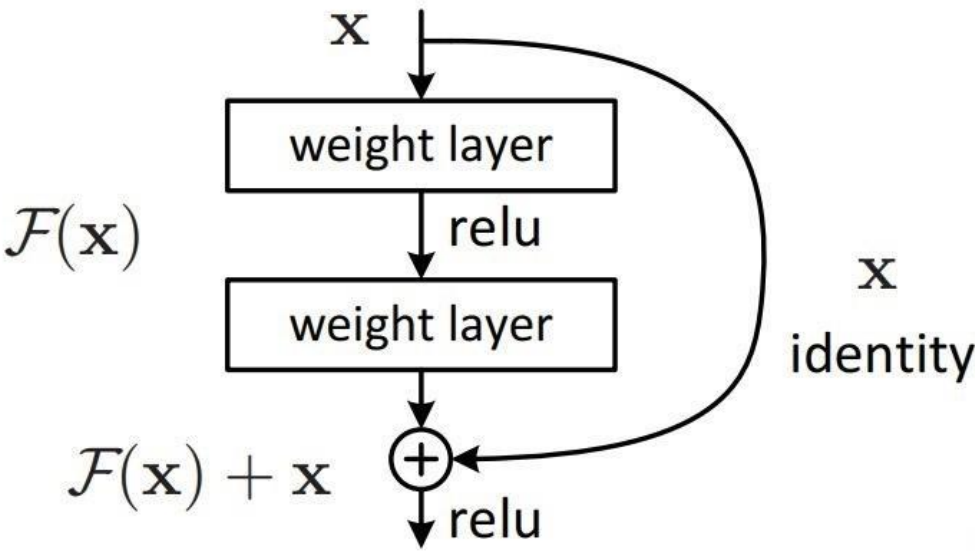
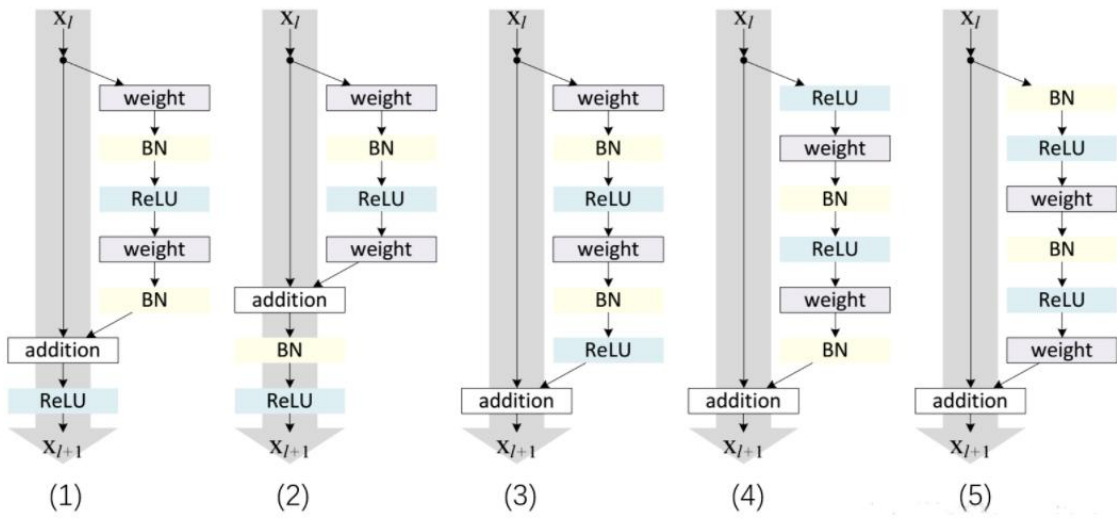


ResNet 残差块的结构:



ResNet 到 ResNetV2:



ResNet原文中使用的结构

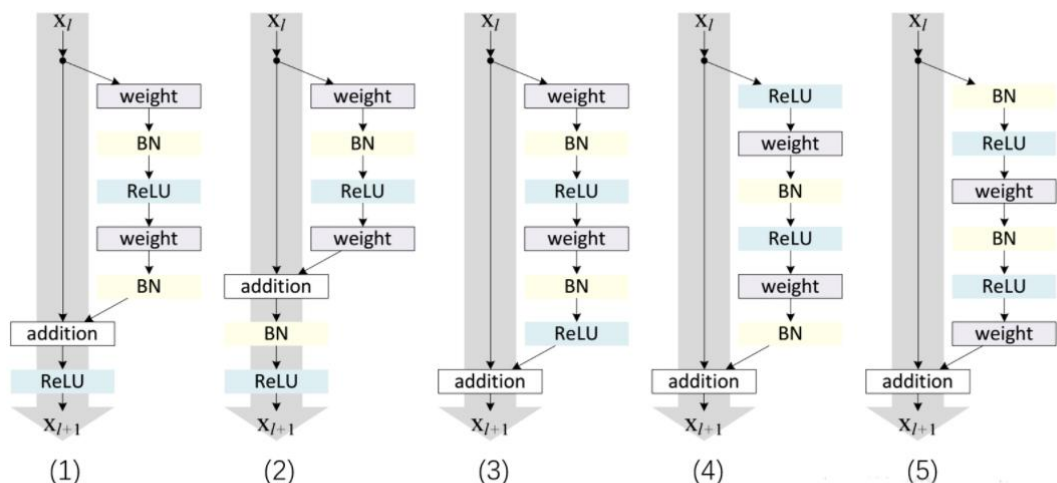
每个图右侧部分我们称作“residual”分支，左侧部分我们称作“identity”分支

图5的结构称作ResNetV2

weight指conv层，BN指Batch Normalization层，ReLU指激活层，addition指相加

case	Fig.	ResNet-110	ResNet-164
original Residual Unit [1]	Fig. 4(a)	6.61	5.93
BN after addition	Fig. 4(b)	8.17	6.50
ReLU before addition	Fig. 4(c)	7.84	6.14
ReLU-only pre-activation	Fig. 4(d)	6.71	5.91
full pre-activation	Fig. 4(e)	6.37	5.46

知乎 @zorechan



ResNet原文中使用的结构

每个图右侧部分我们称作“residual”分支，左侧部分我们称作“identity”分支
weight指conv层，BN指Batch Normalization层，ReLU指激活层，addition指相加

图5的结构称作ResNetV2

$$y_l = h(x_l) + F(x_l, w_l)$$

$$x_{l+1} = x_l + F(x_l, w_l)$$

$$x_L = x_l + \sum_{i=1}^{L-1} F(x_l, w_l)$$

反向传播求梯度：

$$\frac{dloss}{dx_l} = \frac{dloss}{dx_L} \frac{dx_L}{dx_l} = \frac{dloss}{dx_L} \left(1 + \frac{d}{dx_l} \sum_{i=1}^{L-1} F(x_l, w_l) \right)$$

$\frac{d}{dx_l} \sum_{i=1}^{L-1} F(x_l, w_l)$ 不会一直为-1，所以梯度消失的问题就被有效防止了。

这种优秀特点在假设 $h(x_l) = x_l$ 成立时有效。因此，ResNet 需要尽量保证残差部分不轻易改变。

pre-activation ResBlock 能够实现在网络的前向和反向传播中信息能由一个模块直接传递至任意其它模块而不影响传递（addition 不会破坏 x 的信息，因为求导对于加法而言是“分离的”），可以让网络更好的训练。

ResNet 的变种网络:

1. WRNS

一味的增加深度不是一个有效的方法，residual block 的宽度对网络的性能同样很有帮助。（在 ResNet 通道数上加宽了 k 倍）

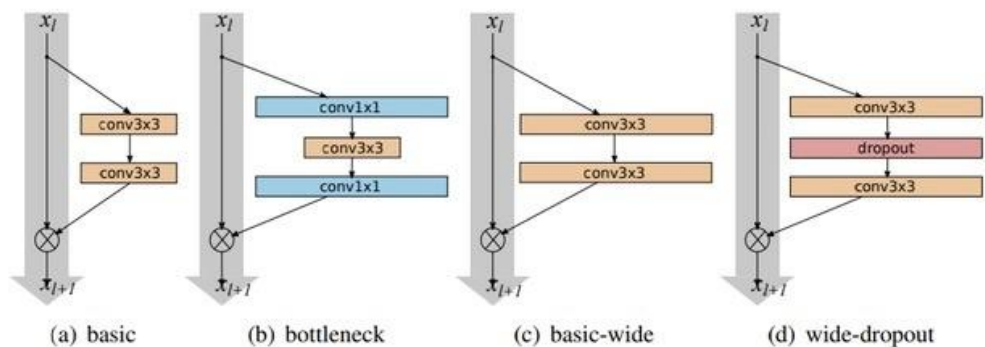


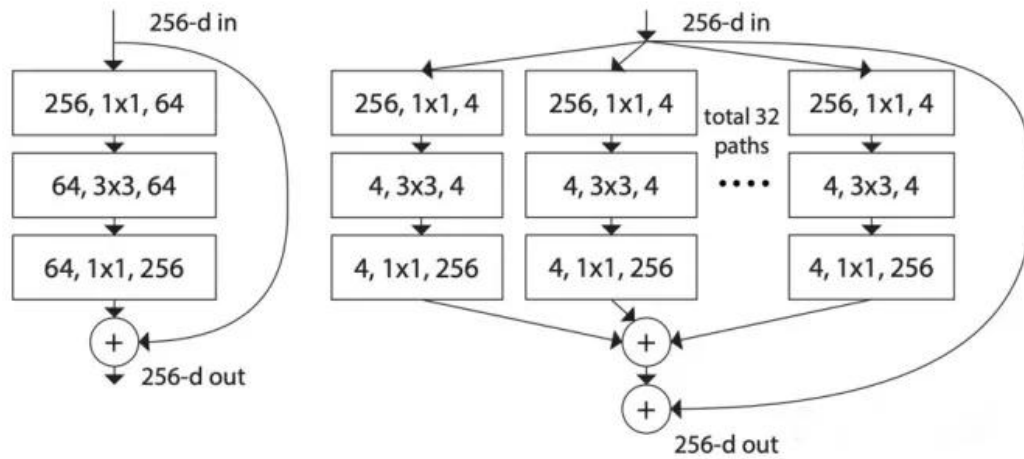
Figure 1: Various residual blocks used in the paper. Batch normalization and ReLU are used after each convolution (omitted for clarity)

group name	output size	block type = $B(3, 3)$
conv1	32×32	$[3 \times 3, 16]$
conv2	32×32	$\begin{bmatrix} 3 \times 3, 16 \times k \\ 3 \times 3, 16 \times k \end{bmatrix} \times N$
conv3	16×16	$\begin{bmatrix} 3 \times 3, 32 \times k \\ 3 \times 3, 32 \times k \end{bmatrix} \times N$
conv4	8×8	$\begin{bmatrix} 3 \times 3, 64 \times k \\ 3 \times 3, 64 \times k \end{bmatrix} \times N$
avg-pool	1×1	$[8 \times 8]$

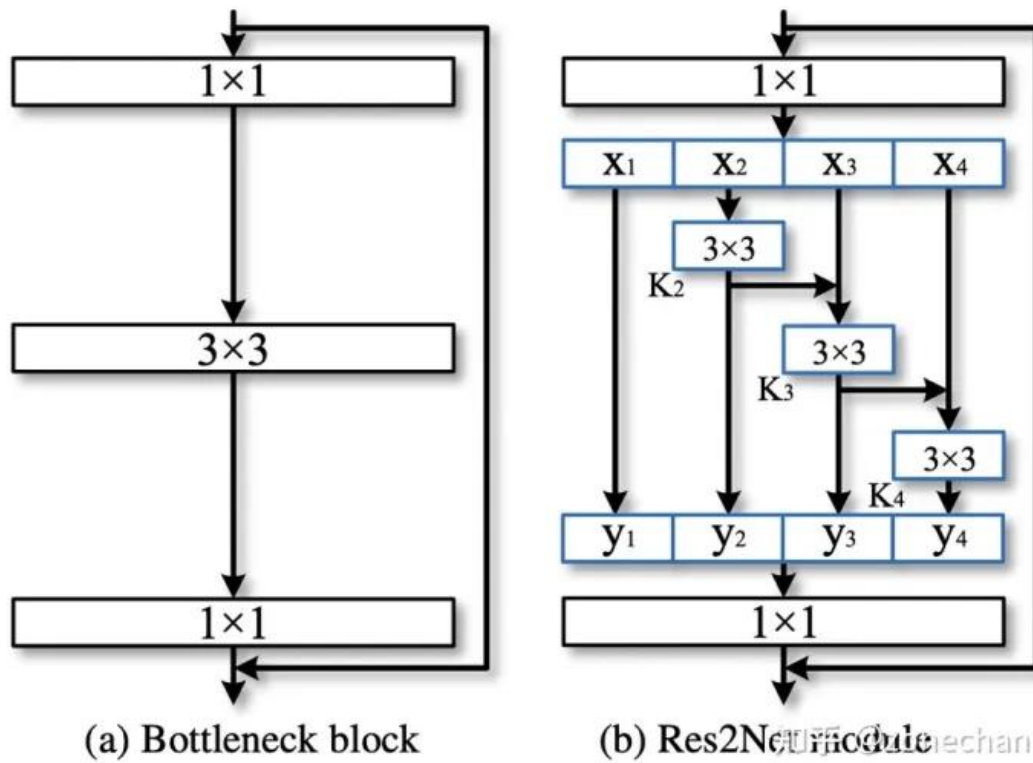
随着网络深度与宽度的加深，训练参数量过大会导致过拟合，所以在 residual block 里面加入了 dropout。

