

CAP 6412 Advanced Computer Vision

<http://www.cs.ucf.edu/~bgong/CAP6412.html>

Boqing Gong

Jan 21, 2016

Today

- Administrivia
- Neural networks & backpropagation (Part III)
- Visualizing and understanding CNNs, by Jason

Past due (3pm today)

- Assignment 1: Review the following paper

[Visualization] Zeiler, Matthew D., and Rob Fergus. “Visualizing and understanding convolutional networks.” In *Computer Vision—ECCV 2014*, pp. 818-833. Springer International Publishing, 2014.

Template for paper review:

<http://www.cs.ucf.edu/~bgong/CAP6412/Review.docx>

Assignment 2 (due 01/26 Tuesday, 12pm)

- Review the following paper
- Hosang, Jan, Rodrigo Benenson, Piotr Dollár, and Bernt Schiele. "What makes for effective detection proposals?." *arXiv preprint arXiv:1502.05082* (2015).

Template for paper review:

<http://www.cs.ucf.edu/~bgong/CAP6412/Review.docx>

An assignment with no due dates

- See “Paper Presentation” on UCF webcourse
- Sharing your slides
 - **Refer to the original sources of images, figures, etc. in your slides**
 - Convert them to a PDF file
 - Upload the PDF file to “Paper Presentation” after your presentation

Next week: CNN & object localization

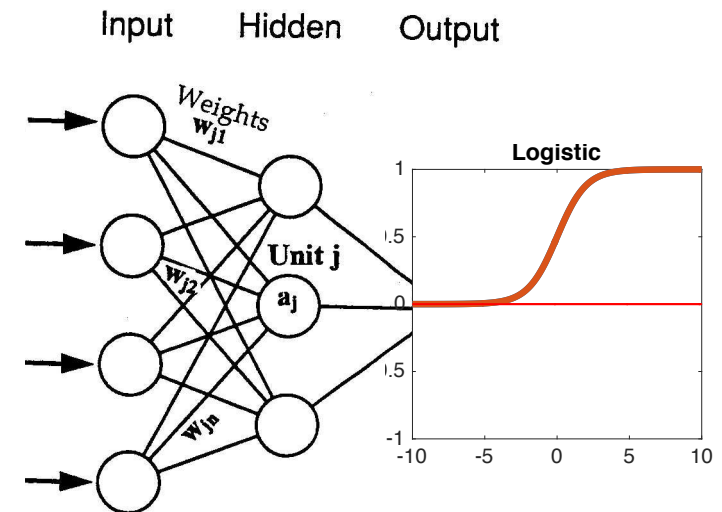
<p>Tuesday (01/26)</p> <p>Samer Iskander</p>	<p>J. Hosang, R. Benenson, and B. Schiele. How good are detection proposals, really? BMVC 2014.</p> <p>{Major} J. Hosang, R. Benenson, P. Dollár, and B. Schiele. What makes for effective detection proposals? PAMI 2015.</p> <p>{Major} [Faster R-CNN] Ren, Shaoqing, Kaiming He, Ross Girshick, and Jian Sun. "Faster R-CNN: Towards real-time object detection with region proposal networks." In <i>Advances in Neural Information Processing Systems</i>, pp. 91-99. 2015.</p>
<p>Thursday (01/28)</p> <p>Syed Ahmed</p>	<p>{Major} [R-CNN] Girshick, Ross, Jeff Donahue, Trevor Darrell, and Jagannath Malik. "Rich feature hierarchies for accurate object detection and semantic segmentation." In <i>Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on</i>, pp. 580-587. IEEE, 2014.</p> <p>[Fast R-CNN] Girshick, Ross. "Fast R-CNN." <i>arXiv preprint arXiv:1504.08083</i> (2015).</p>

Today

- Administrivia
- **Neural networks & backpropagation (Part III)**
- Visualizing and understanding CNNs, by Jason

Review: A case study

- Binary classification
 - Output between 0 and 1
 - Tells the probability of the input x belonging to either class $+1/-1$
- Step 1: choose network structure
- Step 2: choose activation function
- Step 3: determine the model parameters Θ ,
to meet desired properties



Review: Learning the model parameters Θ

- Is equivalent to

Choose one hypothesis $h \in \mathcal{H}$ to approximate concept c

- where,

Binary classification concept: $c : \mathcal{X} \mapsto \mathcal{Y} = \{0, 1\}$

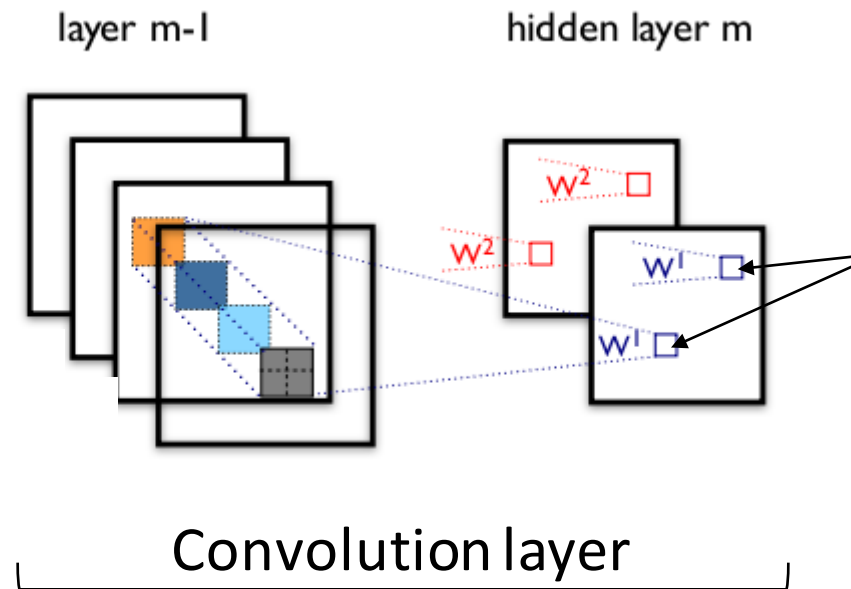
Hypotheses $\mathcal{H} = \{\text{NET}(\Theta) \mid \Theta_d \in \mathbb{R}\}$

- **Discussion:** complexity (capacity, richness) of $\mathcal{H} = \{\text{NET}(\Theta) \mid \Theta_d \in \mathbb{R}\}$

Detour: Complexity of hypotheses

- Related notions (optional):
 - Rademacher complexity
 - VC dimension (Vapnik-Chervonenkis dimension)
 - Growth function
 - Bayesian information criterion for model selection
 - Akaike information criterion for model selection
- Informal notion: number of parameters 🗨
- **In general, low complexity gives rise to high learning efficiency**

