Semantic Segmentation
The Task
Evaluation metric

- Pixel classification!
- Accuracy?
  - Heavily unbalanced
  - Common classes are over-emphasized
- Intersection over Union
  - Average across classes and images
- Per-class accuracy
  - Compute accuracy for every class and then average
Things vs Stuff

**THINGS**
- Person, cat, horse, etc
- Constrained shape
- Individual instances with separate identity
- May need to look at objects

**STUFF**
- Road, grass, sky etc
- Amorphous, no shape
- No notion of instances
- Can be done at pixel level
- “texture”
Challenges in data collection

• Precise localization is hard to annotate

• Annotating every pixel leads to heavy tails

• Common solution: annotate few classes (often things), mark rest as “Other”

• Common datasets: PASCAL VOC 2012 (~1500 images, 20 categories), COCO (~100k images, 20 categories)
Pre-convnet semantic segmentation

• Things
  • Do object detection, then segment out detected objects

• Stuff
  • "Texture classification"
  • Compute histograms of filter responses
  • Classify local image patches
Semantic segmentation using convolutional networks
Semantic segmentation using convolutional networks
Semantic segmentation using convolutional networks
Semantic segmentation using convolutional networks

Can be considered as a feature vector for a pixel
Semantic segmentation using convolutional networks

Convolve with #classes 1x1 filters

w/4

h/4

#classes
Semantic segmentation using convolutional networks

• Pass image through convolution and subsampling layers
• Final convolution with #classes outputs
• Get scores for subsampled image
• Upsample back to original size
Semantic segmentation using convolutional networks
The resolution issue

• Problem: Need fine details!

• Shallower network / earlier layers?
  • Deeper networks work better: more abstract concepts
  • Shallower network => Not very semantic!

• Remove subsampling?
  • Subsampling allows later layers to capture larger and larger patterns
  • Without subsampling => Looks at only a small window!
Solution 1: Image pyramids

Solution 2: Skip connections

Compute class scores at multiple layers, then upsample and add
Solution 2: Skip connections

Red arrows indicate backpropagation
Skip connections

Skip connections

• Problem: early layers not semantic

Solution 3: Dilation

- Need subsampling to allow convolutional layers to capture large regions with small filters
  - Can we do this without subsampling?
Solution 3: Dilation

• Need subsampling to allow convolutional layers to capture large regions with small filters
  • Can we do this without subsampling?
Solution 3: Dilation

• Need subsampling to allow convolutional layers to capture large regions with small filters
  • Can we do this without subsampling?
Solution 3: Dilation

• Instead of subsampling by factor of 2: dilate by factor of 2
• Dilation can be seen as:
  • Using a much larger filter, but with most entries set to 0
  • Taking a small filter and “exploding”/ “dilating” it
• Not panacea: without subsampling, feature maps are much larger: memory issues
Putting it all together