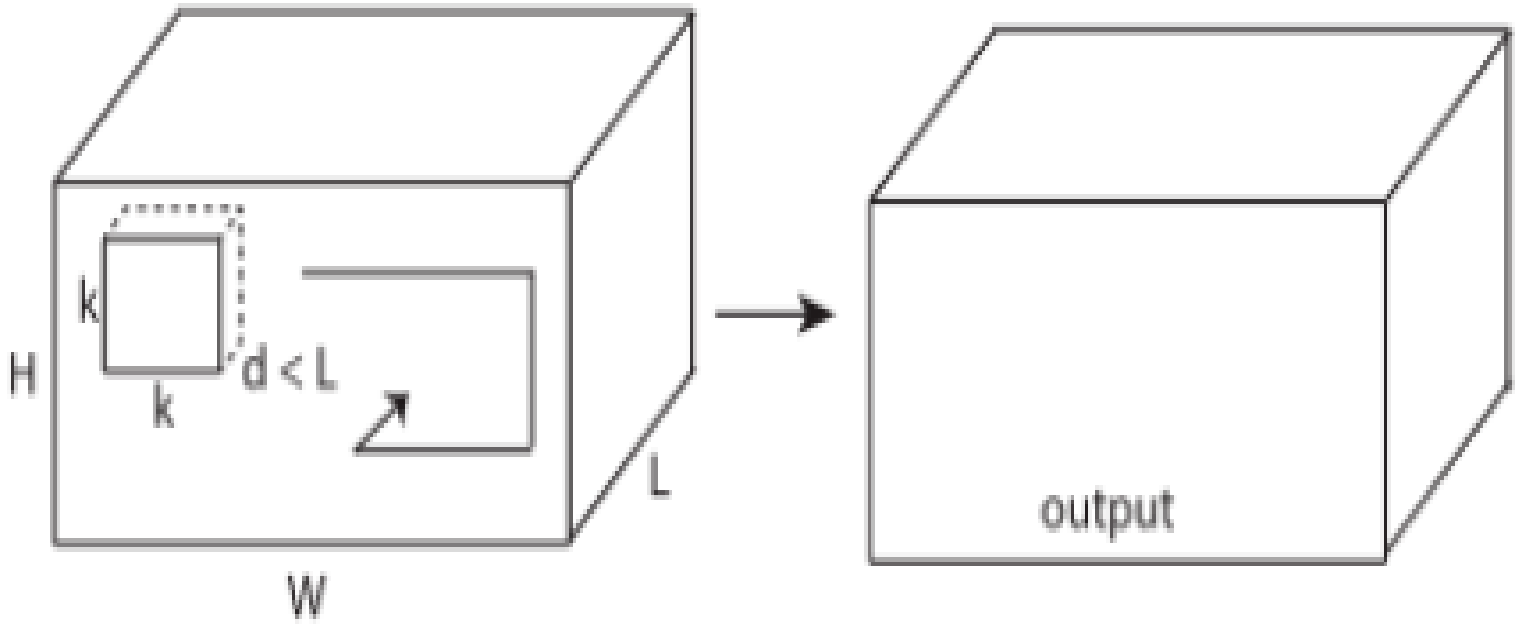


(a) 2D convolution

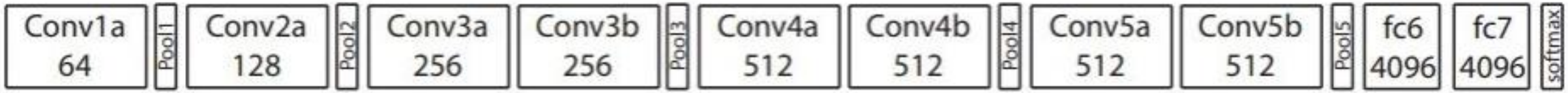


(b) 2D convolution on multiple frames

3D卷积网络输入视频,输出仍然为3D的特征图,保留输入的时间信息。



(c) 3D convolution



Learning Spatiotemporal Features with 3D Convolutional Networks

论文地址：http://vlg.cs.dartmouth.edu/c3d/c3d_video.pdf

网络有8个卷积层（filter:3×3×3，stride: 1×1×1），5个池化层（filter:2×2×2，stride: 2×2×2，除了第一个filter:1×2×2，stride: 1×2×2），2个全链接层（4096），和1个softmax分类层。

name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
nv1	112×112	7×7, 64, stride 2				
v2_x	56×56	3×3 max pool, stride 2				
		$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}$
v3_x	28×28	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}$
v4_x	14×14	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times23$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}$
v5_x	7×7	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

