CPSC 540: Machine Learning

Fully-Convolutional Networks

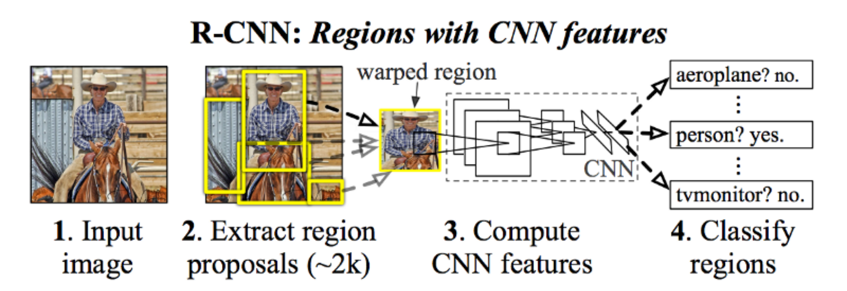
Mark Schmidt

University of British Columbia

Winter 2019

Last Time: "Pipeline" Approach to Object Localization

- Early approach (region CNN):
  1. Propose a bunch of potential boxes.
  2. Compute features of box using a CNN.
  3. Classify each box based on an SVM.
  4. Refine each box using linear regression.

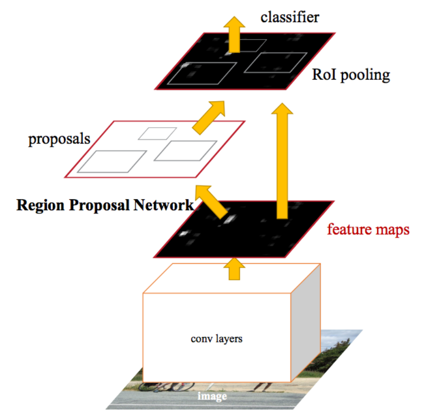


https://blog.athelas.com/a-brief-history-of-cnns-in-image-segmentation-from-r-cnn-to-mask-r-cnn-34ea83205de4

- Improved on state of the art, but not very elegant with its 4 steps.

Region Convolutional Neural Networks: "End to End" Approach

- Modern approaches try to do the whole task with one neural network.
  - The network extracts features, proposes boxes, and classifies boxes.



https://blog.athelas.com/a-brief-history-of-cnns-in-image-segmentation-from-r-cnn-to-mask-r-cnn-34ea83205de4

- This is called an end-to-end model.

End-to-End Computer Vision Models

- Key ideas behind end-to-end systems:
  1. Write each step as a differentiable operator.
  2. Train all steps using backpropagation and stochastic gradient.

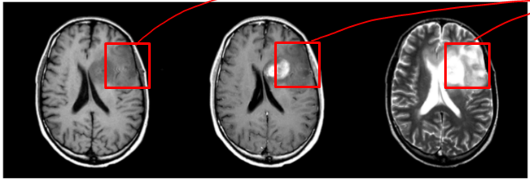
- There now exist end-to-end models for all the standard vision tasks.
  - Depth estimation, pose estimation, optical flow, tracking, 3D geometry, and so on.
  - A bit hard to track the progress at the moment.
  - A survey of ≈ 200 papers from 2016:
    - `http://www.themtank.org/a-year-in-computer-vision`

  .

- Let's focus on the task of pixel labeling...

Straightforward CNN Extensions to Pixel Labeling

- Approach 1: apply an existing CNN to classify pixel given neighbourhood.
  - Misses long range dependencies in the image.
  - It's slow: for 200 by 200 image, need to do forward propagation 40000 times.

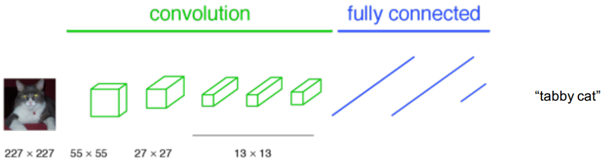


$x_i = (\ldots, \ldots, \ldots)$

- Approach 2: add per-pixel labels to final layer of an existing CNN.
  - Fully-connected layers lose spatial information.
  - Relies on having fixed-size images.

Fully-Convolutional Neural Networks

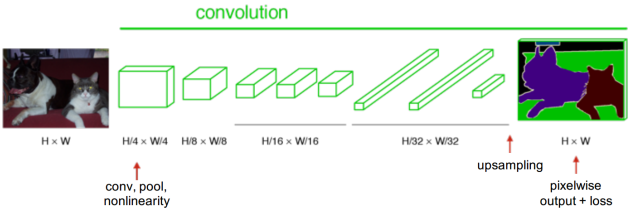
- Classic CNN architecture:



https://leonardoaraujosantos.gitbooks.io/artificial-inteligence/content/image_segmentation.html

Fully-Convolutional Neural Networks

- Fully-convolutional neural networks (FCNs): CNNs with no fully-connected layers.
  - All layers maintain spatial information.



