Slide 1: Right Network Portion Overview

- **Feature Expansion for Segmentation**:
    - The right side of the network expands spatial support of lower-resolution feature maps to assemble necessary information for two-channel volumetric segmentation.
    - *"The right portion of the network extracts features and expands the spatial support of the lower resolution feature maps in order to gather and assemble the necessary information to output a two channel volumetric segmentation."*
- **Final Layer & Probabilistic Segmentation**:
    - Uses a 1×1×11 \times 1 \times 11×1×1 kernel size in the last convolutional layer.
    - Converts output into probabilistic segmentations (foreground and background) using softmax.
    - *"The two features maps computed by the very last convolutional layer, having 1×1×1 kernel size and producing outputs of the same size as the input volume, are converted to probabilistic segmentations of the foreground and background regions by applying soft-max voxelwise."*

Hi I'm Ze, Li,

Let's start by looking at the role of the right portion of the network in this architecture. This part of the network is responsible for expanding the spatial support of lower-resolution feature maps to accumulate necessary information for the final output. Specifically, it helps create a two-channel volumetric segmentation that distinguishes between foreground and background regions.

In the final layer, we use a very small 1×1×1kernel to maintain the same size as the input volume. The outputs from this layer are converted to probabilistic segmentations by applying a softmax function voxelwise, which gives us probabilities of foreground and background at each voxel. This setup helps ensure that we have a detailed and accurate segmentation at the output.

Slide 2: Operations in the Right Network Pathway

- **De-convolution Operation for Spatial Expansion**:
    - De-convolutions increase the input size, as shown in Figure 3, by projecting each voxel to a larger spatial area.
    - Each de-convolution is followed by one to three convolutional layers with half the number of 5×5×5 kernels from the previous layer.
    - *"After each stage of the right portion of the CNN, a de-convolution operation is employed in order increase the size of the inputs (Figure 3) followed by one to three convolutional layers involving half the number of 5 × 5 × 5 kernels employed in the previous layer."*

- **Residual Learning in Convolutional Stages**:
    - Similar to the left part of the network, residual functions are learned in the convolutional stages to stabilize training.
    - *"Similar to the left part of the network, also in this case we resort to learn residual functions in the convolutional stages."*

Now, we can focus on the operations within the right pathway of the network. To progressively expand the spatial dimensions, each stage in the right portion of the network applies a de-convolution operation. As shown in Figure 3, 2×2×2 de-convolutions with stride 2 project each input voxel to a larger spatial region, effectively increasing the resolution of the feature maps. This step helps in upsampling the lower-resolution feature maps to restore the spatial resolution. Following each de-convolution, we add one to three convolutional layers, and each of these uses half the number of 5×5×5 kernels compared to the previous layer. This gradual reduction setup allows us to gradually expand and control the complexity of the model while still capturing detailed features at each upsampling stage.

Additionally, we use residual connections here as well. Similar to the left portion of the network, these residual functions are learned in the convolutional stages, which helps stabilize training by keeping the gradient flow consistent throughout the network.

1. **De-Convolution (Upsampling) Operation**:
    - As shown in the left part of Figure 3, the 2×2×2 de-convolution with stride 2 increases the spatial dimensions of the input by projecting each voxel into a larger region. This step helps in upsampling the lower-resolution feature maps to restore the spatial resolution.
2. **Following Convolutional Layers with 5×5×5 Kernels**:
    - After upsampling through de-convolution, the network applies one to three regular convolutional layers. These convolutional layers use 5×5×5 kernels to further refine the upsampled feature maps.
    - Each successive convolution layer in this sequence has half the number of 5×5×5 kernels compared to the previous layer. This gradual reduction in the number of kernels helps control the complexity of the model while still capturing detailed features at each upsampling stage.

So, while the de-convolution operation itself does not use 5×5×55 \times 5 \times 55×5×5 kernels, these kernels are employed in the subsequent convolutional layers to further process and refine the expanded feature maps.

Slide 3: Detail Preservation & Global Context

- **Skip Connections for Fine-grained Detail**:

    - Early stage features from the left part of the CNN are sent to the right side to retain fine-grained details, improving the final contour prediction and reducing convergence time.
    - *"Similarly to [14], we forward the features extracted from early stages of the left part of the CNN to the right part. This is schematically represented in Figure 2 by horizontal connections. In this way we gather fine grained detail that would be otherwise lost in the compression path and we improve the quality of the final contour prediction."*
- **Receptive Fields and Global Context**:
    - Table 1 shows receptive fields of each layer, illustrating that the innermost layers capture the entire input volume.
    - This global perception helps segment less visible anatomy by providing features that span the entire anatomy of interest, enforcing beneficial global constraints.
    - *"We report in Table 1 the receptive fields of each network layer, showing the fact that the innermost portion of our CNN already captures the content of the whole input volume. We believe that this characteristic is important during segmentation of poorly visible anatomy: the features computed in the deepest layer perceive the whole anatomy of interest at once, since they are computed from data having a spatial support much larger than the typical size of the anatomy we seek to delineate, and therefore impose global constraints."*

One of the key design features here is the use of skip connections to retain fine-grained details that would otherwise be lost. We take features from early stages of the left part of the network and pass them to the corresponding stages in the right pathway. These horizontal connections, illustrated in Figure 2, ensure that we preserve essential details, especially at object boundaries, which improves the accuracy of our final contour prediction. As a bonus, these connections also help the model converge faster.

Finally, let's look at the receptive fields as detailed in Table 1. The receptive fields grow significantly with each layer, and by the innermost layers, they span the entire input volume. This global view is critical, especially for segmenting parts of anatomy that may be hard to see. The network can impose global constraints because these deeper features perceive the anatomy as a whole. This characteristic enables more consistent and holistic segmentation, capturing not just local features but the broader spatial context.

This structured approach highlights how the network handles feature extraction, spatial expansion, and integration of fine details, making it effective for complex tasks like volumetric segmentation.