Visualizing and explaining neural networks

https://deepdreamgenerator.com/
Outline

• Basic visualization techniques
• Mapping activations back to the image
• Synthesizing images to maximize activation
• Saliency maps
• Quantifying interpretability of units
Overview and basic visualization techniques

Visualize first-layer weights directly
Overview and basic visualization techniques

Features from a CIFAR10 network, via Stanford CS231n
Overview and basic visualization techniques

Visualize maximally activating patches:
pick a unit; run many images through the network; visualize patches that produce the highest output values
Overview and basic visualization techniques

Visualize maximally activating patches
Overview and basic visualization techniques

Synthesize images to maximize activation
Overview and basic visualization techniques

What about FC layers?
Visualize nearest neighbor images according to activation vectors

Source: Stanford CS231n
Overview and basic visualization techniques

What about FC layers?
Fancy dimensionality reduction, e.g., \( t\text{-SNE} \)

Source: Andrej Karpathy
Overview and basic visualization techniques

Given: a particular input image
Overview and basic visualization techniques

Given: a particular input image

Visualize activations for this image

“cat”
Overview and basic visualization techniques

Given: a particular input image

Visualize activations for this image

“cat”

Visualize pixel values responsible for the activation

Source
Deep visualization toolbox

YouTube video

Outline

• Basic visualization techniques
• Mapping activations back to the image
• Synthesizing images to maximize activation
• Saliency maps
• Quantifying interpretability of units
Mapping activations back to pixels

- Let’s take a single value in an intermediate feature map and propagate its gradient back to the original image pixels
- What does this tell us?
Mapping activations back to pixels

1. Forward an image through the network
2. Choose a feature map and an activation
3. Zero out all values except for the one of interest
4. Propagate that value back to the image

Figure source
Mapping activations back to pixels

- Commonly used methods differ in how they treat the ReLU

```
<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-1</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>-5</td>
<td>-7</td>
</tr>
<tr>
<td>-3</td>
<td>2</td>
<td>4</td>
</tr>
</tbody>
</table>
```

```
<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>2</td>
<td>4</td>
</tr>
</tbody>
</table>
```

Propagating back negative gradients bad for visualization

Deconvnet visualization

AlexNet Layer 4

AlexNet Layer 5

M. Zeiler and R. Fergus, *Visualizing and Understanding Convolutional Networks*, ECCV 2014
Guided backpropagation visualization

Outline

• Basic visualization techniques
• Mapping activations back to the image
• Synthesizing images to maximize activation
• Saliency maps
• Quantifying interpretability of units
Visualization by optimization (model inversion)

- How can we synthesize images that maximize activation of a given neuron?
- Basic approach: find image $x$ maximizing target activation $f(x)$ subject to natural image regularization penalty $R(x)$:

$$x^* = \arg \max_x f(x) - \lambda R(x)$$
Visualization by optimization (model inversion)

- Maximize $f(x) - \lambda R(x)$
  - $f(x)$ is score for a category before softmax
  - $R(x)$ is L2 regularization
  - Perform gradient ascent starting with zero image, add dataset mean to result

Visualization by optimization (model inversion)

- Alternative approach to regularization: at each step of gradient ascent, apply operator $r$ that regularizes the image:

$$x \leftarrow r \left( x + \eta \frac{\partial f}{\partial x} \right)$$

- Combination that gives good-looking results:
  - L2 decay
  - Gaussian blur (every few iterations)
  - Clip pixel values with small magnitude
  - Clip pixel values with small contribution to the activation (estimated by product of pixel value and gradient)

Visualization by optimization (model inversion)

- Example visualizations:

Visualization by optimization (model inversion)

- Example visualizations of intermediate features:

Multifaceted feature visualization

- Key idea: most neurons in high layers respond to a mix of different patterns or “facets”
- For coherent visualizations, zero in on individual facets

A. Nguyen, J. Yosinski, J. Clune, Multifaceted Feature Visualization: Uncovering the Different Types of Features Learned By Each Neuron in Deep Neural Networks, ICML workshop, 2016
Multifaceted feature visualization

- Key idea: most neurons in high layers respond to a mix of different patterns or “facets”
- For coherent visualizations, zero in on individual facets

- Algorithm:
  - Cluster FC activations of training images to identify facets
  - For each facet, initialize optimization with mean image of that facet
  - To attempt to produce image of a single object, use center-biased regularization (start with blurry image, gradually increase resolution and update center pixels more than edge pixels)