2. ResNext

打破了增加深度或是宽度的常规思路。

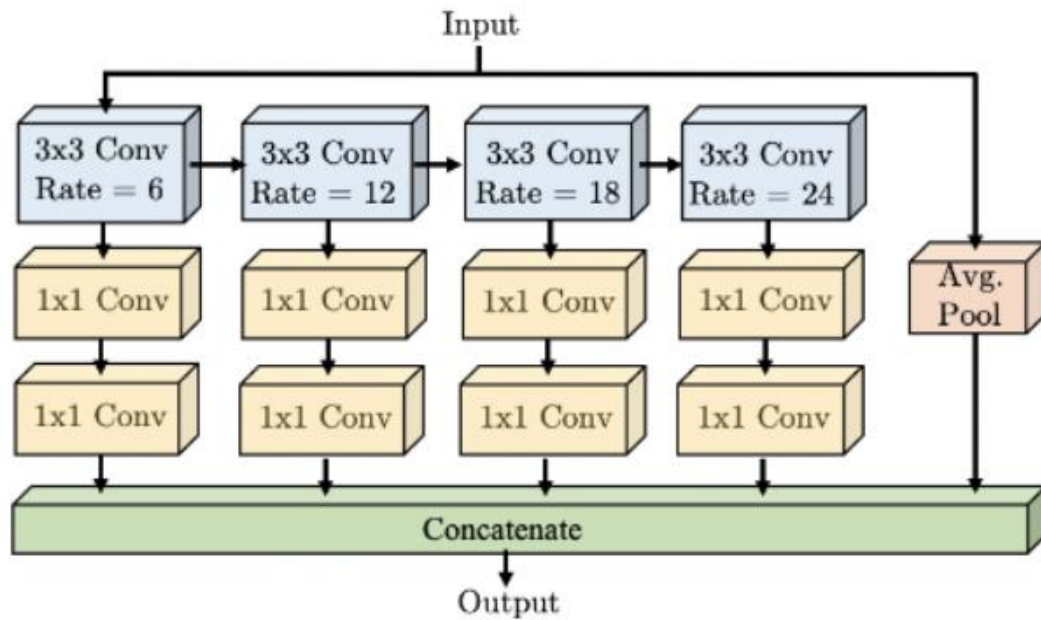


3. Res2Net



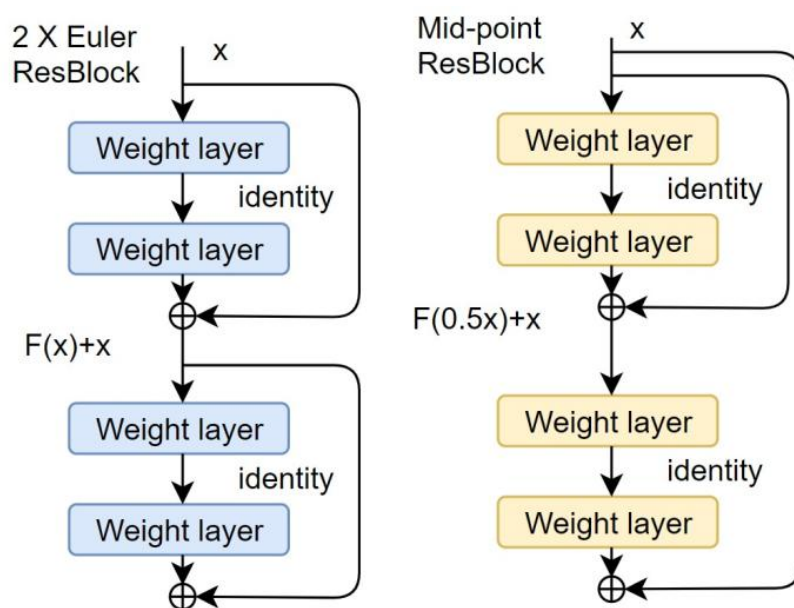
4. Unipose

魔改 Res2Net



和 Res2Net 相比，WASP 模块不会立即并行化输入流。它首先通过卷积过滤器进行处理，然后创建一个新分支，从而创建瀑布流。

