


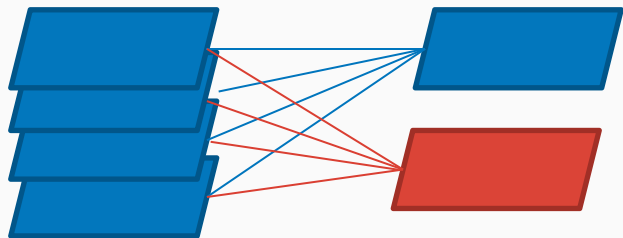
卷积变化与Xception,Mobilenet

Harry
2019.12.7

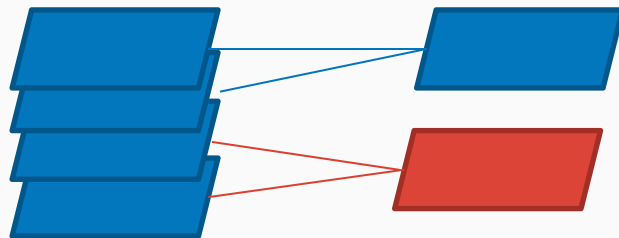
A light blue right-angled triangle is located in the bottom right corner of the slide, pointing towards the top-left.

卷积之间的联系与区别

conv

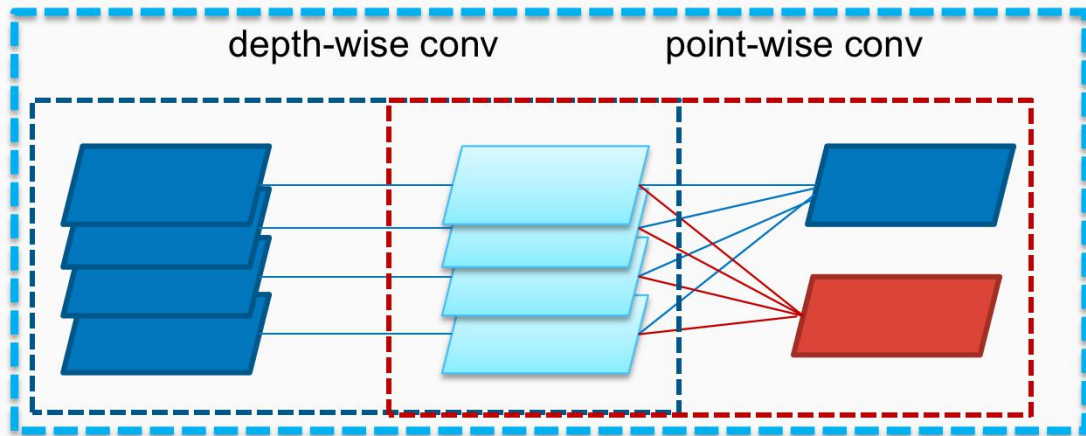


group conv



depth-wise conv

point-wise conv



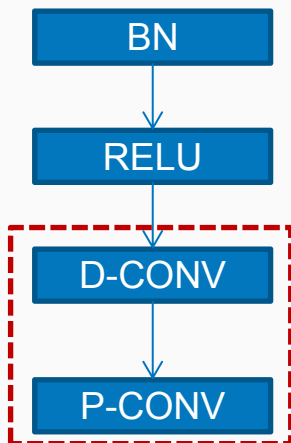
separable convolution

depth-wise conv

其一般与point-wise conv联合使用。联合使用时，可以看成其把传统卷积变成空间卷积与通道卷积分开进行运算，解耦操作之间的相关性，减少了参数量与计算量。

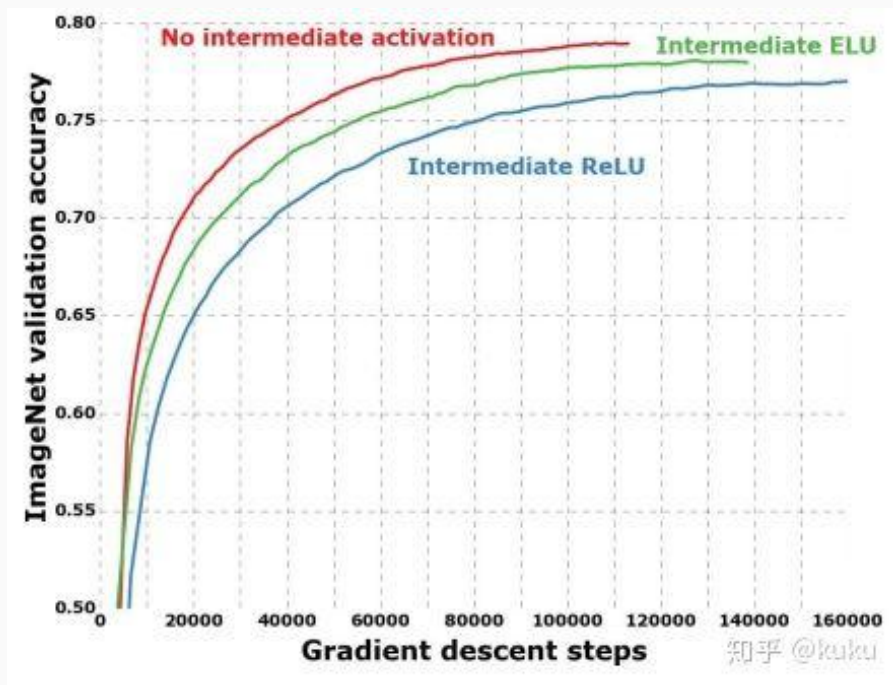
- 先用3x3的卷积建立空间之间的相关性（depth-wise）。
- 再用1x1的卷积建立通道间的相关性（point-wise）