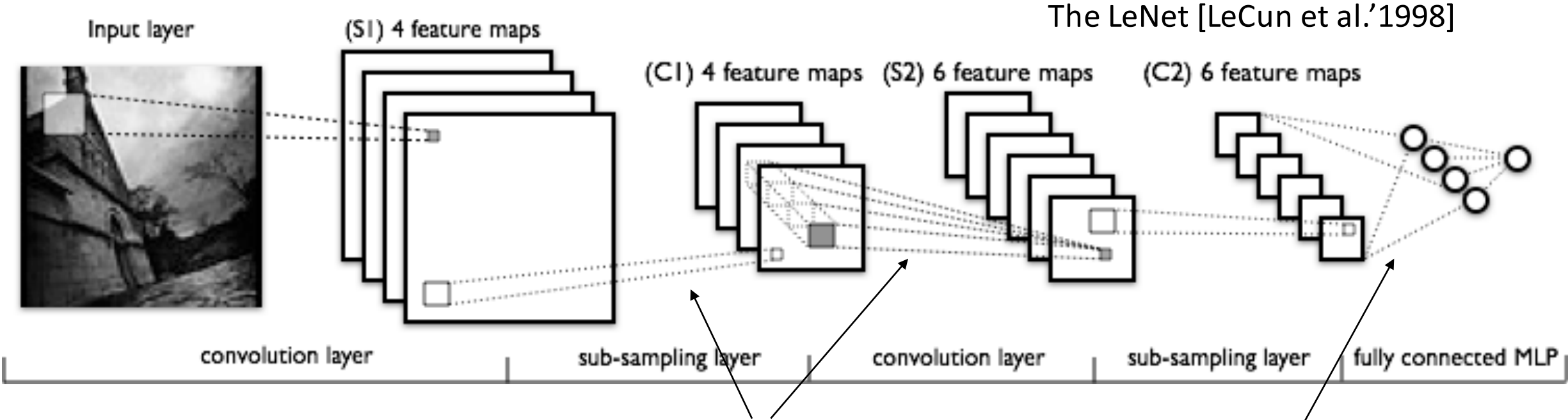
Detour: Weight sharing in CNN



Neurons of the same feature map share the same weights (the filter)

Significantly reduced # parameters

Detour: Sparse connection in CNN



Sparse connections vs. Full connection

Smaller # parameters,
better learning efficiency

Review: Learning the model parameters Θ

- Is equivalent to

Choose one hypothesis $h \in \mathcal{H}$ to approximate concept c

- by,

$$\Theta^* \leftarrow \arg \min_{\Theta} \mathbb{E}_{(x,y) \sim P_{XY}} [\text{NET}(x; \Theta) \neq y]$$

Unknown underlying distribution P_{XY}

$$\Theta^* \leftarrow \arg \min_{\Theta} \mathbb{E}_{(x,y) \sim P_{XY}} [\text{NET}(x; \Theta) \neq y]$$

Unknown underlying distribution P_{XY}
→ Empirical risk minimization

$$\Theta^* \leftarrow \arg \min_{\Theta} \mathbb{E}_{(x,y) \sim P_{XY}} [\text{NET}(x; \Theta) \neq y]$$

$$\hat{\Theta} \leftarrow \arg \min_{\Theta} \frac{1}{n} \sum_{i=1}^n [\text{NET}(x_i; \Theta) \neq y_i]$$

$\hat{\Theta} \rightarrow \Theta^*$ given many training data $(x_i, y_i), i = 1, 2, \dots, n$

Optimization ... and we are done!

$$\hat{\Theta} \leftarrow \arg \min_{\Theta} \frac{1}{n} \sum_{i=1}^n [\text{NET}(x_i; \Theta) \neq y_i]$$

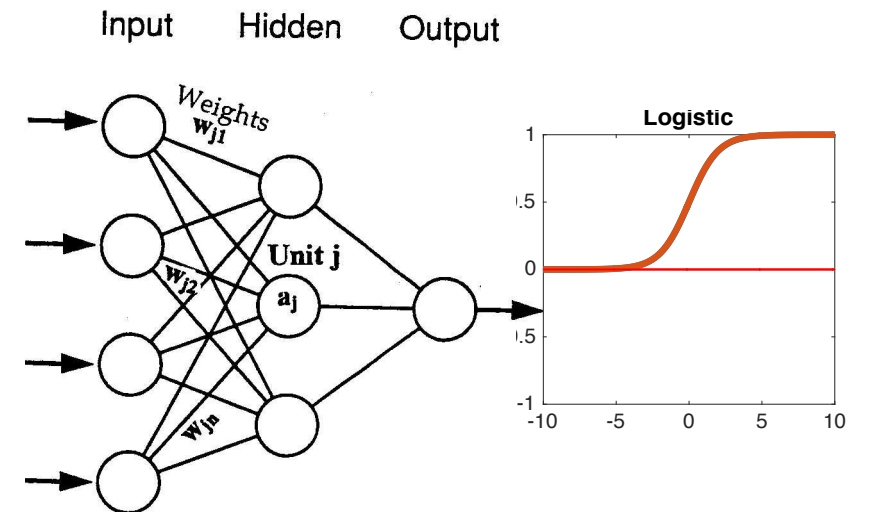
Optimization ... and we are done?

- Challenge: non-differentiable loss function

$$\text{NET}(x_i; \Theta) \in [0, 1], \quad y_i \in 0, 1$$

$$\hat{\Theta} \leftarrow \arg \min_{\Theta} \frac{1}{n} \sum_{i=1}^n [\text{NET}(x_i; \Theta) \neq y_i]$$

0-1 Loss function



Optimization ... with a new loss function

- Squared loss is differentiable everywhere
- It is cheap to optimize using gradient descent

$$\hat{\Theta} \leftarrow \arg \min_{\Theta} \frac{1}{n} \sum_{i=1}^n \underbrace{(y_i - \text{NET}(x_i; \Theta))^2}_{\text{Squared Loss}}$$

Next class

- Other loss functions
- Optimization by gradient descent & back-propagation

$$\hat{\Theta} \leftarrow \arg \min_{\Theta} \frac{1}{n} \sum_{i=1}^n \underbrace{(y_i - \text{NET}(x_i; \Theta))^2}_{\text{Squared Loss}}$$

Today

- Administrivia
- Neural networks & backpropagation (Part III)
- **Visualizing and understanding CNNs, by Jason**



Visualizing and Understanding Convolutional Networks

Matthew D. Zeiler,
Rob Fergus
New York University

Presented by Jason Tiller (JasonTiller@knights.ucf.edu)