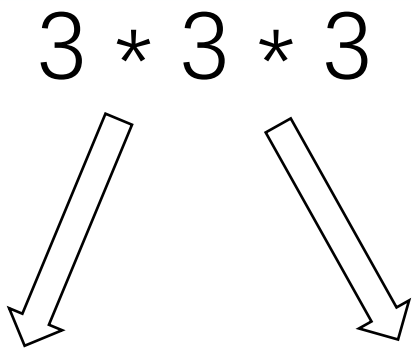
ResNet 形式结构

研究背景：

P3D主要解决问题：

1、原始3D卷积计算量参数量过大，因此采用了分开的卷积；

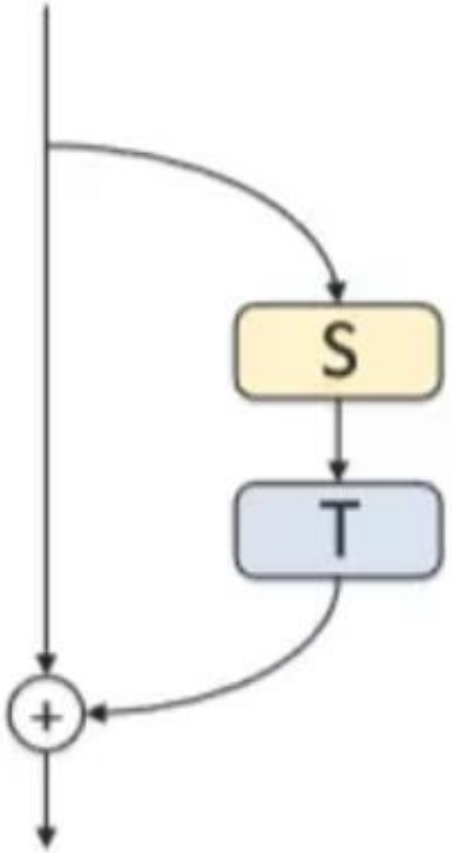
2、是如何构建更深的网络，它这里仿照了resnet去解决这个问题



(1)第一个问题是关于空间维度（S）上的2D滤波器和时域（T）上的1D滤波器的模块应该直接或间接地相互影响。

(2)第二个问题是两种滤波器是否都应直接影响最终输出。

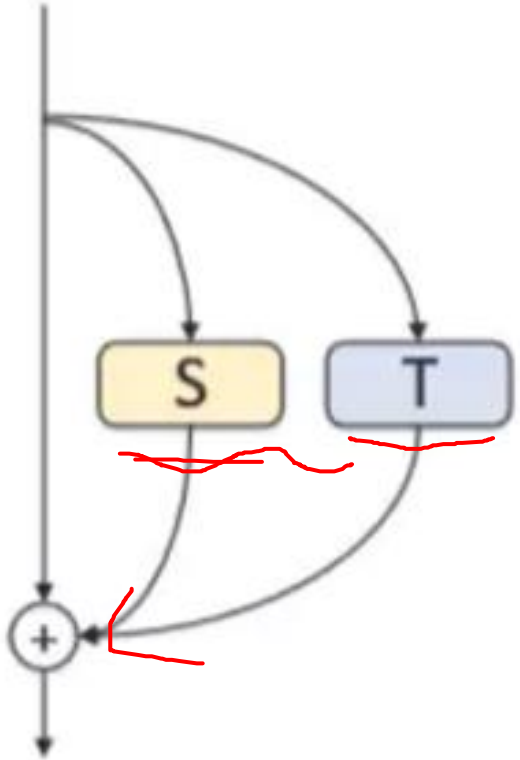
P3D 块 结 构



(a) P3D-A

这是一种S与T直接影响的方式，先对feature map做空间的2D卷积然后再做时间1D卷积，最后时间卷积的结果与shorcut一起构成残差块的输出结果。