常见block的堆叠方式



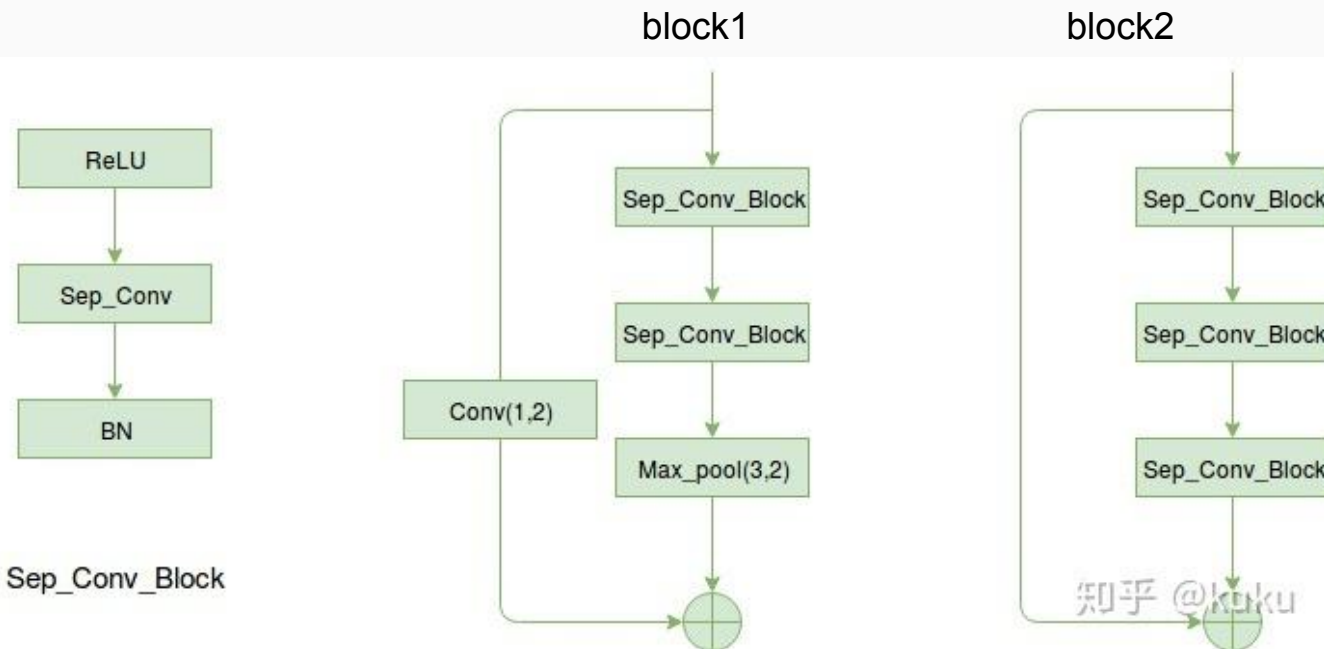
depth-wise conv

xception论文中D-Pconv之间使用激活函数的效果图



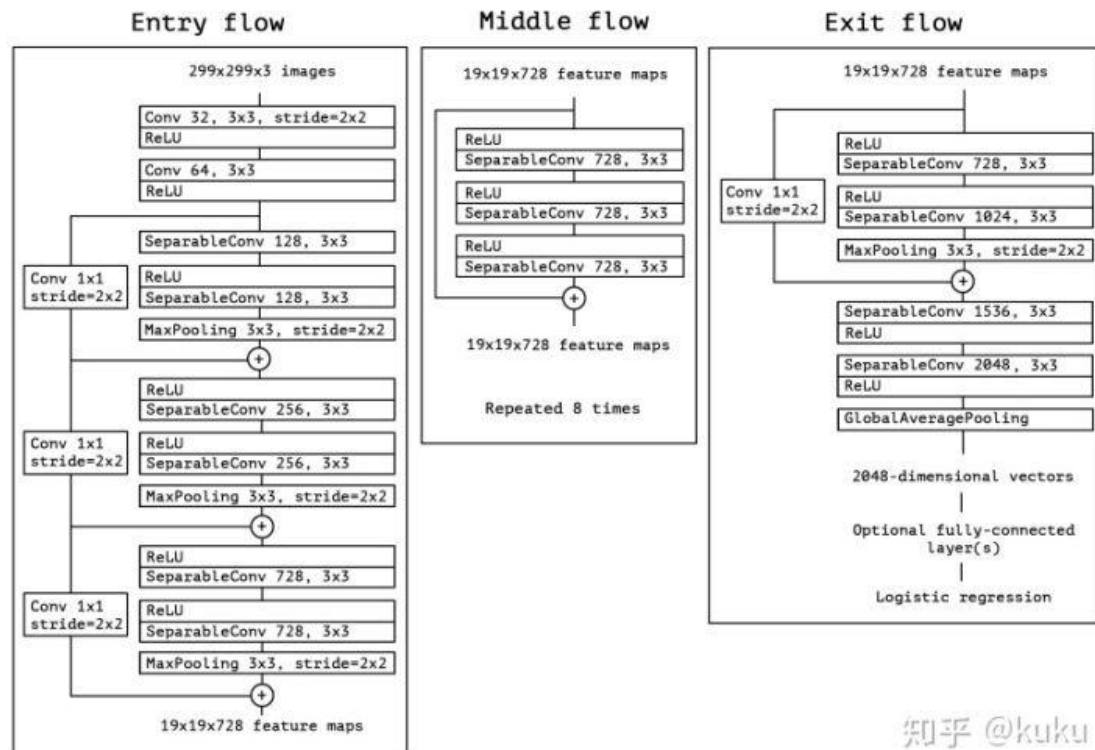
xception

Xception中尺寸变化时的block单元与中间的block操作单元



