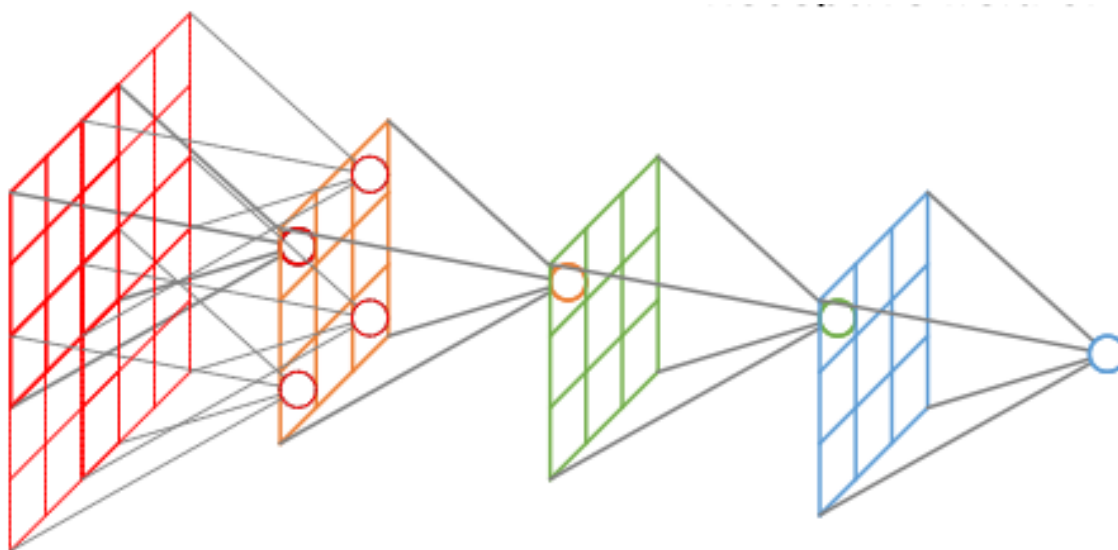


## 无痛理解CNN中的感受野receptive fieldCNN中感受野的计算

从直观上讲，感受野就是视觉感受区域的大小。在卷积神经网络中，感受野的定义是决定某一层输出结果中一个元素所对应的输入层的区域大小



感受野计算时有下面的几个情况需要说明：

- a) 第一层卷积层的输出特征图像素的感受野的大小等于滤波器的大小；
- b) 深层卷积层的感受野大小和它之前所有层的滤波器大小和步长有关系；
- c) 计算感受野大小时，忽略了图像边缘的影响，即不考虑padding的大小。

至于如何计算感受野，我的建议是top to down的方式，即从网络的最后向前推

感受野的大小是由kernel size (filter) 和stride size (步长) 一起决定的，公式：

$$rfs_{size} = f(out, stride, ksize) = (out - 1) * stride + ksize$$
，其中out是指上一层感受野的大小，stride是当前层stride

最后一层不带入公式，它的ksize是前一层的out

**理解：**

Feature Map的尺寸
$$= (input\_size + 2 * padding\_size - ksize) / stride + 1$$

根据定义 感受野是决定某一层输出结果中一个元素所对应的输入层的区域大小

这里指的是要求解的那层的一个元素也就是最初输入的out=1:

$$rsize = f(out, stride, ksize) = (out - 1) * stride + ksize$$

感受野近似于用feature map反推input\_size ，只是不考虑padding

例1:

以VGG16为例:

vgg16构造模型图:

<http://ethereon.github.io/netscope/#/gist/dc5003de6943ea5a6b8b>

一个元素，所以out=1

$$\text{pool2: } rsize = (1-1)*2+2=2$$

$$\text{conv1\_2: } rsize = (2-1)*1+3=4$$

$$\text{conv1\_1: } rsize = (4-1)*1+3=6$$

例2:

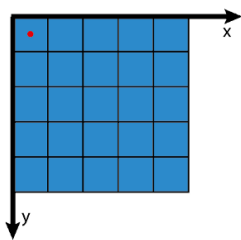
r指感受野 s: stride, p: padding, k: filter

$n$  : number of features  
 $r$  : receptive field size  
 $j$  : jump (distance between two consecutive features)  
 $start$  : center coordinate of the first feature

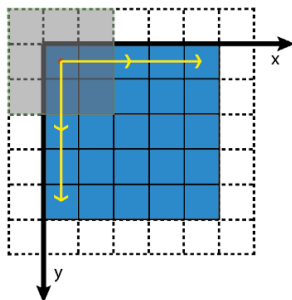
$k$  : convolution kernel size  
 $p$  : convolution padding size  
 $s$  : convolution stride size

$$\begin{aligned}
 n_{out} &= \left\lfloor \frac{n_{in} + 2p - k}{s} \right\rfloor + 1 \\
 j_{out} &= j_{in} * s \\
 r_{out} &= r_{in} + (k - 1) * j_{in} \\
 start_{out} &= start_{in} + \left( \frac{k - 1}{2} - p \right) * j_{in}
 \end{aligned}$$

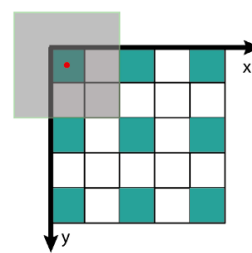
Layer 0:  $n_0 = 5$ ;  $r_0 = 1$ ;  $j_0 = 1$ ;  
 $start_0 = 0.5$



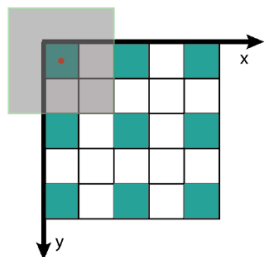
Conv1:  $k_1 = 3$ ;  $p_1 = 1$ ;  $s_1 = 2$



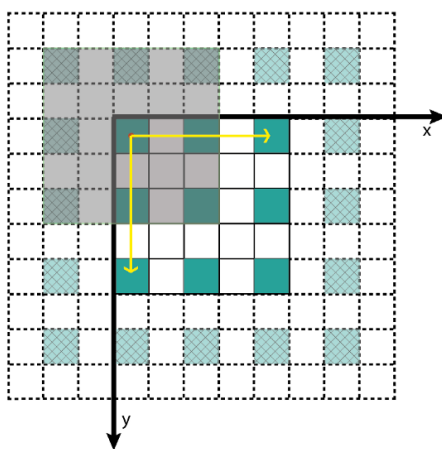
Layer 1:  $n_1 = 3$ ;  $r_1 = 3$ ;  $j_1 = 2$ ;  
 $start_1 = 0.5$



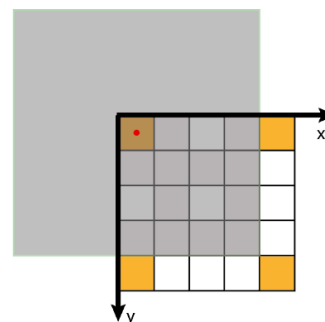
Layer 1:  $n_1 = 3$ ;  $r_1 = 3$ ;  $j_1 = 2$ ;  
 $start_1 = 0.5$



Conv2:  $k_2 = 3$ ;  $p_2 = 1$ ;  $s_2 = 2$



Layer 2:  $n_3 = 2$ ;  $r_3 = 7$ ;  $j_3 = 4$ ;  
 $start_1 = 0.5$



layer1对layer0:

out=1

$$(1-1)*2+3=3$$

layer2对layer0:

out=1

$$(1-1)*2+3=3$$

$$(3-1)*2+3=7$$