TensorFlow   API r1.4

tf.nn.depthwise_conv2d

```
depthwise_conv2d(
    input,
    filter,
    strides,
    padding,
    rate=None,
    name=None,
    data_format=None
)
```

Defined in `tensorflow/python/ops/nn_impl.py` .

See the guide: Neural Network > Convolution

Depthwise 2-D convolution.

Given a 4D input tensor ('NHWC' or 'NCHW' data formats) and a filter tensor of shape `[filter_height, filter_width, in_channels, channel_multiplier]` containing `in_channels` convolutional filters of depth 1, `depthwise_conv2d` applies a different filter to each input channel (expanding from 1 channel to `channel_multiplier` channels for each), then concatenates the results together. The output has `in_channels * channel_multiplier` channels.

In detail,

```
output[b, i, j, k * channel_multiplier + q] = sum_{di, dj}
    filter[di, dj, k, q] * input[b, strides[1] * i + rate[0] * di,
                                    strides[2] * j + rate[1] * dj, k]
```

Must have `strides[0] = strides[3] = 1` . For the most common case of the same horizontal and vertical strides, `strides = [1, stride, stride, 1]` . If any value in `rate` is greater than 1, we perform atrous depthwise convolution, in which case all values in the `strides` tensor must be equal to 1.

Args:

- `input` : 4-D with shape according to `data_format` .
- `filter` : 4-D with shape `[filter_height, filter_width, in_channels, channel_multiplier]` .
- `strides` : 1-D of size 4. The stride of the sliding window for each dimension of `input` .
- `padding` : A string, either `'VALID'` or `'SAME'` . The padding algorithm. See the comment here
- `rate` : 1-D of size 2. The dilation rate in which we sample input values across the `height` and `width` dimensions in atrous convolution. If it is greater than 1, then all values of strides must be 1.
- `name` : A name for this operation (optional).
- `data_format` : The data format for input. Either "NHWC" (default) or "NCHW".

Returns:

A 4-D `Tensor` with shape according to `data_format` . E.g., for "NHWC" format, shape is `[batch, out_height, out_width, in_channels * channel_multiplier].`

Stay Connected

Blog

GitHub

Twitter

Support

Issue Tracker

Release Notes

Stack Overflow

English