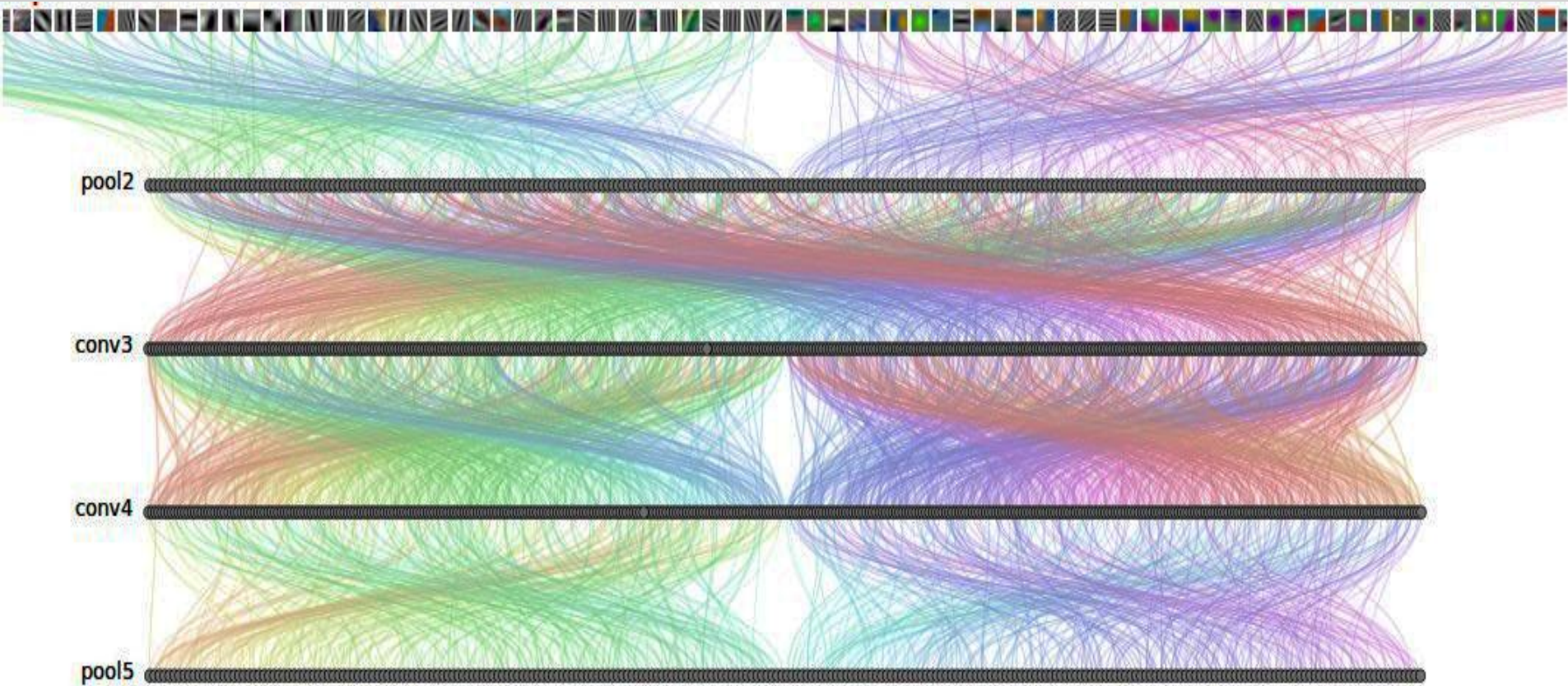


Motivation

- CNNs are the current state-of-the-art solutions for many problem domains, including most object and scene-recognition benchmarks
- However, we have little insight into *what* the CNN is learning
- We're stuck with "trial-and-error" approaches to improving our CNN models



DrawCNN: visualizing the Places dataset





Problem Statement

- CNNs are incredibly powerful, but operate as somewhat of a “black box.”
- In order to better construct the architecture for a CNN, we need to be able to visualize and understand the inner representations: features learned, sensitivity, etc.



Main Contribution of the Paper

- Introduces a novel visualization technique (DeconvNet) that gives insight into the function of intermediate feature layers and the operation of the classifier
- Occlusion tolerance: shows object (not context) recognition
- Beats performance of 2012 ImageNet benchmark champion
- Feature generalization: CNN performs well on other datasets

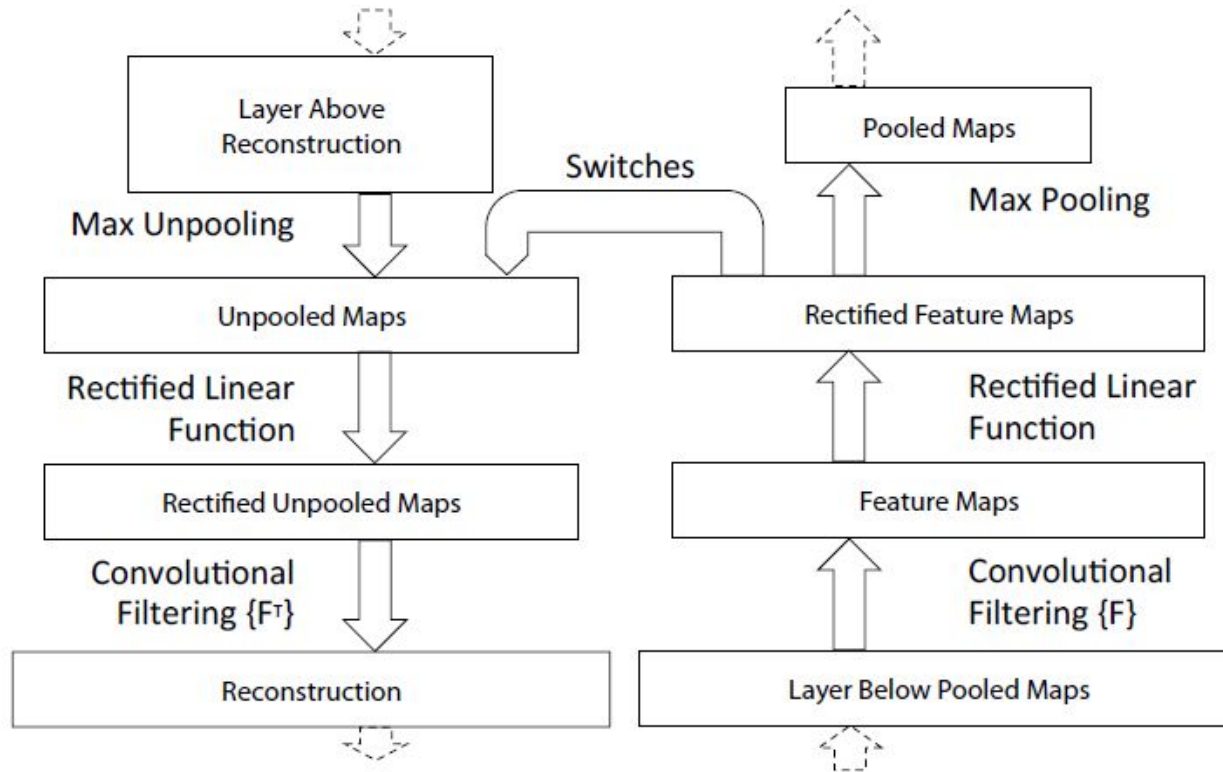


Approach Outline

- DeconvNet
 - Unpooling
 - Rectification
 - Filtering
- Feature Generalization
 - Retraining softmax classifier