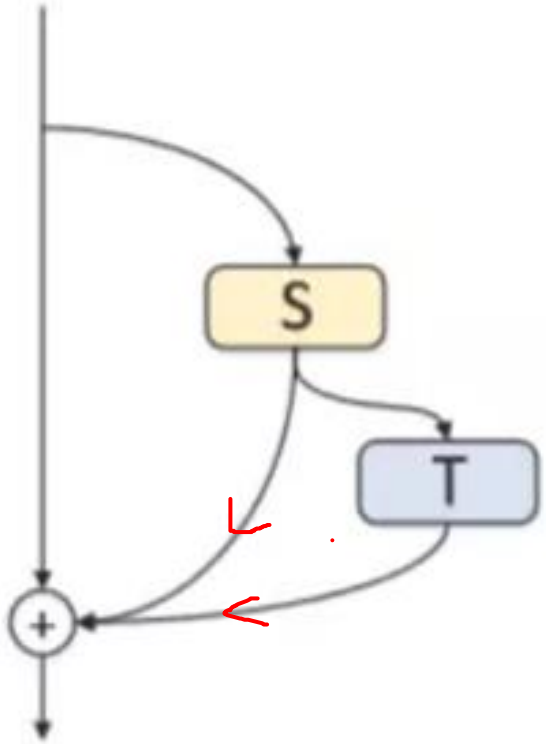
P3D 块 结 构



(b) P3D-B

两者是以并行的方式对feature map进行卷积操作，最终两个的结果直接累加到shortcut中构成残差块的输出结果。

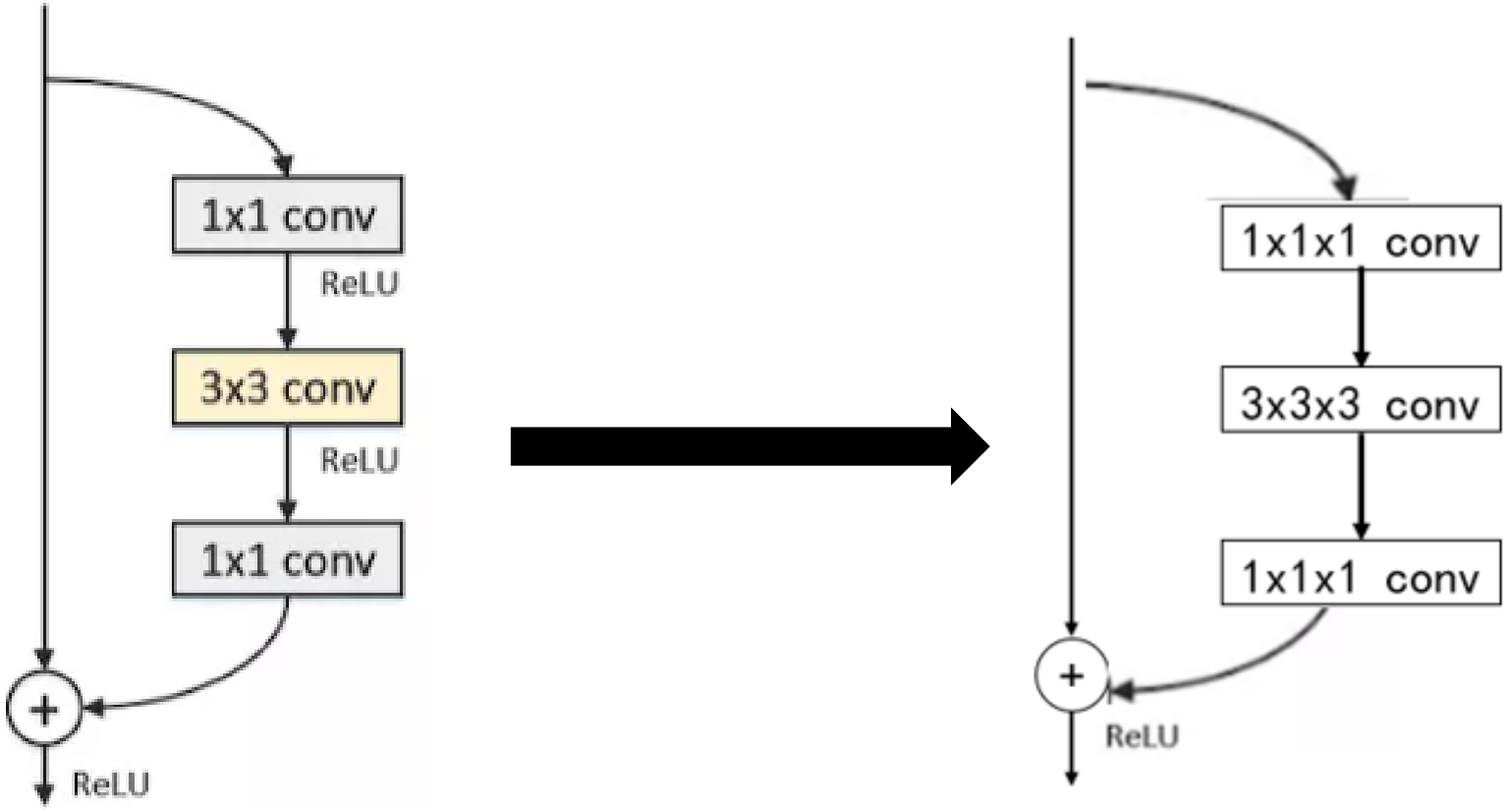
P3D 块 结 构



(c) P3D-C

这种方式是前面两种方式的一种结合，除此还建立了S到最终结果的shortcut。

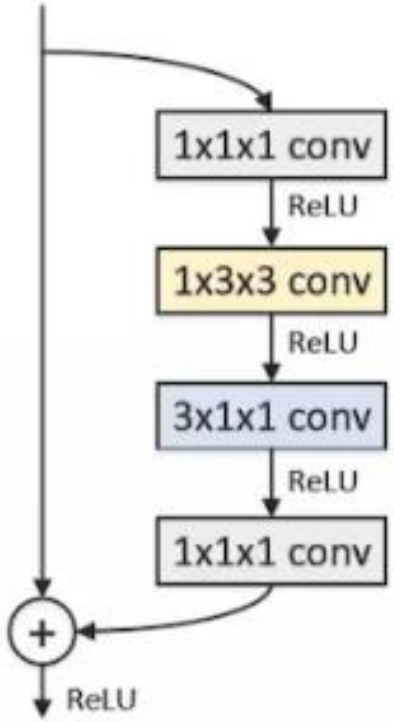
Bottleneck 结构设计



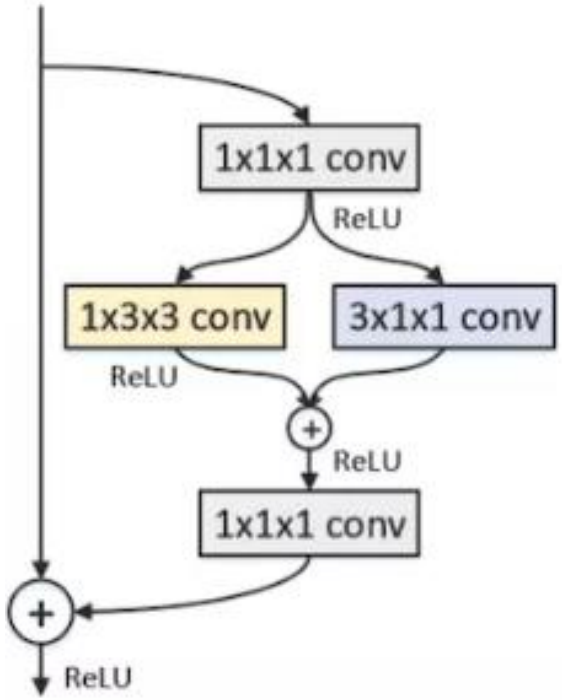
Residual Unit

3D Residual Unit

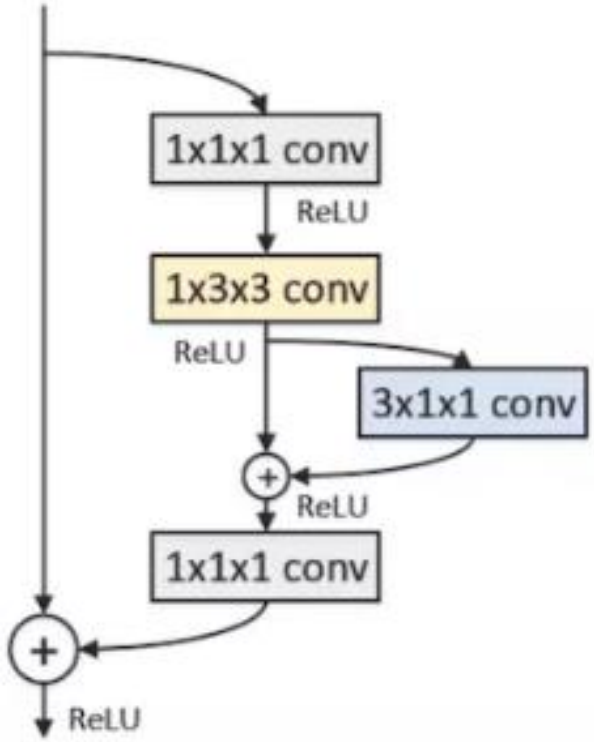
