Semantic Segmentation 
and 
Image Processing 

with Convnets
Overview

• Methods where output is also an image
  • Fully Convolutional Nets [Long et al., CVPR 2015]
  • Depth, normals and semantic labels from a single image [Eigen ICCV 2015]

• Image processing with Convnets
  • Image colorization [Zhang et al. ECCV 2016]
A Fuller Understanding of Fully Convolutional Networks

Evan Shelhamer*  Jonathan Long*  Trevor Darrell

UC Berkeley in CVPR'15, PAMI'16
pixels in, pixels out

- semantic segmentation
- monocular depth + normals Eigen & Fergus 2015
- optical flow Fischer et al. 2015
- boundary prediction Xie & Tu 2015
- colorization Zhang et al. 2016
convnets perform classification

< 1 millisecond

“tabby cat”

1000-dim vector

end-to-end learning
lots of pixels, little time?

~1/10 second

end-to-end learning
a classification network

“tabby cat”
becoming fully convolutional
becoming fully convolutional
upsampling output

convolution

H × W

H/4 × W/4

H/8 × W/8

H/16 × W/16

H/32 × W/32

H × W
end-to-end, pixels-to-pixels network
end-to-end, pixels-to-pixels network
spectrum of deep features

combine where (local, shallow) with what (global, deep)

(fuse features into deep jet)

(cf. Hariharan et al. CVPR15 “hypercolumn”)

image

intermediate layers
skip layers

end-to-end, joint learning of semantics and location
skip layer refinement

input image  stride 32  stride 16  stride 8  ground truth
no skips  1 skip  2 skips
skip FCN computation

A multi-stream network that fuses features/predictions across layers
Relative to prior state-of-the-art SDS:

- 30% relative improvement for mean IoU
- 286× faster

*Simultaneous Detection and Segmentation
Hariharan et al. ECCV14
SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation

Max pooling indices transferred to decoder to improve output resolution

https://arxiv.org/abs/1511.00561
UNet: Convolutional Networks for Biomedical Image Segmentation

Segmentation of a 512×512 image takes less than a second on a recent GPU

https://arxiv.org/abs/1505.04597
Further Resources

http://blog.quare.ai/notes/semantic-segmentation-deep-learning-review
Overview

• Methods where output is now an image
  • Fully Convolutional Nets [Long et al., CVPR 2015]
  • Depth, normals and semantic labels from a single image [Eigen ICCV 2015]

• Image processing with Convnets
  • Image colorization [Zhang et al. ECCV 2016]
Beyond Object Classification with Convolutional Networks

David Eigen (NYU -> Clarifai)
Rob Fergus (Facebook / NYU)
Motivation

• Understand input scene
  – Semantic
  – Geometric

Input Image

Semantic Map
Motivation

• Understand input scene
  – Semantic
  – Geometric
Motivation

• Understand input scene
  – Semantic
  – Geometric
• **Predict Pixel Maps from a Single Image**
Architecture

Input: 320x240

Output 1: 19x14
Architecture

Input: 320x240

Output 2: 75x55
Input: 320x240
Output: 147x109
Architecture

Input: 320x240

- Convolutions
- Conv + Pool
- Concat
- Upsample

Conv + Pool — Concat — Convolutions
Architecture

Input: 320x240

conv+pool  concat  convolutions
Losses

Depth:

\[ d = D - D^* \]

\[ L_{\text{depth}}(D, D^*) = \frac{1}{n} \sum_i d_i^2 - \frac{1}{2n^2} \left( \sum_i d_i \right)^2 + \frac{1}{n} \sum_i \left( (\nabla_x d_i)^2 + (\nabla_y d_i)^2 \right) \]

Norm

Label:
Training

- Pre-train Alexnet/VGGnet scale 1 with Imagenet
- Scale 2 & 3 random initialization
- Joint train layers 1 & 2 for each task
  - Loss on output of layer 2

- Fix layers 1 & 2, train layer 3

- For depth & normals task, share scale 1
  - But separate scale 2 & 3’s
  - 1.6x speedup
Evaluation

