

# Tensorflow中CNN的相关函数



## 卷积函数



浙江大学城市学院  
ZHEJIANG UNIVERSITY CITY COLLEGE

卷积函数定义在tensorflow/python/ops下的nn\_impl.py和nn\_ops.py文件中:

- `tf.nn.conv2d`(input, filter, strides, padding, use\_cudnn\_on\_gpu=None, name=None)
- `tf.nn.depthwise_conv2d`(input, filter, strides, padding, name=None)
- `tf.nn.separable_conv2d`(input, depthwise\_filter, pointwise\_filter, strides, padding, name=None)
- 等等



## 卷积函数



浙江大学城市学院  
ZHEJIANG UNIVERSITY CITY COLLEGE

**tf.nn.conv2d(input, filter, strides, padding, use\_cudnn\_on\_gpu=None, name=None)**

- **input**: 需要做卷积的输入数据。注意：这是一个4维的张量 ([batch, in\_height, in\_width, in\_channels])，要求类型为float32或float64其中之一。
- **filter**: 卷积核。[filter\_height, filter\_width, in\_channels, out\_channels]
- **strides**: 图像每一维的步长，是一个一维向量，长度为4
- **padding**: 定义元素边框与元素内容之间的空间。"SAME"或"VALID"，这个值决定了不同的卷积方式。当为"SAME"时，表示边缘填充，适用于全尺寸操作；当为"VALID"时，表示边缘不填充。
- **use\_cudnn\_on\_gpu**: bool类型，是否使用cudnn加速
- **name**: 该操作的名称
- **返回值**: 返回一个tensor，即feature map



## 卷积函数

### 使用示例

```
import tensorflow as tf
import numpy as np

# tf.nn.conv2d(input, filter, strides, padding, use_cudnn_on_gpu=None, name=None)
input_data = tf.Variable(np.random.rand(10, 9, 9, 4), dtype=np.float32)
filter_data = tf.Variable(np.random.rand(3, 3, 4, 2), dtype=np.float32)
y = tf.nn.conv2d(input_data, filter_data, strides=[1,1,1,1], padding = 'SAME')
#y = tf.nn.conv2d(input_data, filter_data, strides=[1,1,1,1], padding = 'VALID')

print(input_data)
print(y)
```

```
Tensor("Variable_2/read:0", shape=(10, 9, 9, 4), dtype=float32)
Tensor("Conv2D_1:0", shape=(10, 9, 9, 2), dtype=float32)
```





## 池化函数

池化函数定义在tensorflow/python/ops下的nn.py和gen\_nn\_ops.py文件中:

- 最大池化: `tf.nn.max_pool(value, ksize, strides, padding, name=None)`
- 平均池化: `tf.nn.avg_pool(value, ksize, strides, padding, name=None)`
- 等等



## 池化函数



浙江大学城市学院  
ZHEJIANG UNIVERSITY CITY COLLEGE

**最大池化:** `tf.nn.max_pool(value, ksize, strides, padding, name=None)`

**平均池化:** `tf.nn.avg_pool(value, ksize, strides, padding, name=None)`

- **value**: 需要池化的输入。一般池化层接在卷积层后面，所以输入通常是conv2d所输出的feature map，依然是4维的张量 ([batch, height, width, channels]) 。
- **ksize**: 池化窗口的大小，由于一般不在batch和channel上做池化，所以ksize一般是[1,height, width,1],
- **strides**: 图像每一维的步长，是一个一维向量，长度为4
- **padding**: 和卷积函数中padding含义一样
- **name**: 该操作的名称
- **返回值**: 返回一个tensor



## 池化函数



浙江大学城市学院  
ZHEJIANG UNIVERSITY CITY COLLEGE

```
import tensorflow as tf
import numpy as np

input_data = tf.Variable(np.random.rand(10, 6, 6, 4), dtype=np.float32)
filter_data = tf.Variable(np.random.rand(2, 2, 4, 2), dtype=np.float32)
y = tf.nn.conv2d(input_data, filter_data, strides=[1,1,1,1], padding = 'SAME')

# 最大池化
# tf.nn.max_pool(value, ksize, strides, padding, name=None)
# output = tf.nn.max_pool(value=y, ksize=[1,2,2,1], strides=[1,2,2,1], padding='SAME')

# 平均池化
# tf.nn.avg_pool(value, ksize, strides, padding, name=None)
output = tf.nn.avg_pool(value=y, ksize=[1,2,2,1], strides=[1,2,2,1], padding='SAME')

print('conv:', y)
print('pool_padding_valid:', output)
```

conv: Tensor("Conv2D\_3:0", shape=(10, 6, 6, 2), dtype=float32)  
pool\_padding\_valid: Tensor("AvgPool:0", shape=(10, 3, 3, 2), dtype=float32)

计算维度输出： $\text{shape}(\text{output}) = (\text{shape}(\text{value}) - \text{ksize} + 1) / \text{strides}$