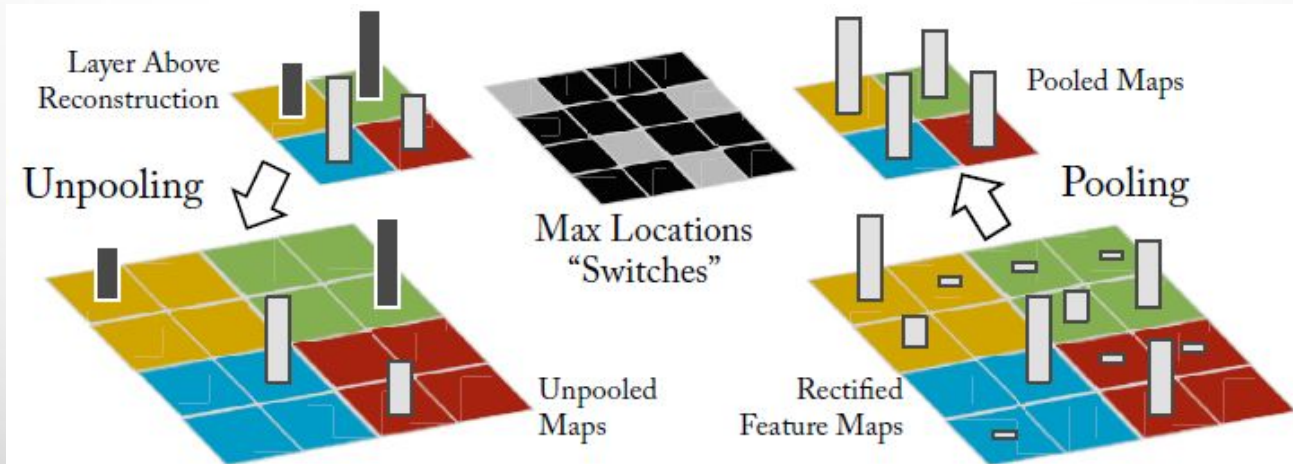
DeconvNet: The Big Picture





Unpooling

- Max pooling is non-invertible
- Approx inverse by recording the location of the maximum pixel (“switch” variable)
- This preserves the general structure of the stimulus

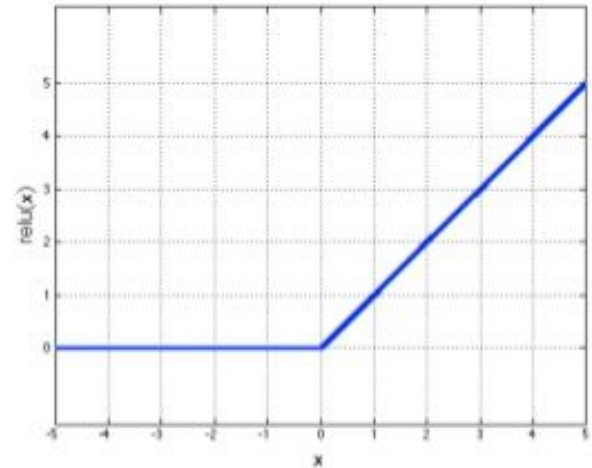




Rectification

- ConvNet uses ReLU nonlinearities to ensure feature maps are positive
- Valid feature reconstructions must also be positive
- We pass reconstructed signals through a ReLU nonlinearity for an approximate reconstruction of rectified unpooled maps

Rectified Linear Unit (ReLU)



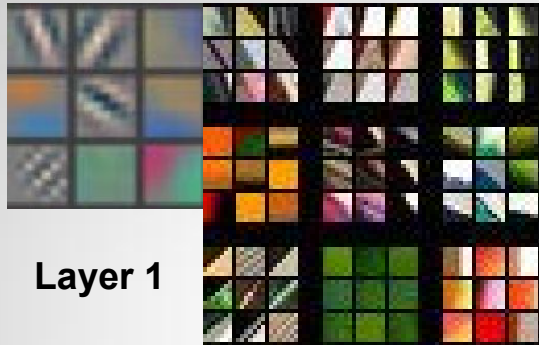


Filtering

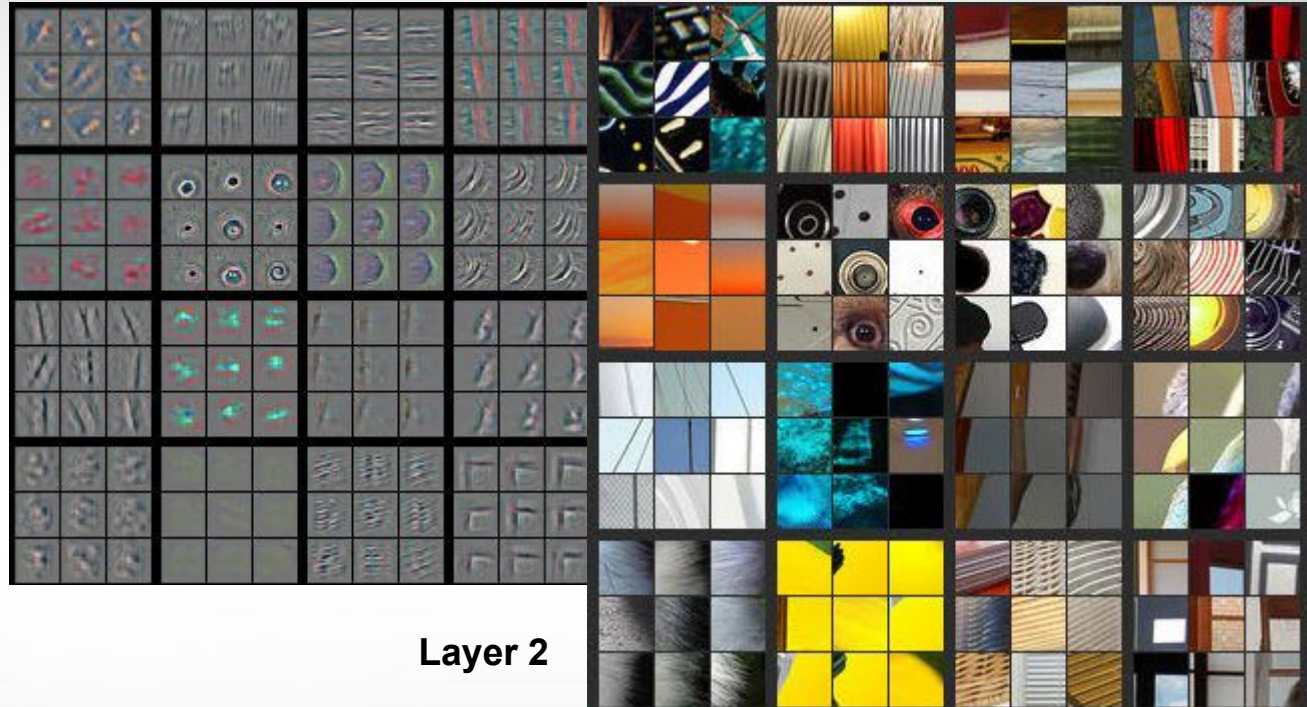
- ConvNet uses learned filters to convolve the feature maps from the previous layer
- DeconvNet uses 'transposed' filters applied to the rectified unpooled maps to reconstruct features
- Transposed filters are simply flipped along both axes before being applied