• NYU Depth dataset
  – RGB, Depth and per-pixel labels
  – Indoor scenes

• Supervised training of models

• Compare to range of other methods
  – Also on SIFTFlow and PASCAL VOC’11
Depths Comparison

<table>
<thead>
<tr>
<th>Eigen NIPS’14 (2 scales)</th>
<th>Ours</th>
<th>Ground Truth</th>
</tr>
</thead>
</table>

[Images of depth comparisons for different environments]
Depth Comparison

- $m3d = \text{Make3D [Saxena & Ng 2006]}$
### Surface Normal Estimation (GT [6])

| Method                  | Angle Distance Mean | Angle Distance Median | Within $t^\circ$ Deg.  
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th>11.25$^\circ$</th>
<th>22.5$^\circ$</th>
<th>30$^\circ$</th>
</tr>
</thead>
<tbody>
<tr>
<td>3DP [6]</td>
<td>34.2</td>
<td>30.0</td>
<td>18.5</td>
<td>38.6</td>
<td>50.0</td>
</tr>
<tr>
<td>Ladicky &amp; al [16]</td>
<td>32.5</td>
<td>22.3</td>
<td>27.4</td>
<td>50.2</td>
<td>60.1</td>
</tr>
<tr>
<td>Fouhey &amp; al [7]</td>
<td>35.1</td>
<td>19.2</td>
<td>37.6</td>
<td>53.3</td>
<td>58.9</td>
</tr>
<tr>
<td>Wang &amp; al [33]</td>
<td>26.6</td>
<td>15.3</td>
<td>40.1</td>
<td>61.4</td>
<td>69.0</td>
</tr>
<tr>
<td>Ours (AlexNet)</td>
<td>23.1</td>
<td>15.1</td>
<td>39.4</td>
<td>63.6</td>
<td>72.7</td>
</tr>
<tr>
<td>Ours (VGG)</td>
<td>20.5</td>
<td>13.2</td>
<td>44.0</td>
<td>68.5</td>
<td>77.2</td>
</tr>
</tbody>
</table>

### Surface Normal Estimation (GT [27])

| Method                  | Angle Distance Mean | Angle Distance Median | Within $t^\circ$ Deg.  
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th>11.25$^\circ$</th>
<th>22.5$^\circ$</th>
<th>30$^\circ$</th>
</tr>
</thead>
<tbody>
<tr>
<td>3DP [6]</td>
<td>37.7</td>
<td>34.1</td>
<td>14.0</td>
<td>32.7</td>
<td>44.1</td>
</tr>
<tr>
<td>Ladicky &amp; al [16]</td>
<td>35.5</td>
<td>25.5</td>
<td>24.0</td>
<td>45.6</td>
<td>55.9</td>
</tr>
<tr>
<td>Wang &amp; al [33]</td>
<td>28.8</td>
<td>17.9</td>
<td>35.2</td>
<td>57.1</td>
<td>65.5</td>
</tr>
<tr>
<td>Ours (AlexNet)</td>
<td>25.9</td>
<td>18.2</td>
<td>33.2</td>
<td>57.5</td>
<td>67.7</td>
</tr>
<tr>
<td>Ours (VGG)</td>
<td>22.2</td>
<td>15.3</td>
<td>38.6</td>
<td>64.0</td>
<td>73.9</td>
</tr>
</tbody>
</table>
Results: Normals

Angle from Ground Truth

<table>
<thead>
<tr>
<th>Method</th>
<th>Mean</th>
<th>Median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ours+VGG</td>
<td>18</td>
<td>7.5</td>
</tr>
<tr>
<td>Ours</td>
<td>27</td>
<td>15</td>
</tr>
<tr>
<td>Wang &amp; al</td>
<td>36</td>
<td>22.5</td>
</tr>
<tr>
<td>Fouhey &amp; al '14</td>
<td>36</td>
<td>30</td>
</tr>
<tr>
<td>Ladicky &amp; al</td>
<td>36</td>
<td>22.5</td>
</tr>
<tr>
<td>3DP</td>
<td>36</td>
<td>30</td>
</tr>
</tbody>
</table>
Output from each scale

