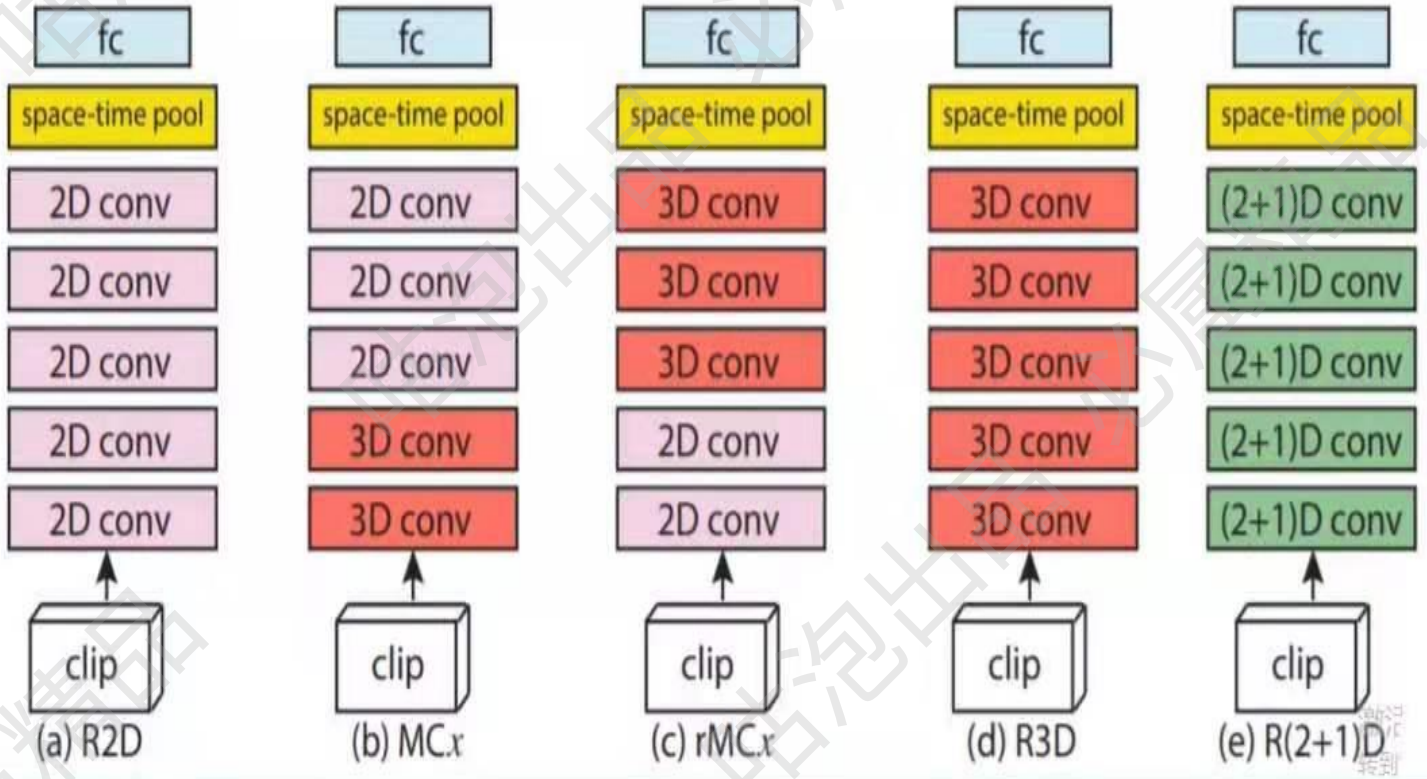
R(2+1)D网络

- 时序数据处理方法：

  - 视频数据经典处理方法，多一个时间维度（每一帧图像组成了视频）



R(2+1)D网络

- 各种方法概述：
  - R2D：把时间维度堆到特征图个数
  - 例如30帧的视频：30*3*224*224转换成90*224*224
  - 放弃了时间维度信息，直接转换成了特征图，方法简单但是效果一般
  - 只是为了能进行计算，还是2D卷积操作

R(2+1)D网络

- 各种方法概述：
  - C3D：3D卷积直接处理视频数据
  - 3D卷积中多了一个维度来处理多少帧的图像
  - 后来又提出来R3D，其实就是基本网络结构用resnet来做
  - 操作很简单，但是效果的话还是需要对比实验来看了

R(2+1)D网络

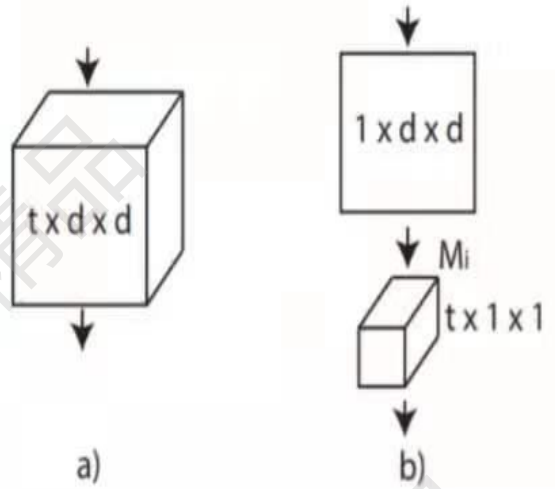
- 各种方法概述：
  - MCX：相当于2D和3D的混合方法
  - 但是先2D还是先3D呢？时间信息是高阶还是低阶好呢？
  - 更多的实验给出的是还是把3D卷积放在后面合适
  - 方法也比较简单，实验效果也还不错

R(2+1)D网络

- 各种方法概述：
  - R(2+1)D：拆分3D卷积为2D卷积+1D卷积
  - 2D卷积效果一般，3D的参数又比较多，能不能拆分一下呢？
  - 其中2D表示空间卷积，1D表示时间卷积
  - 相当于串联了两个模块，先2D再1D进行特征提取

R(2+1)D网络

- 各种方法概述：
  - 2D卷积正常提取图像区域特征
  - 1D卷积只考虑时间维度特征
  - 效果综合来看还是不错的



Net	# params	Clip@1	Video@1	Clip@1	Video@1
Input		8×112×112		16×112×112	
R2D	11.4M	46.7	59.5	47.0	58.9
f-R2D	11.4M	48.1	59.4	50.3	60.5
R3D	33.4M	49.4	61.8	52.5	64.2
MC2	11.4M	50.2	62.5	53.1	64.2
MC3	11.7M	50.7	62.9	53.7	64.7
MC4	12.7M	50.5	62.5	53.7	65.1
MC5	16.9M	50.3	62.5	53.7	65.1
rMC2	33.3M	49.8	62.1	53.1	64.9
rMC3	33.0M	49.8	62.3	53.2	65.0
rMC4	32.0M	49.9	62.3	53.4	65.1
rMC5	27.9M	49.4	61.2	52.1	63.1
R(2+1)D	33.3M	**52.8**	**64.8**	**56.8**	**68.0**