1. $1 * 1$ Convolution

$1 * 1$ convolution was used to reduce the number of channels while introducing non-linearity. In $1 * 1$ Convolution simply means the filter is of size $1 * 1$. This $1 * 1$ filter will convolve over the ENTIRE input image pixel by pixel.
Considering an input of $64 * 64 * 3$, if we choose a $1 * 1$ filter (which would be $1 * 1 * 3$), then the output will have the same Height and Weight as input but only one channel — $64 * 64 * 1$ Now consider inputs with large number of channels — 192 for example in Figure 1. If we want to reduce the depth and but keep the Height$*$Width of the feature maps (Receptive field) the same, then we can choose $1 * 1 * 3$ filters (remember Number of filters = Output Channels) to achieve this effect. This effect of cross channel down-sampling is called 'Dimensionality reduction'.

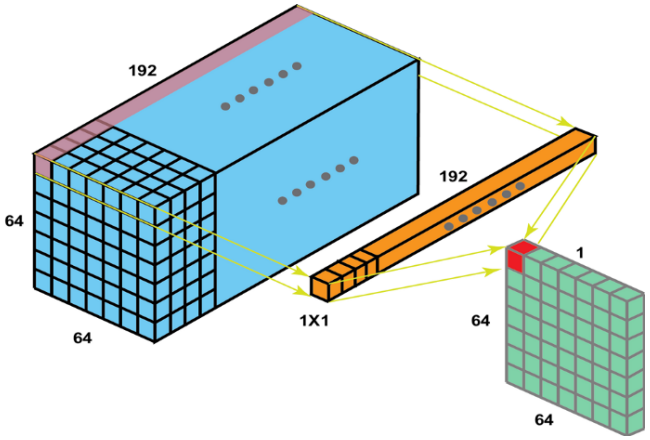

Figure 1. Reducing the number of channels by $1 * 1$ convolution.

$1 * 1$ Convolution is effectively used for

- Dimensionality Reduction/Augmentation;
- Reduce computational load by reducing parameter map;
- Add additional non-linearity to the network;
- Create deeper network through "Bottle-Neck" layer (See ResNet);
- Create smaller CNN network which retains higher degree of accuracy (See SqueezeNet).

2. Group Convolution

In group convolution, the input feature map is grouped, and each group is convolved separately, as shown in Figure 2. Suppose the size of the input feature map is $C * H * W$, and the number of output feature maps is N. If it is set to be divided into G groups, the number of input feature maps for each group is C/G, and the output features of each group The number of maps is N/G, the size of each convolution kernel is $C/G * K * K$, the total number of convolution kernels is still N, the number of convolution kernels in each group is N/G, and the convolution kernel is only The input maps of the same group are convolved. The total parameter of the convolution kernel is $N * C/G * K * K$, which can be seen,The total number of parameters is reduced to the original $1/G$.
Group Convolution can be seen as structured sparse, The size of each convolution kernel is changed from $C * K * K$ to $C/G * K * K$, the parameters of the remaining $(CC/G) * K * K$ can be regarded as 0, and sometimes the parameters can even be reduced at the same time to obtain better results.

When the number of groups is equal to the number of input maps, the number of output maps is also equal to the number of input maps, that is, $G = N = C$, N convolution kernels each size is $1 * K * K$, Group Convolution becomes Depthwise Convolution (see MobileNet and Xception).

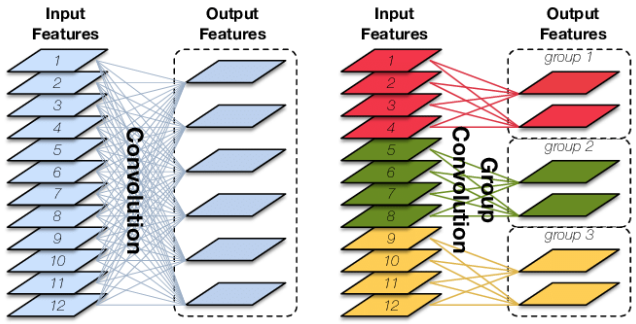

FIGURE 2. Standard convolution vs. group convolution.