Input

Depth

Coarse → Fine

Normals
Semantic Labels: NYUD
Results: NYUD 40 Classes

- Use RGB + ground truth depth & normals as inputs

---

<table>
<thead>
<tr>
<th>Method</th>
<th>Per-Pixel Acc.</th>
<th>Per-Class Acc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ours (VGG)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Long &amp; al</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ours</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gupta &amp; al '14</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gupta &amp; al '13</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Ours (VGG) achieved a Per-Pixel Accuracy of 62.5% and a Per-Class Accuracy of 48%.
- Long & al achieved a Per-Pixel Accuracy of 60.75% and a Per-Class Accuracy of 41%.
- Gupta & al '14 achieved a Per-Pixel Accuracy of 64.25% and a Per-Class Accuracy of 34%.
- Gupta & al '13 achieved a Per-Pixel Accuracy of 59% and a Per-Class Accuracy of 27%.
Results: NYUD Labels

- Use RGB + ground truth depth & normals as inputs

### 4 Classes: Pixel Acc.

- **Ours**: 82.5
- **Gupta & al**: 75
- **Mueller & al**: 67.5
- **Stuckler & al**: 60
- **Khan & al**: 60
- **Couprie & al**: 42.5

### 13 Classes: Pixel Acc.

- **Ours**: 80
- **Khan**: 67.5
- **Hermans**: 55
- **Wang**: 42.5
- **Couprie**: 30
Semantic Labels: Pascal VOC’11

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Long et al [19]</td>
<td>90.3</td>
<td>75.9</td>
<td>83.2</td>
<td>62.7</td>
</tr>
<tr>
<td>Ours (VGG)</td>
<td>90.3</td>
<td>72.4</td>
<td>82.9</td>
<td>62.2</td>
</tr>
</tbody>
</table>
Contribution from different scales

- On NYU Depth

<table>
<thead>
<tr>
<th>Contributions of Scales</th>
<th>Depth</th>
<th>Normals</th>
<th>4-Class</th>
<th>13-Class</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>RGB+D+N</td>
<td>RGB</td>
</tr>
<tr>
<td></td>
<td>Pixelwise Error</td>
<td>Pixelwise Accuracy</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Scale 1 only</td>
<td>0.218</td>
<td>29.7</td>
<td>71.5</td>
<td>58.1</td>
</tr>
<tr>
<td>Scale 2 only</td>
<td>0.290</td>
<td>31.8</td>
<td>77.4</td>
<td>65.1</td>
</tr>
<tr>
<td>Scales 1 + 2</td>
<td>0.216</td>
<td>26.1</td>
<td>80.1</td>
<td>69.8</td>
</tr>
<tr>
<td>Scales 1 + 2 + 3</td>
<td>0.198</td>
<td>25.9</td>
<td>80.6</td>
<td>70.5</td>
</tr>
</tbody>
</table>

- Depth & normals: scale 1 most important
- Semantic labels: scale 2 most important (if D & N are available)
Using Predicted Depths

• Use predicted depth/normals as input?

- NYU Depth 13-class
  - Per-Pixel Acc.
    - Scales 1+2
      - RGB only: 53.1
      - RGB + Pred D&N: 65.1
      - RGB + GT D&N: 69.8
    - Scale 2 only
      - RGB only: 38.3
      - RGB + Pred D&N: 43.8
      - RGB + GT D&N: 52.3

- Per-Class Acc.
  - Scales 1+2
    - RGB only: 49.5
    - RGB + Pred D&N: 50.6
    - RGB + GT D&N: 58.9
  - Scale 2 only
    - RGB only: 43.8
    - RGB + Pred D&N: 52.3
    - RGB + GT D&N: 58.9

RGB only | RGB + Pred D&N | RGB + GT D&N
Summary

• Relatively simple multi-scale model gives good results for depth, normals & labels

