Chapter 1

Upsampling Layer ("Transposed Convolution")

Learnable Upsampling

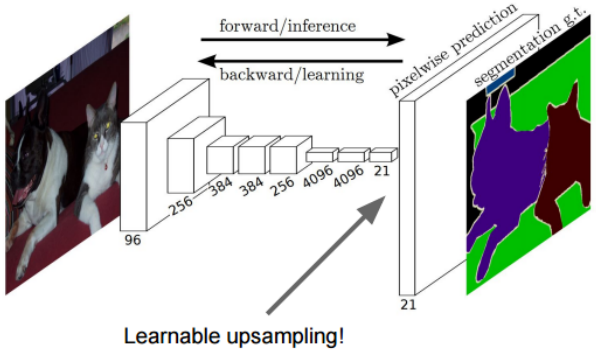

FIGURE 1.1: From the paper: "Fully Convolutional Networks for Semantic Segmentation"

Transposed Convolution

Transposed convolutions – also called fractionally strided convolutions – work by swapping the forward and backward passes of a convolution. One way to put it is to note that the kernel defines a convolution, but whether its a direct convolution or a transposed convolution is determined by how the forward and backward passes are computed.

The transposed convolution is implemented as the backward pass of a corresponding non-transposed convolution. It can be thought of as dilating the input (by adding "stride - 1" zeros between adjacent input elements), padding it with the needed number of zeros so it is not out. And then, apply the convolution with the filter flipped 180 degrees.

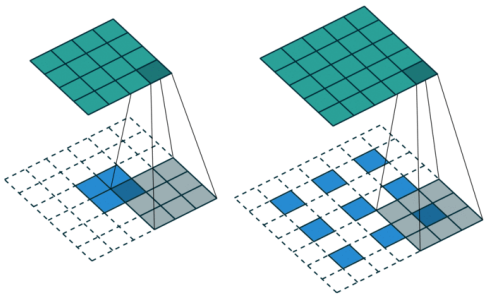

FIGURE 1.2: **Left**: Visually, for a transposed convolution with stride one and no padding, we just pad the original input (blue entries) with zeroes (white entries). **Right**: Stride two and padding, the transposed convolution would look like this

Skip Connections

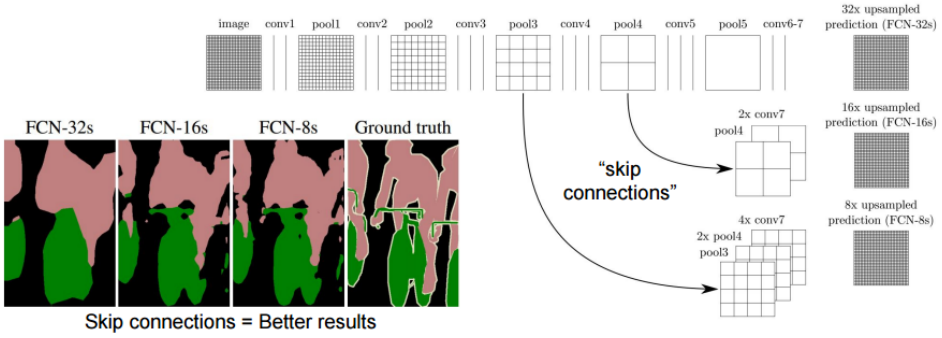

FIGURE 1.3: Skip Connections