Examples



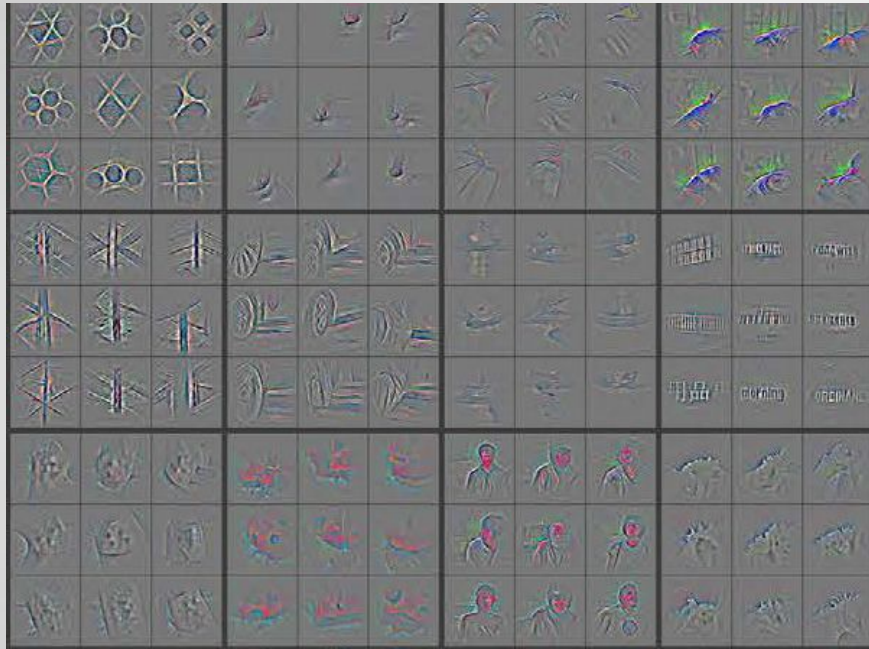
Layer 1



Layer 2



Examples



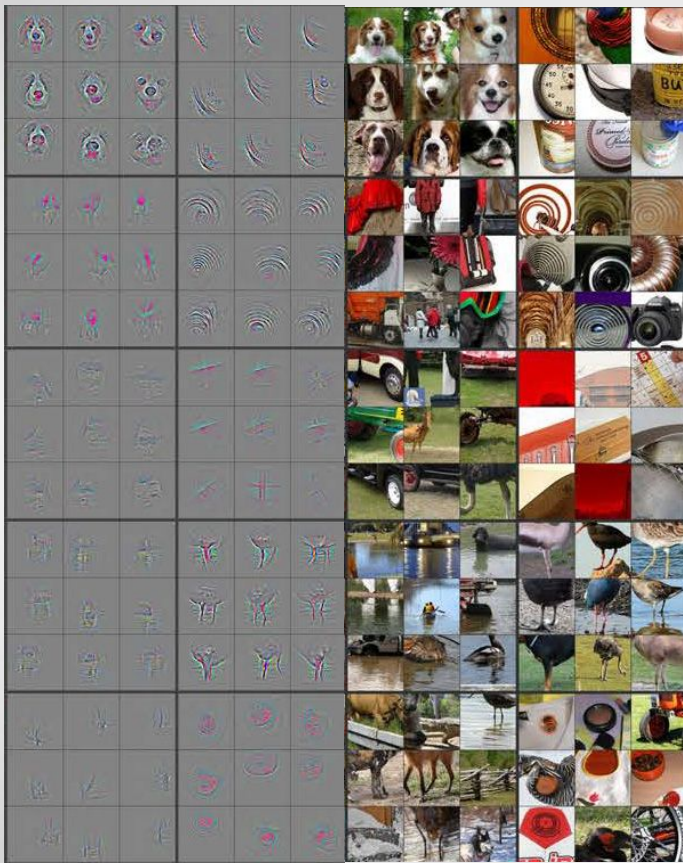
Layer 3



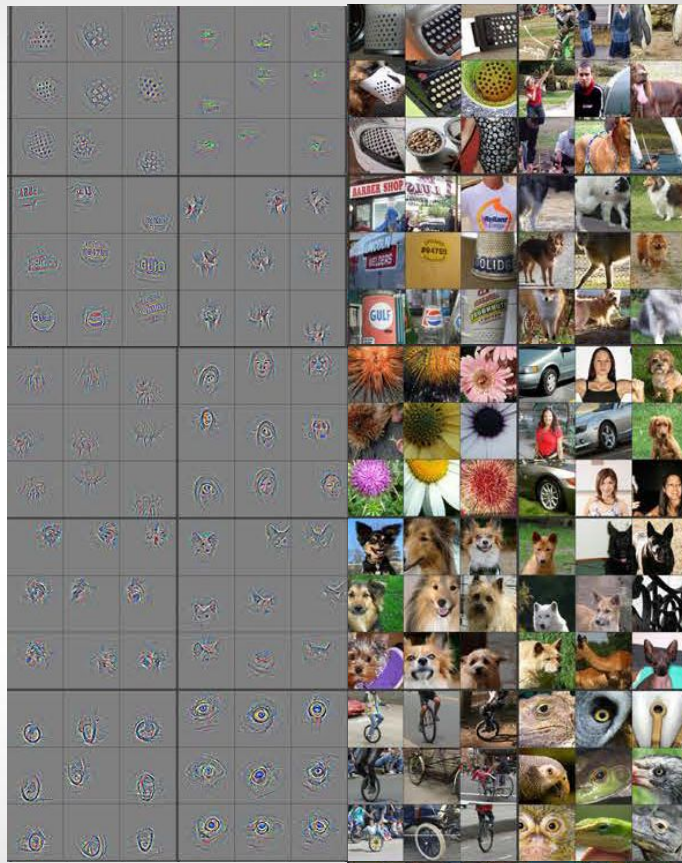


Examples

Layer 4

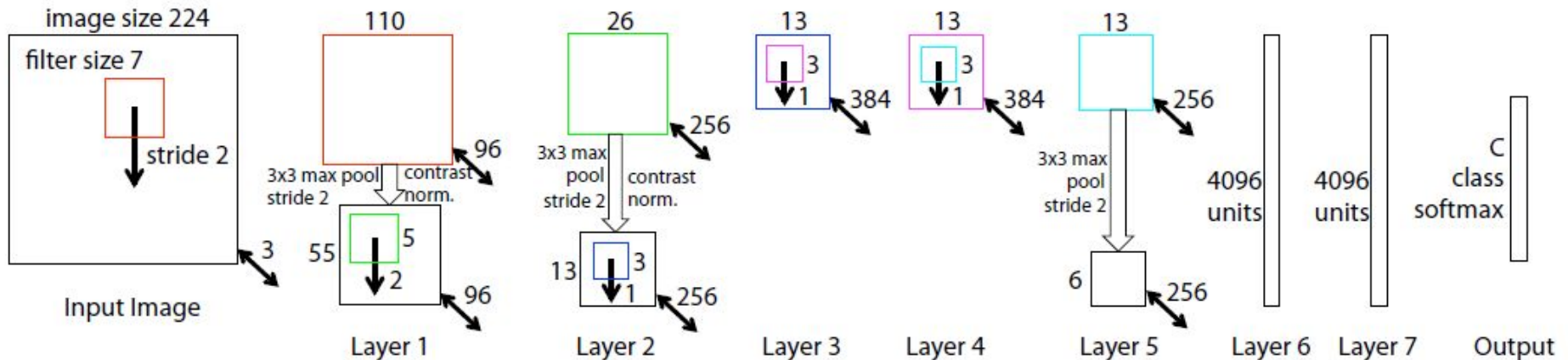


Layer 5



CNN Architecture

- Based off of the CNN we saw last week
 - Stride reduced
 - No split for two GPUs





Experiments

- Comparison against 2012 ImageNet Champion
- Architectural changes using 2012 ImageNet
- Occlusion Tolerance
- Feature Generalization
 - CalTech-101
 - CalTech-256
 - PASCAL 2012
- Feature Complexity Analysis



Results: VS 2012 ImageNet Champion

- Reimplemented 2012 ImageNet champion and compared results in several different setups.
- Improved CNN soundly beats the 2012 ImageNet champion with both one, and multiple convnet models.

(It did not beat 2013 champion, however.)

Error %	Val Top-1	Val Top-5	Test Top-5
Gunji <i>et al.</i> [12]	-	-	26.2
DeCAF [7]	-	-	19.2
Krizhevsky <i>et al.</i> [18], 1 convnet	40.7	18.2	--
Krizhevsky <i>et al.</i> [18], 5 convnets	38.1	16.4	16.4
Krizhevsky <i>et al.</i> * [18], 1 convnets	39.0	16.6	--
Krizhevsky <i>et al.</i> * [18], 7 convnets	36.7	15.4	15.3
Our replication of Krizhevsky <i>et al.</i> , 1 convnet	40.5	18.1	--
1 convnet as per Fig. 3	38.4	16.5	--
5 convnets as per Fig. 3 – (a)	36.7	15.3	15.3
1 convnet as per Fig. 3 but with layers 3,4,5: 512,1024,512 maps – (b)	37.5	16.0	16.1
6 convnets, (a) & (b) combined	36.0	14.7	14.8
Howard [15]	-	-	13.5
Clarifai [28]	-	-	11.7



Results: Architectural changes

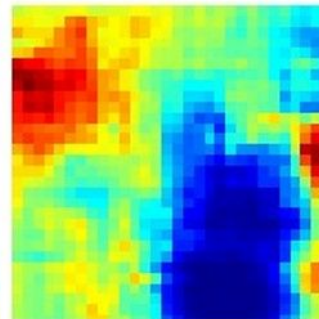
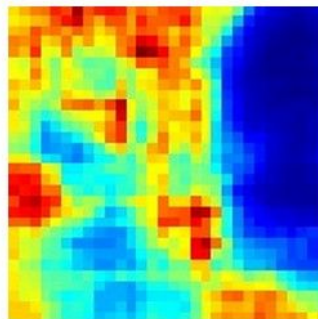
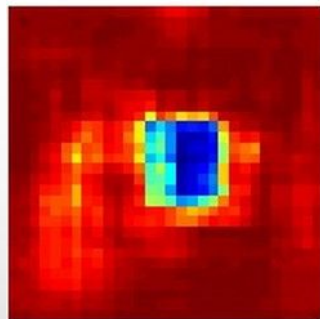
- Removing fully connected layers (layers 6-7) only dropped performance slightly