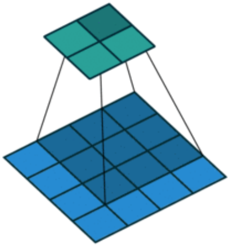
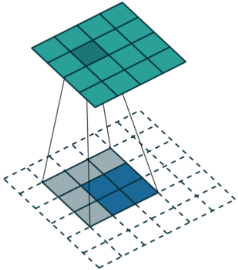
https://leonardoaraujosantos.gitbooks.io/artificial-inteligence/content/image_segmentation.html

- Final layer upsamples to original image size.
  - With a learned “transposed convolution”.

- Parameter tieing within convolutions allows images of different sizes.

Transposed Convolution Layer

- The upsampling layer is also called a transposed convolution or "deconvolution".
  - Implemented as another convolution.

Convolution:  Transposed: 

https://github.com/vdumoulin/conv_arithmetic

- Reasons for the names:
  - "Tranposed" because sparsity pattern is transpose of a downsampling convolution.
  - "Deconvolution" is not related to the "deconvolution" in signal processing.

Fully-Convolutional Neural Networks

- FCNs quickly achieved state of the art results on many tasks.

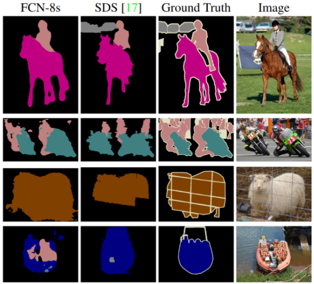


Figure 6. Fully convolutional segmentation nets produce state-of-the-art performance on PASCAL. The left column shows the output of our highest performing net, FCN-8s. The second shows the segmentations produced by the previous state-of-the-art system

https://people.eecs.berkeley.edu/~jonlong/long_shelhamer_fcn.pdf

- FCN end-to-end solution is very elegant compared to previous “pipelines”:
  - No super-pixels, object proposals, merging results from multiple classifiers, and so on.

Variations on FCNs

- The transposed convolution at the last layer can lose a lot of resolution.
- One option is adding "skip" connections from earlier higher-resolution layers.

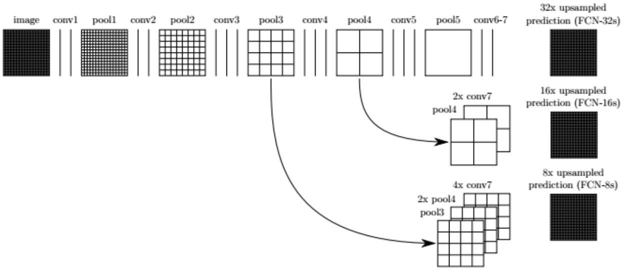
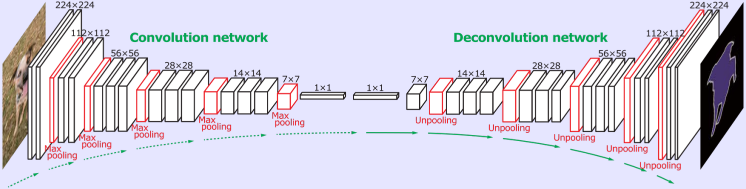


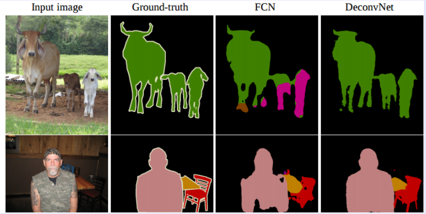
Figure 3. Our DAG nets learn to combine coarse, high layer information with fine, low layer information. Pooling and prediction layers are shown as grids that reveal relative spatial coarseness, while intermediate layers are shown as vertical lines. First row (FCN-32s): Our single-stream net, described in Section 4.1, upsamples stride 32 predictions back to pixels in a single step. Second row (FCN-16s): Combining predictions from both the final layer and the pool4 layer, at stride 16, lets our net predict finer details, while retaining high-level semantic information. Third row (FCN-8s): Additional predictions from pool3, at stride 8, provide further precision.

https://people.eecs.berkeley.edu/~jonlong/long_shelhamer_fcn.pdf

Variations on FCNs

- Another approach to preserving resolution is deconvolutional networks:





https://leonardoaraujosantos.gitbooks.io/artificial-inteligence/content/image_segmentation.html

Combining FCNs and CRFs

- Another way to address this is combining FCNs and CRFs.