• Coarse interpretation of scene important for understanding depth/normals


• Code available
Overview

• Methods where output is also an image
  – Fully Convolutional Nets [Long et al., CVPR 2015]
  – Depth, normals and semantic labels from a single image [Eigen ICCV 2015]

• Image processing with Convnets
  – Image colorization [Zhang et al. ECCV 2016]
Denoising with ConvNets

- Burger et al. “Can plain NNs compete with BM3D?” CVPR 2012
Deblurring with Convnets

- Blind deconvolution
  - Learning to Deblur, Schuler et al., arXiv 1406.7444, 2014
Inpainting with Convnets

- Image Denoising and Inpainting with Deep Neural Networks, Xie et al. NIPS 2012.
- Mask-specific inpainting with deep neural networks, Köhler et al., Pattern Recognition 2014
Removing Local Corruption

• Restoring An Image Taken Through a Window Covered with Dirt or Rain, Eigen et al., ICCV 2013.
Removing Local Corruption

Restoring An Image Taken Through a Window Covered with Dirt or Rain

Rain Sequence
Each frame processed independently

David Eigen, Dilip Krishnan and Rob Fergus
ICCV 2013
Overview

• Methods where output is now also an image
  • Fully Convolutional Nets [Long et al., CVPR 2015]
  • Depth, normals and semantic labels from a single image [Eigen ICCV 2015]

• Image processing with Convnets
  • Image colorization [Zhang et al. ECCV 2016]
Colorful Image Colorization
Richard Zhang, Phillip Isola, Alexei (Alyosha) Efros
richzhang.github.io/colorization
Ansel Adams, Yosemite Valley Bridge
Ansel Adams, Yosemite Valley Bridge - Our Result
Grayscale image: $L$ channel

$X \in \mathbb{R}^{H \times W \times 1}$

Color information: $ab$ channel

$\hat{Y} \in \mathbb{R}^{H \times W \times 2}$
Grayscale image: $L$

$X \in \mathbb{R}^{H \times W \times I}$

Semantics? Higher-level abstraction?

Concatenate $(L, ab)$

$(X, \hat{Y})$

“Free” supervisory signal
Inherent Ambiguity

Grayscale
Inherent Ambiguity

Our Output

Ground Truth
Better Loss Function

- Regression with L2 loss
  \[ \text{inac}_{L_2}(\hat{Y}, Y) = \frac{1}{2} \sum_{h,w} \|Y_{h,w} - \hat{Y}_{h,w}\|^2 \]

Colors in \(ab\) space (continuous)
• Regression with L2 loss inadequate

\[ L_2(\hat{Y}, Y) = \frac{1}{2} \sum_{h,w} \| Y_{h,w} - \hat{Y}_{h,w} \|_2^2 \]

• Use multinomial classification

\[ L(\hat{Z}, Z) = -\frac{1}{HIW} \sum_{h,w} \sum_q Z_{h,w,q} \log(\hat{Z}_{h,w,q}) \]

• Class rebalancing to encourage

\[ L(\hat{Z}, Z) = -\frac{1}{HW} \sum_{h,w} v(Z_{h,w}) \sum_q Z_{h,w,q} \log(\hat{Z}_{h,w,q}) \]
Better Loss Function

- Regression with L2 loss inadequate

$$L_2(\hat{Y}, Y) = \frac{1}{2} \sum_{h,w} \| Y_{h,w} - \hat{Y}_{h,w} \|^2_2$$

- Use multinomial classification

$$L(\hat{Z}, Z) = -\frac{1}{HW} \sum_{h,w} \sum_q Z_{h,w,q} \log(\hat{Z}_{h,w,q})$$