5. HO-ResNet



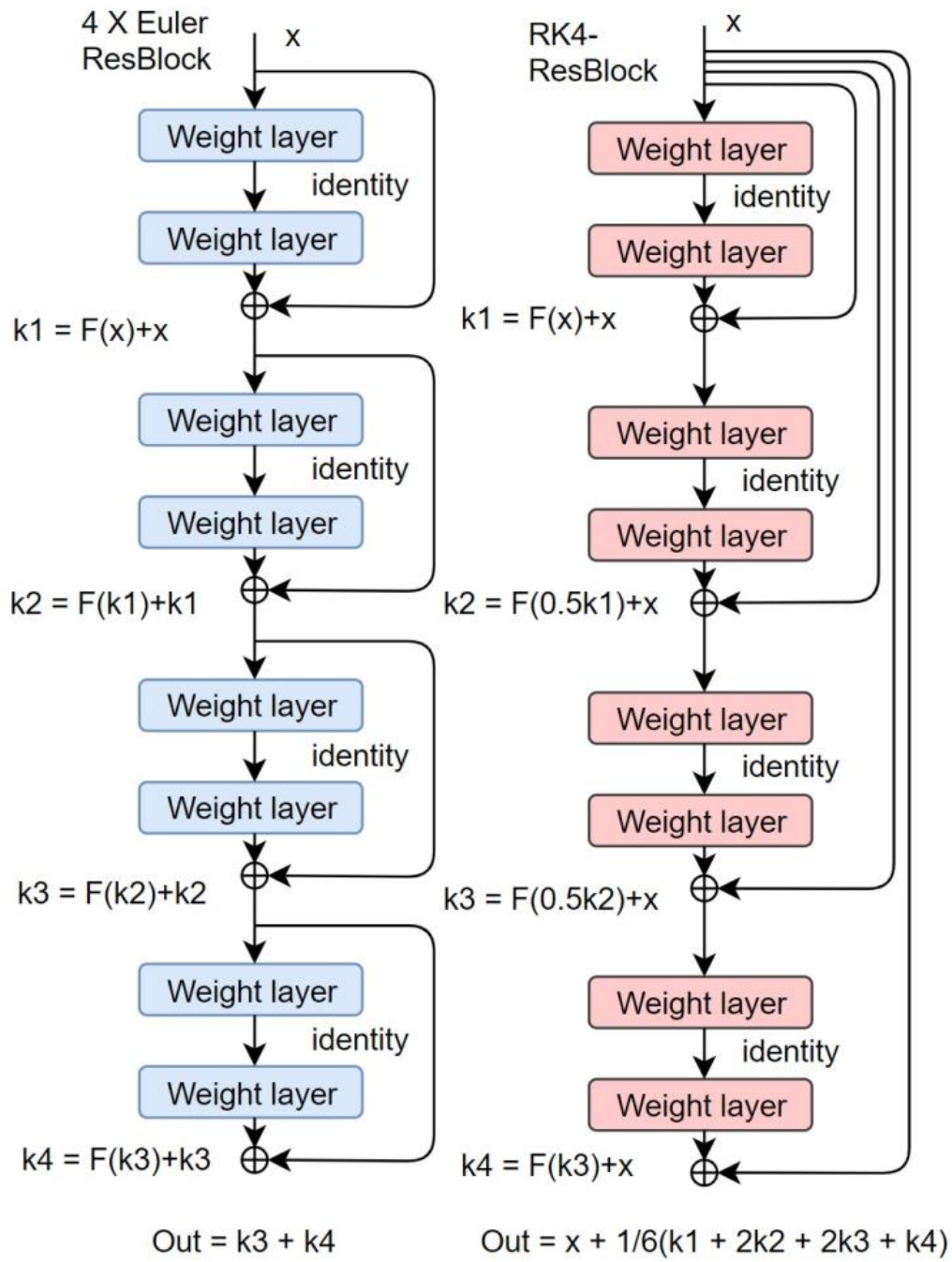


Figure 4: **RK4 ResBlock**: Comparing four stacked ResBlock-Euler with a single design which following the 4th-order Runge-Kutta method.