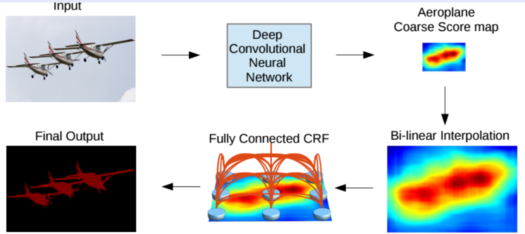


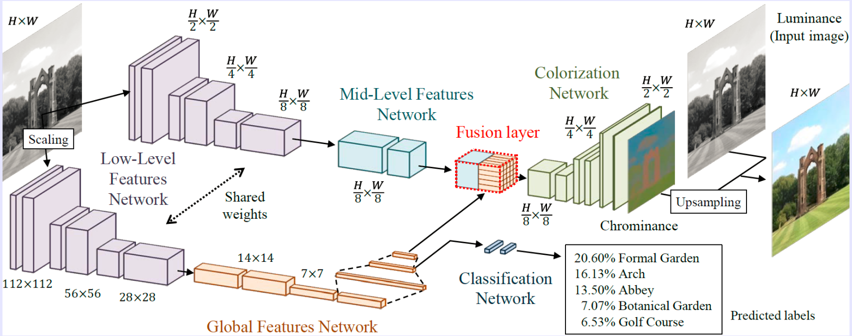
Fig. 1: Model Illustration. A Deep Convolutional Neural Network such as VGG-16 or ResNet-101 is employed in a fully convolutional fashion, using atrous convolution to reduce the degree of signal downsampling (from 32x down 8x). A bilinear interpolation stage enlarges the feature maps to the original image resolution. A fully connected CRF is then applied to refine the segmentation result and better capture the object boundaries.

https://arxiv.org/pdf/1606.00915.pdf

- DeepLab uses a fully-connected pairiwse CRF on output layer.
  - Though most recent version removed CRF.

Image Colourization

- An end-to-end image colorization network:



http://hi.cs.waseda.ac.jp/~iizuka/projects/colorization/en

- Trained to reproduce colour of existing images after removing colour.

Image Colourization

- Image colorization results:

Colorado National Park, 1941 | Textile Mill, June 1937 | Berry Field, June 1909 | Hamilton, 1936

http://hi.cs.waseda.ac.jp/~iizuka/projects/colorization/en

- Gallery:
  http://hi.cs.waseda.ac.jp/~iizuka/projects/colorization/extra.html
- Video: https://www.youtube.com/watch?v=ys5nMO4Q0iY

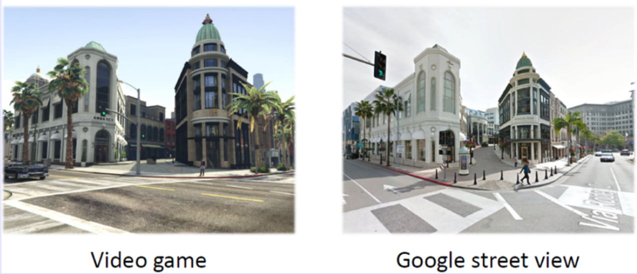
R-CNNs for Pixel Labeling

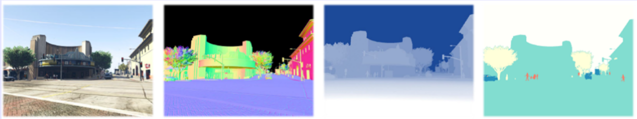
- An alternative approach: learn to apply binary mask to R-CNN results:

https://blog.athelas.com/a-brief-history-of-cnns-in-image-segmentation-from-r-cnn-to-mask-r-cnn-34ea83205de4

Where does data come from?

- Unfortunately, getting densely-labeled data is often hard.
- For pixel labeling and depth estimation, we explored getting data from GTA V:





- Easy to collect data at night, in fog, or in dangerous situations.

Where does data come from?

- Recent works use that you don't need full labeling.
    - Unobserved children in DAG don't induce dependencies.
        - Although you would do better if you have an accurate dense labeling.

- Test object segmentation based on "single pixel" labels from training data:
    - And some tricks to separate objects and remove false positives.

- Show video...

Summary

- End to end models: use a neural network to do all steps.
  - Computer vision can now actually work!
- Fully-convolutional networks:
  - Elegant way to apply convolutional networks for dense labeling problems.