- Class rebalancing to encourage

$$L(\hat{Z}, Z) = -\frac{1}{HW} \sum_{h,w} v(Z_{h,w}) \sum_q Z_{h,w,q} \log(\hat{Z}_{h,w,q})$$
Network Architecture

\[ \hat{Y} = \mathcal{F}(X) \]

\[ \hat{Z} \in [0, 1]^{H \times W \times Q} \]

conv1 64
conv2 128
conv3 256
conv4 512
conv5 512
fc6 4096
fc7 4096

lightness

ab color

+L
Network Architecture

\[
\hat{Z} = \mathcal{G}(X)
\]

\[
\hat{Y} = \mathcal{H}(\hat{Z})
\]

Failure Cases
Biases
Evaluation

Visual Quality

Quantitative

- Per-pixel accuracy
- Perceptual realism
- Semantic interpretability

Task generalization
- ImageNet classification
- Task & dataset generalization
- PASCAL classification, detection, segmentation

Qualitative

- Low-level stimuli
- Legacy grayscale photos
- Hidden unit activations
<table>
<thead>
<tr>
<th></th>
<th>Visual Quality</th>
<th>Representation Learning</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Quantitative</strong></td>
<td>Per-pixel accuracy</td>
<td>Task generalization</td>
</tr>
<tr>
<td></td>
<td>Perceptual realism</td>
<td>ImageNet classification</td>
</tr>
<tr>
<td></td>
<td>Semantic interpretability</td>
<td>Task &amp; dataset generalization</td>
</tr>
<tr>
<td></td>
<td>Low-level stimuli</td>
<td>PASCAL classification, detection, segmentation</td>
</tr>
<tr>
<td><strong>Qualitative</strong></td>
<td>Legacy grayscale photos</td>
<td>Hidden unit activations</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# Evaluation

<table>
<thead>
<tr>
<th></th>
<th>Visual Quality</th>
<th>Representation Learning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quantitative</td>
<td>Per-pixel accuracy</td>
<td>Task generalization</td>
</tr>
<tr>
<td></td>
<td>Perceptual realism</td>
<td>ImageNet classification</td>
</tr>
<tr>
<td></td>
<td>Semantic interpretability</td>
<td>Task &amp; dataset generalization</td>
</tr>
<tr>
<td></td>
<td>Low-level stimuli</td>
<td>PASCAL classification, detection, segmentation</td>
</tr>
<tr>
<td>Qualitative</td>
<td>Legacy grayscale photos</td>
<td>Hidden unit activations</td>
</tr>
</tbody>
</table>

Perceptual Realism / Amazon Mechanical Turk Test
clap if “fake”

clap if “fake”
Fake, 0% fooled
clap if “fake”

clap if “fake”
Fake, 55% fooled
clap if "fake"

clap if "fake"
Fake, 58% fooled
from Reddit /u/SherySantucci
Photo taken by Reddit /u/ Timteroo, Mural from street artist Eduardo Kobra
Perceptual Realism Test

<table>
<thead>
<tr>
<th></th>
<th>Ground Truth</th>
<th>Ours (L2)</th>
<th>Ours (class)</th>
<th>Ours (full)</th>
<th>Larsson et al. [Concurrent]</th>
<th>Random</th>
</tr>
</thead>
<tbody>
<tr>
<td>AMT Labeled</td>
<td>50%</td>
<td>32.3</td>
<td>23.9</td>
<td>21.2</td>
<td>27.2</td>
<td>13.0</td>
</tr>
<tr>
<td>Real [%]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1600 images tested per algorithm
Does the method work on *legacy* black and white photos?
Thylacine, Dr. David Fleay, extinct in 1936.
Thylacine, Dr. David Fleay, extinct in 1936.
Amateur Family Photo, 1956.
Henri Cartier-Bresson, Sunday on the Banks of the River Seine, 1938.
Henri Cartier-Bresson, Sunday on the Banks of the River Seine, 1938.
Dorothea Lange, Migrant Mother, 1936.
Dorothea Lange, Migrant Mother, 1936.
Additional Information

• Demo

• Reddit ColorizeBot
  – Type “colorizebot” under any image post

• Code
  – https://github.com/richzhang/colorization

• Website – full paper, user examples, visualizations
  – http://richzhang.github.io/colorization
For the full paper, additional examples and our model:
richzhang.github.io/colorization