A. Nguyen, J. Yosinski, J. Clune, Multifaceted Feature Visualization: Uncovering the Different Types of Features Learned By Each Neuron in Deep Neural Networks, ICML workshop, 2016
Multifaceted feature visualization

A. Nguyen, J. Yosinski, J. Clune, Multifaceted Feature Visualization: Uncovering the Different Types of Features Learned By Each Neuron in Deep Neural Networks, ICML workshop, 2016
Google DeepDream

Choose an image and a layer in a CNN; repeat:

1. Forward: compute activations at chosen layer
2. Set gradient of chosen layer equal to its activation
   • Equivalent to maximizing $\sum_i f_i^2(x)$
3. Backward: Compute gradient w.r.t. image
4. Update image (with some tricks)

Source: Stanford CS231n

https://ai.googleblog.com/2015/06/inceptionism-going-deeper-into-neural.html
https://deepdreamgenerator.com/
Dreaming to distill

- Key idea: add regularization terms to encourage the mean and variance of values in intermediate feature maps to match batchnorm statistics of the network

H. Yin et al. *Dreaming to Distill: Data-free Knowledge Transfer via DeepInversion*. CVPR 2020
Dreaming to distill: Results

H. Yin et al. Dreaming to Distill: Data-free Knowledge Transfer via DeepInversion. CVPR 2020
Outline

• Basic visualization techniques
• Mapping activations back to the image
• Synthesizing images to maximize activation
• Saliency maps
• Quantifying interpretability of units
Saliency maps

• Which parts of the image played the most important role in the network’s decision?

Prediction: “car” 64%

Source: K. Saenko
“White box” saliency via gradients

- Backpropagate gradient of class score (before softmax) to the image, display max of absolute values across color channels

“White box” saliency via gradients

- Can be used for *weakly supervised* segmentation:

Gradient-weighted class activation mapping (Grad-CAM)

“Black box” saliency via masking

- Slide square occluder across image, see how class score changes

M. Zeiler and R. Fergus, *Visualizing and Understanding Convolutional Networks*, ECCV 2014
“Black box” saliency via masking

- Saliency of a class at a pixel is expected score for that class over all masks where the pixel is visible

“Black box” saliency via masking

• Saliency of a class at a pixel is expected score for that class over all masks where the pixel is visible

V. Petsiuk, A. Das, K. Saenko, RISE: Randomized Input Sampling for Explanation of Black-box Models, BMVC 2018
“Black box” saliency via masking

• Application: detecting model/dataset bias

Prediction: “cow” 76%

Source: K. Saenko
“Black box” saliency via masking

- Application: detecting model/dataset bias

Baseline: A *man* sitting at a desk with a laptop computer.

Improved model: A *woman* sitting in front of a laptop computer.

“Black box” saliency via masking

- Application: detecting model/dataset bias

RISE applied to satellite image classification model shows that shadows have great influence on the model

Source: RISE poster
Outline

• Basic visualization techniques
• Mapping activations back to the image
• Synthesizing images to maximize activation
• Saliency maps
• Quantifying interpretability of units
Quantifying interpretability of units

- From the beginning, people have observed that many units in higher layers seem to fire on meaningful concepts
- But how can we quantify this?

Figure: Zeiler & Fergus
Quantifying interpretability of units

- For a given unit, measure the overlap between areas of high activation and semantic segmentations for a large set of visual concepts.

Quantifying interpretability of units

- For a given unit, measure the overlap between areas of high activation and semantic segmentations for a large set of visual concepts

Quantifying interpretability of units

Histogram of object detectors for Places AlexNet conv5 units
81/256 units with IoU > 0.04

conv5 unit 79  car (object)  IoU=0.13

conv5 unit 107  road (object)  IoU=0.15

Quantifying interpretability of units

Comparison of number of unique detectors across architectures

Summary

• Basic visualization techniques
  • Showing weights, top activated patches, nearest neighbors
• Mapping activations back to the image
  • Deconvolution
  • Guided back-propagation
• Synthesizing images to maximize activation
  • Gradient ascent with natural image regularization
• Saliency maps
  • “White box” vs. “black box”
• Explainability/interpretability
  • Explaining network decisions, detecting bias
  • Quantifying interpretability of intermediate units