- Similarly, dropping two convolutional layers had only a moderate loss of performance
- Removing both, however, severely degrades performance
- Increasing size of fully-connected layers is slightly better
- Increasing size of convolutional layers is much better
- Increasing both results in overfitting

Error %	Train Top-1	Val Top-1	Val Top-5
Our replication of Krizhevsky <i>et al.</i> [18], 1 convnet	35.1	40.5	18.1
Removed layers 3,4	41.8	45.4	22.1
Removed layer 7	27.4	40.0	18.4
Removed layers 6,7	27.4	44.8	22.4
Removed layer 3,4,6,7	71.1	71.3	50.1
Adjust layers 6,7: 2048 units	40.3	41.7	18.8
Adjust layers 6,7: 8192 units	26.8	40.0	18.1
Our Model (as per Fig. 3)	33.1	38.4	16.5
Adjust layers 6,7: 2048 units	38.2	40.2	17.6
Adjust layers 6,7: 8192 units	22.0	38.8	17.0
Adjust layers 3,4,5: 512,1024,512 maps	18.8	37.5	16.0
Adjust layers 6,7: 8192 units and Layers 3,4,5: 512,1024,512 maps	10.0	38.3	16.9



Results: Occlusion Tolerance

- Creating a heat-map of correct classification based on occlusion location verifies object identification, rather than contextual identification.





Results: Feature Generalization

- Caltech-101 dataset
 - 9,000 images
 - 101 categories
 - Retrained softmax layer using 30 images/class
 - Beat best reported result by 2.2%!

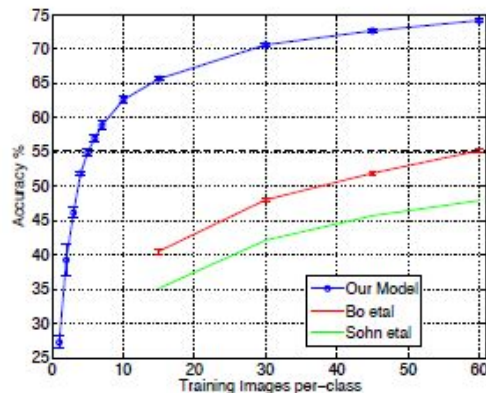
# Train	Acc % 15/class	Acc % 30/class
Bo <i>et al.</i> [3]	—	81.4 ± 0.33
Yang <i>et al.</i> [17]	73.2	84.3
Non-pretrained convnet	22.8 ± 1.5	46.5 ± 1.7
ImageNet-pretrained convnet	83.8 ± 0.5	86.5 ± 0.5



Results: Feature Generalization

- Caltech-256 dataset
 - 30,000 images
 - 256 categories
 - Retrained softmax using 60 images/class
 - Beats best reported solution by 19%!