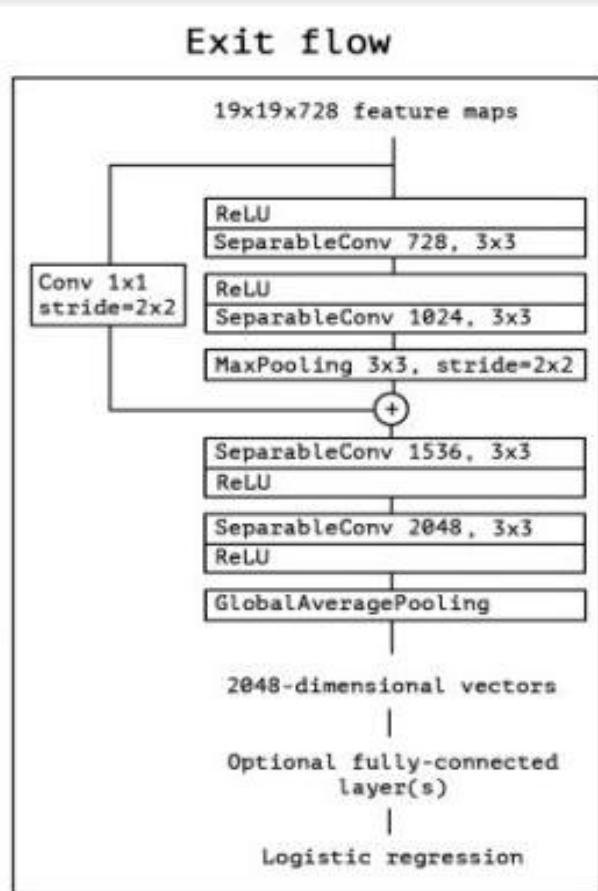
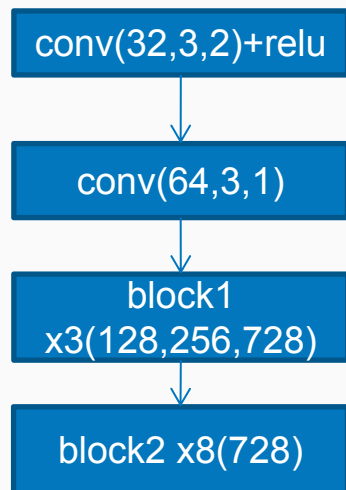
xception

整体结构



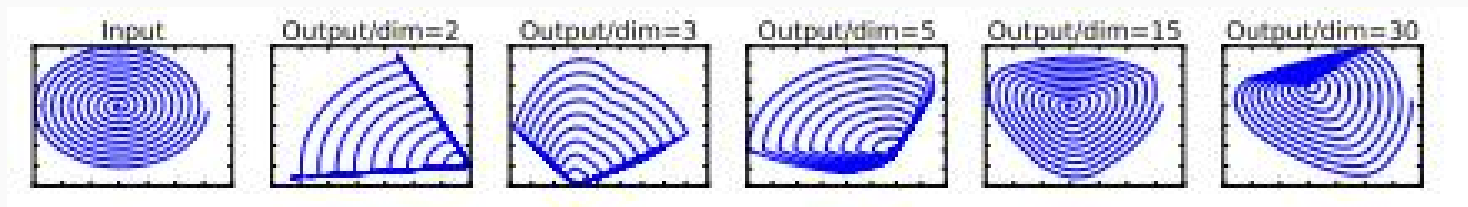
xception

整体结构