Bottleneck 结构设计



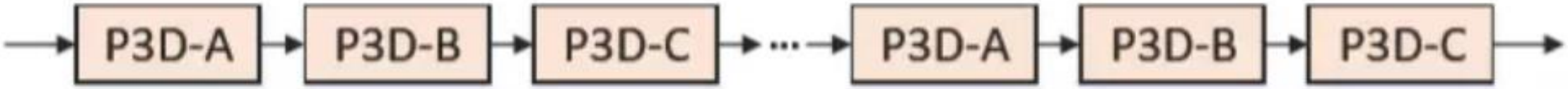
(a) P3D-A



(b) P3D-B



(c) P3D-C



P3D ResNet

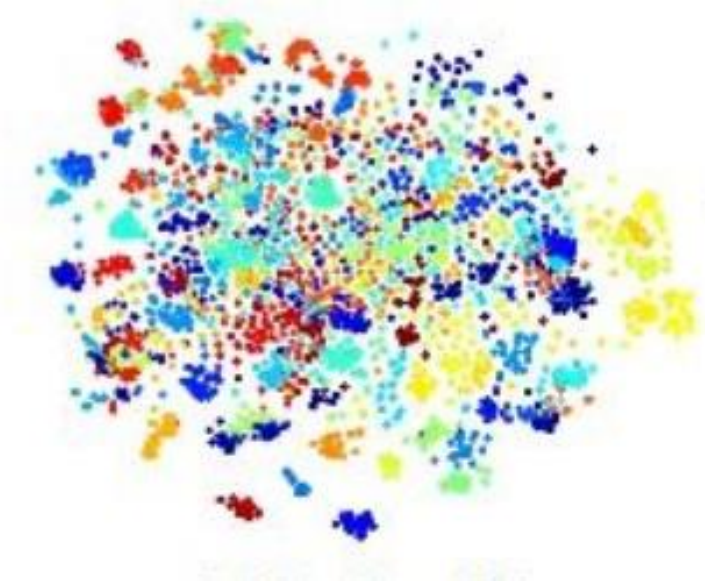
先用特定的一种块结构代替ResNet50中的残差单元，得到P3D-A ResNet，P3D-B ResNet，P3D-C ResNet。另外又从结构多样性的角度考虑将P3D-A,P3D-B,P3D-C三种结构块按序排列混合起来构成P3D ResNet。

Method	Model size	Speed	Accuracy
ResNet-50	92MB	15.0 frame/s	80.8%
P3D-A ResNet	98MB	9.0 clip/s	83.7%