# Train	Acc % 15/class	Acc % 30/class	Acc % 45/class	Acc % 60/class
Sohn <i>et al.</i> [24]	35.1	42.1	45.7	47.9
Bo <i>et al.</i> [3]	40.5 ± 0.4	48.0 ± 0.2	51.9 ± 0.2	55.2 ± 0.3
Non-pretr.	9.0 ± 1.4	22.5 ± 0.7	31.2 ± 0.5	38.8 ± 1.4
ImageNet-pretr.	65.7 ± 0.2	70.6 ± 0.2	72.7 ± 0.4	74.2 ± 0.3





Results: Feature Generalization

- PASCAL 2012 dataset

- 11,500 images
- 27,000 objects
- 20 categories
- Full-scene images (multiple objects)

Acc %	[22]	[27]	[21]	Ours	Acc %	[22]	[27]	[21]	Ours
Airplane	92.0	97.3	94.6	96.0	Dining table	63.2	77.8	69.0	67.7
Bicycle	74.2	84.2	82.9	77.1	Dog	68.9	83.0	92.1	87.8
Bird	73.0	80.8	88.2	88.4	Horse	78.2	87.5	93.4	86.0
Boat	77.5	85.3	60.3	85.5	Motorbike	81.0	90.1	88.6	85.1
Bottle	54.3	60.8	60.3	55.8	Person	91.6	95.0	96.1	90.9
Bus	85.2	89.9	89.0	85.8	Potted plant	55.9	57.8	64.3	52.2
Car	81.9	86.8	84.4	78.6	Sheep	69.4	79.2	86.6	83.6
Cat	76.4	89.3	90.7	91.2	Sofa	65.4	73.4	62.3	61.1
Chair	65.2	75.4	72.1	65.0	Train	86.7	94.5	91.1	91.8
Cow	63.2	77.8	86.8	74.4	Tv	77.4	80.7	79.8	76.1
Mean	74.3	82.2	82.8	79.0	# won	0	11	6	3

- Best performance on 5 categories. Mean performance 3.2% lower than best reported result.



Results: Feature Complexity Analysis

- By varying the number of convnet layers we can observe how discriminative the learned features are
- As the number of layers increases, accuracy improves
- Supports the idea that deeper features are increasingly complex/powerful

	Cal-101 (30/class)	Cal-256 (60/class)
SVM (1)	44.8 ± 0.7	24.6 ± 0.4
SVM (2)	66.2 ± 0.5	39.6 ± 0.3
SVM (3)	72.3 ± 0.4	46.0 ± 0.3
SVM (4)	76.6 ± 0.4	51.3 ± 0.1
SVM (5)	86.2 ± 0.8	65.6 ± 0.3
SVM (7)	85.5 ± 0.4	71.7 ± 0.2
Softmax (5)	82.9 ± 0.4	65.7 ± 0.5
Softmax (7)	85.4 ± 0.4	72.6 ± 0.1



Related Work

- Applied CNNs to Places dataset (scene recognition)
- Reliable object detectors emerge, **completely unsupervised**
- Interesting applications:
 - Minimal representation
 - Visualizing receptive fields
 - Object localization/segmentation
 - Semantic meaning of RFs

Zhou, Bolei, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. "Object detectors emerge in deep scene cnns." arXiv preprint arXiv:1412.6856 (2014).

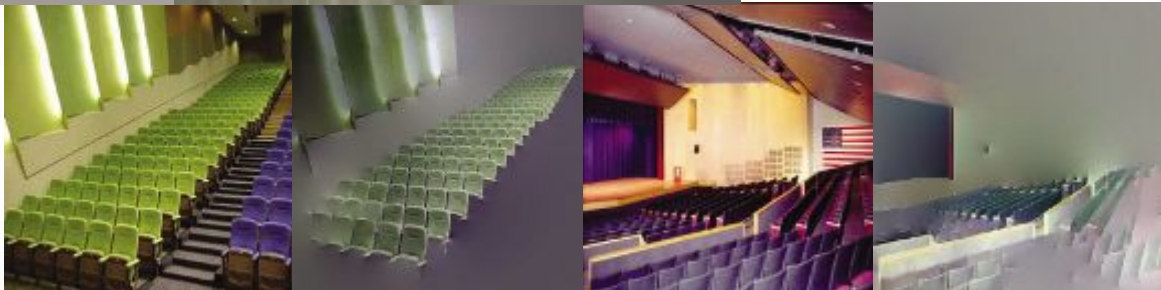


Minimal Image Representation

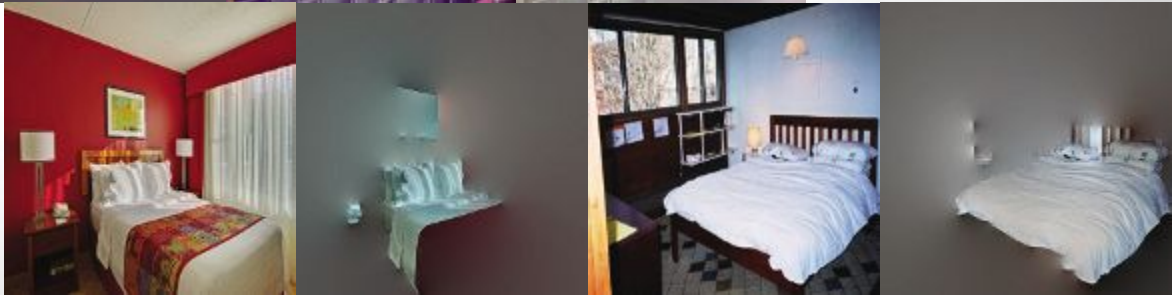


Art Gallery
Paintings (81%)
Pictures (58%)

Auditorium
(Details not reported)

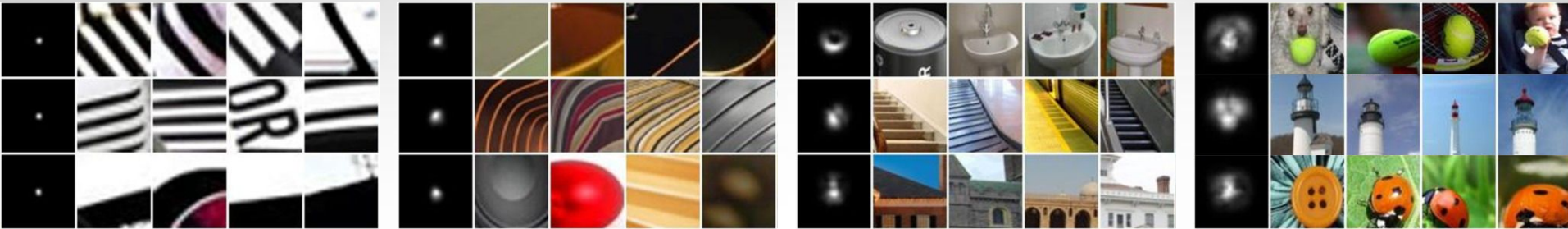


Bedroom
Bed (87%)
Wall (28%)
Window (21%)





Visualizing Receptive Fields



- Generated using a sliding-window stimuli
 - Similar to occlusion tolerance shown previously
- Generates a “discrepancy map” per image
- Sum of calibrated discrepancy maps reveals the RF



Object Localization/Segmentation

Extremely valuable visualizations!

