Deep Learning
Mobilenetv2 and MBConv

April 28, 2023

---

1 MobileNetV2

1.1 Inverted residual and linear bottleneck layer

The MobileNetV2 network is build around inverted residual layers. The input for the layer is a low-dimensional tensor and it consists of three seperate convolutions. First, a point-wise convolution with a ReLU6 activation function. Next a depthwise convolution using 3*3 kernels also followed by ReLU6. Finally another pointwise convolution is applied to project the spatially-filtered feature map back to a low-dimensional tensor. The dimensions of the input tensor and output tensor a equal. To allow gradient flow during backpropagation a residual connection is added (the bend arrow in figure 1).

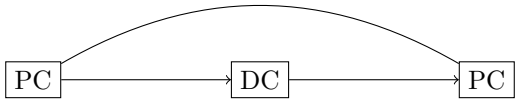

Figure 1: Block diagram to represent the Inverted residual block.

1.2 Channel Attention and Squeeze-and-Excitation Networks (SENet)

In a convolution operation filters are applied to the input channels of an image. Each filter is responsible for different feature maps. For example, one filter might learn edges another might learn textures. Each filter has a different level of importance for a specific application. This is where Channel Attention comes in. The filters are given weights to indicate the importance of a specific filter.

In the model, Channel Attention is combined with the Squeeze-and-Excitation block. The first component in the Squeeze-and-Excitation block, is the Squeeze module. The feature map set is essentially the output tensors of a CNN and to scale the feature maps, according to the Channel Attention method, you would have to scale all the values in the output tensors. This would dramatically increase the amount of parameters and that is where the squeeze module comes in. To reduce the spatial size in CNN's the general method is pooling. This is exactly what the Squeeze module entails, Global Average Pool, it takes the average of the whole output feature. This results in a dimension reduction from $C \times H \times W$ to $C \times 1 \times 1$. This dramatically reduces the number of parameters in Channel Attention.

Next the excitation module, now that the amount of scaling weights is hugely reduced we need a method to adapt the weights for these channels. Multi-Layer Perceptron (MLP) bottleneck structure is used to map the scaling weights. The MLP has one hidden layer in combination with an input and output layer. First the input is compressed to the hidden layer and than expanded back to the original dimensionality. See figure 2 for a schematic overview where r is the reduction factor.

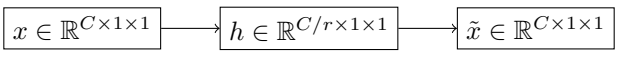

Figure 2: Block diagram to represent the excitation module.

Ideally the reduction factor is set to 1 for improved information propagation and better cross-channel interaction. However, this will be a trade-off between increasing complexity and performance improvement.

Finally the layer consists of a Scale Module. First the "excited" tensor is passed through a sigmoid activation so the values fall within the range of 0-1. Next the feature maps are finally scaled by a element-wise multiplication.

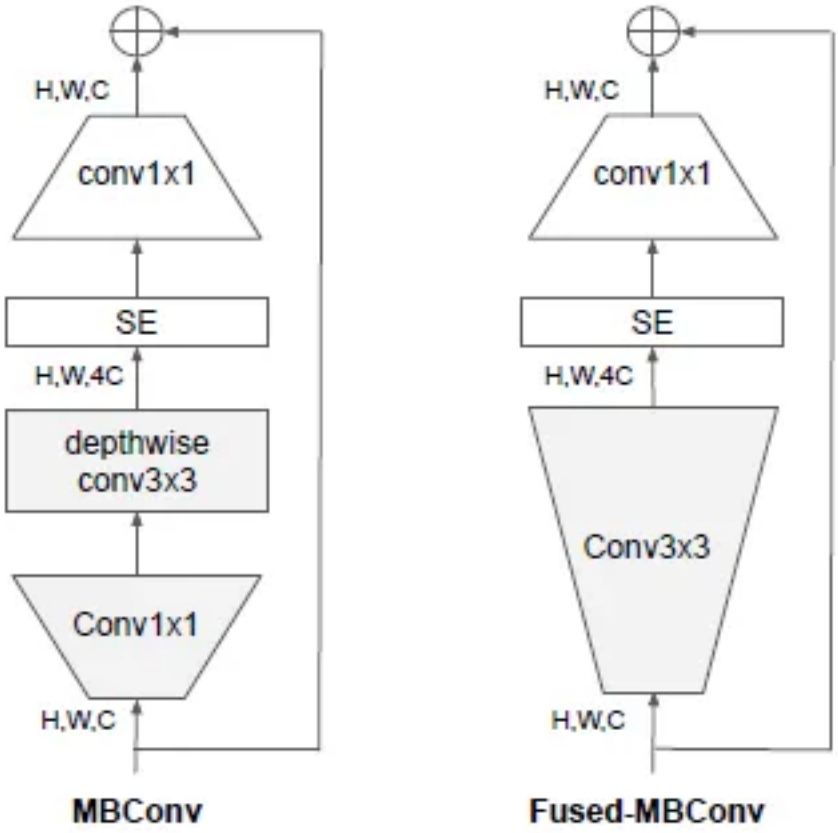

Figure 3: MBConv and Fused-MBConv

2 EfficientNet

2.1 MBConv Fused-MBConv

In the previous section we explained how inverted residual layers and Squeeze-and-Excitation layers are constructed. In this section we dive into how MBConv and Fused-MBConv are build from these layers.

First MBConv, the MBConv combines the inverted residual blocks with the SEnet block. The SEnet block is added in the inverted residual blocks between the depthwise convolution and the last point-wise convolution.

Fused-MBConv works similar to MBConv but the first point-wise convolution and the depth-wise 3x3 convolution are combined to a singe 3x3 convolution. This results in a faster model as one less operation results in less learnable parameters. Both constructions are schematically represented below in figure 3.