mobilenetv2: Inverted Residuals and Linear Bottlenecks

Linear Bottlenecks



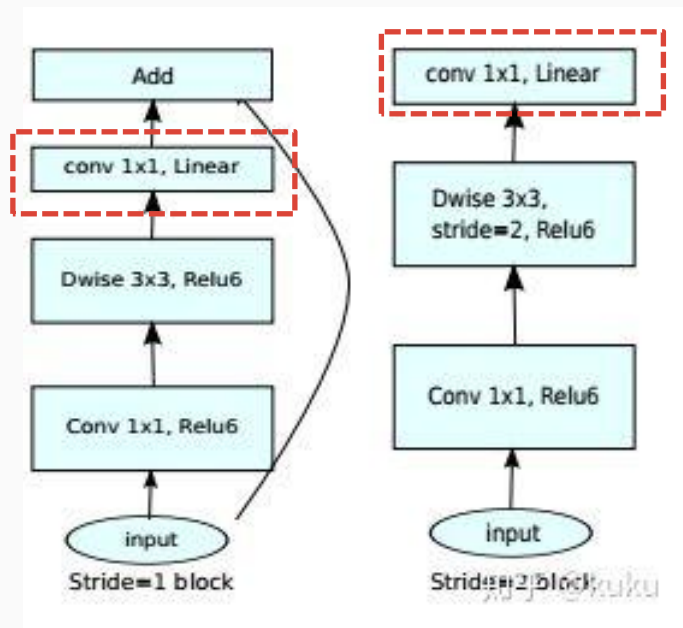
其上的操作是:对 x - y 空间运用随机生成的矩阵 T 进行变换, 然后再接 ReLU . 然后在运用 T^{-1} 反变换回来。观察信息的损失程度。