- Gives per-layer insight into why the CNN activated that way
- Demonstrates increasing feature complexity
- Visualizes what object/parts were learned

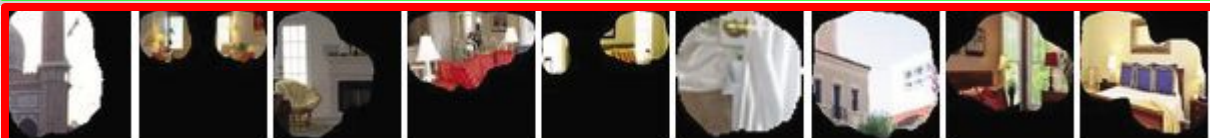
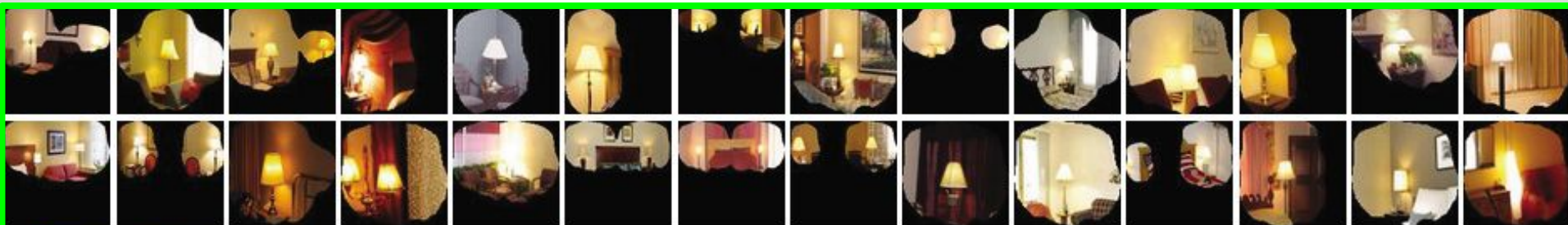




Semantic Meaning of RFs



Pool5, unit 76; label: ocean; Type: scene; Precision: 93%

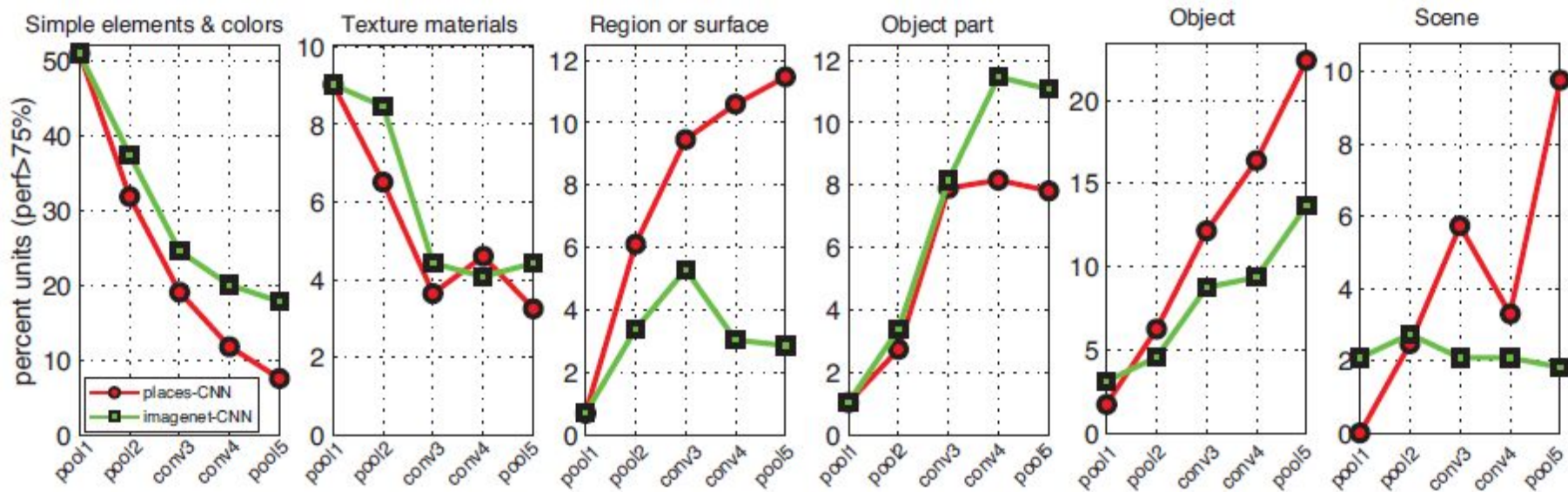


Pool5, unit 13; label: lamps;
Type: object; Precision: 84%



Semantic Meaning of RFs

- Feature complexity is strongly correlated to layer
 - Minor exception: ImageNet has very few “scene” images





Conclusion

- Introduced novel visualization of feature activation
- Demonstrated how the visualization helps identify problems in model architecture
- Verified object classification using occlusion experiments
- Showed that increased CNN depth increases feature complexity
- Demonstrated that trained models can generalize well to other datasets, often beating reported results by large margins



Strengths

- Re-implemented source work for detailed comparison
- Multiple avenues of experimentation
- Significant work to verify results
 - Occlusion tests
 - Feature generalizability
 - Architecture and depth modifications



Weaknesses

- No significant improvements in any single area
 - Only pursued marginal benefits of proposed visualization
- Poor (low-resolution) visualizations hurt overall clarity
 - Limited by submission guidelines, but a paper about a novel visualization technique should have some larger/clearer visualizations



Overall Rating

My Rating (scale 0-5): **1**

- Huge impact for future researchers
- Extensive work and validation of results
- Clear improvements to state-of-the-art solutions
- Limited visualization clarity and performance gains



Future Directions

- Combine with data-augmentation work of A.G. Howard
- Pursue additional performance gains exposed by the novel visualization technique
- Incorporate other cutting-edge CNN techniques to improve scores on modern benchmarks
- Explore additional visualizations not limited to a single image activation



Any Questions?

...Thanks for listening!