P3D-B ResNet	98MB	8.8 clip/s	82.8%
P3D-C ResNet	98MB	8.6 clip/s	83.0%
P3D ResNet	98MB	8.8 clip/s	84.2%

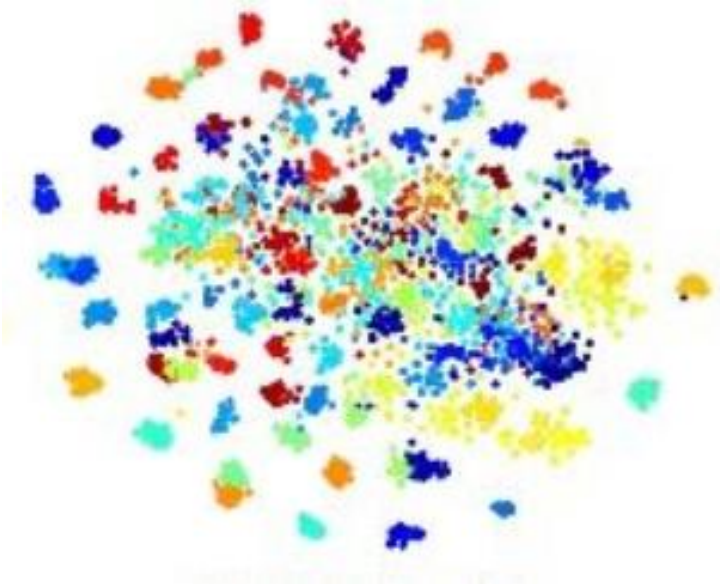
数据处理源码：
https://www.jianshu.com/p/4ebf2a82017b

P3D 模型源码：https://github.com/qijiezhao/pseudo-3d-pytorch

实验数据: UCF50
下载地址：https://www.crcv.ucf.edu/data/UCF50.php



(a) ResNet- 152



(a) P3D ResNet

1、时间域和空间域分开，并且将其灵活的进行组合，增加了网络的多样性

2、与C3D网络相比较，增加的网络的深度，提高了分类的准确性