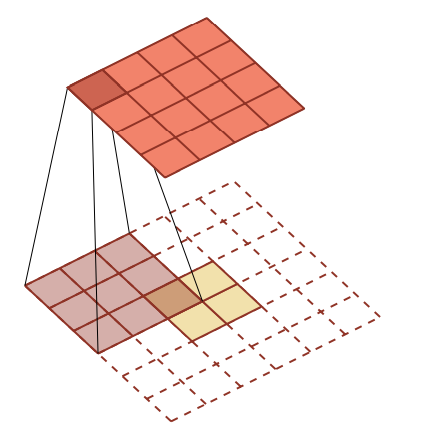
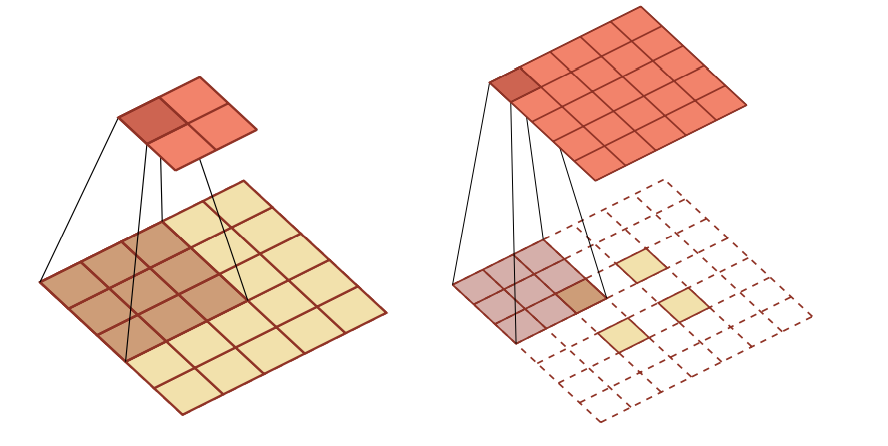
**反卷积**

1.步长S=1

 S=1,K=3,P=2

对一个维的向量，和大小为的卷积核，如果希望通过卷积操作来映射到更高维的向量，可要对向量两端圈0，然后进行卷积，输出向量尺寸：𝑀 + 𝐾 − 1

2.步长S<1

 S=2,K=3,P=2

不仅对向量两端圈0，并且在维的向量每两个元素之间插入个0，再对图像做步长为1的卷积，输出向量尺寸：

**nn.ConvTranspose2d**

**nn.ConvTranspose2d(in\_channels,out\_channels,kernel\_size,stride=1, bias=True)**

**参数解释**

in\_channels，out\_channels：转置卷积输入输出特征图的通道数

kernel\_size：转置卷积卷积核尺寸

stride：转置卷积步长

设deconv是nn.ConvTranspose的实例，则deconv(input,hidden)自动进行前向计算

deconv = nn.ConvTranspose2d(1,128,5,2,bias=False)

img = deconv(train\_dataset[100][0].view(1,1,28,28))

img.size()

>>>

torch.Size([1, 128, 59, 59])