只有当感兴趣的流形信息是在高维空间的子空间里, ReLU 才能够保存所有信息。
反过来说, ReLU 在相对低维空间下造成的信息损失比高维空间下大得多。

mobilenetv2: Inverted Residuals and Linear Bottlenecks

Linear Bottlenecks

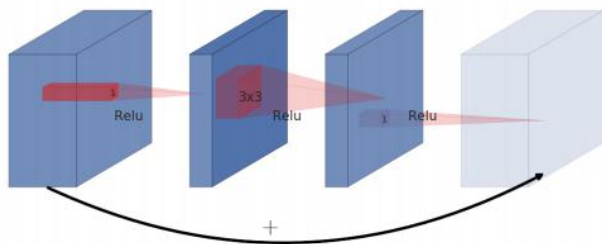


因此作者在通道变小后，没有使用relu6激活函数了。

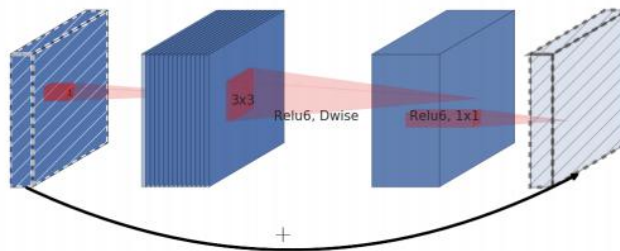
mobilenetv2: Inverted Residuals and Linear Bottlenecks

Inverted Residuals

(a) Residual block

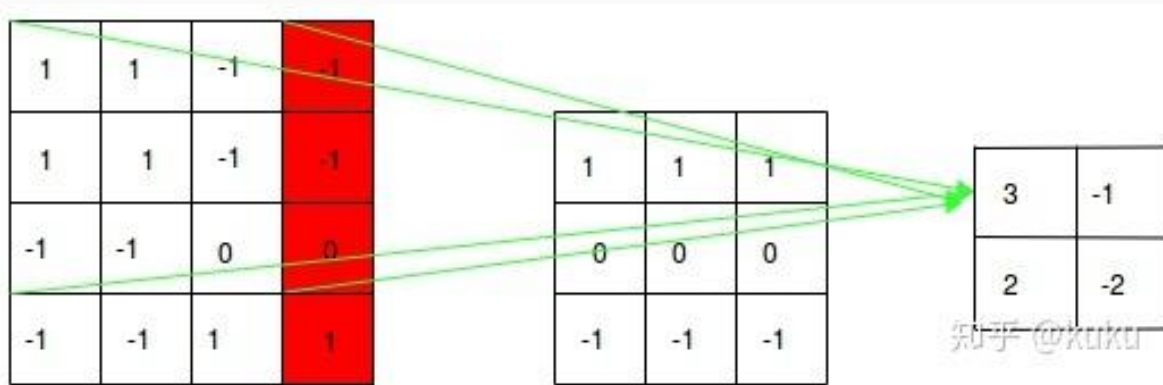


(b) Inverted residual block



mobilenetv2: Inverted Residuals and Linear Bottlenecks

Inverted Residuals



因为即便当前通道特征图为负，但是在其他通道中，依然有很大概率在相关处生成为正的特征图，从而对其进行优化。但是这里只有一个通道，因此作者首先将其放大(论文中放大6倍)来减少这样情况的发生。

mobilenetv2: Inverted Residuals and Linear Bottlenecks

整体框架

Input	Operator	t	c	n	s
$224^2 \times 3$	conv2d	-	32	1	2
$112^2 \times 32$	bottleneck	1	16	1	1
$112^2 \times 16$	bottleneck	6	24	2	2
$56^2 \times 24$	bottleneck	6	32	3	2
$28^2 \times 32$	bottleneck	6	64	4	2
$14^2 \times 64$	bottleneck	6	96	3	1
$14^2 \times 96$	bottleneck	6	160	3	2
$7^2 \times 160$	bottleneck	6	320	1	1
$7^2 \times 320$	conv2d 1x1	-	1280	1	1
$7^2 \times 1280$	avgpool 7x7	-	-	1	-
$1 \times 1 \times 1280$	conv2d 1x1	-	k	-	-

- 实现tensorflow的mobilenet
- 实现pytorch版本的xception
- 注意一个细节: depth-wise的conv的权重不加L2的约束。