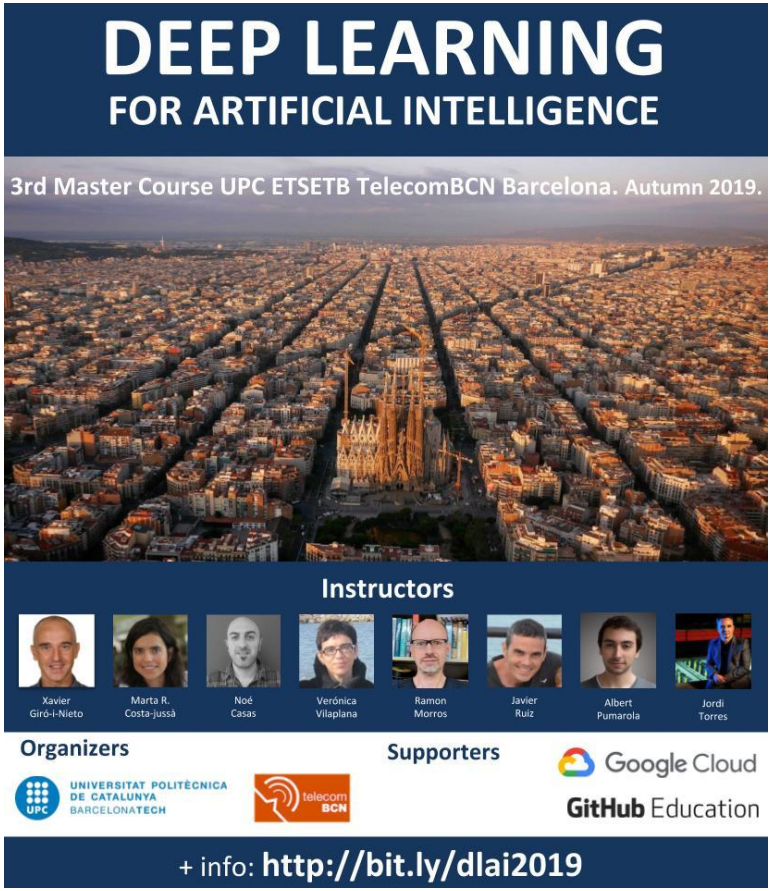


[course site]

#DLUPC

Day 7 Lecture 1

Interpretability

Xavier Giro-i-Nieto
xavier.giro@upc.edu

Associate Professor
Universitat Politècnica de Catalunya
Technical University of Catalonia

Acknowledgements

Amaia Salvador
amaia.salvador@upc.edu

PhD Candidate
Universitat Politècnica de Catalunya

UNIVERSITAT POLITÈCNICA DE CATALUNYA

Eva Mohedano
eva.mohedano@insight-centre.org

Postdoctoral Researcher
Insight-centre for Data Analytics
Dublin City University

DCU

Insight
Centre for Data Analytics

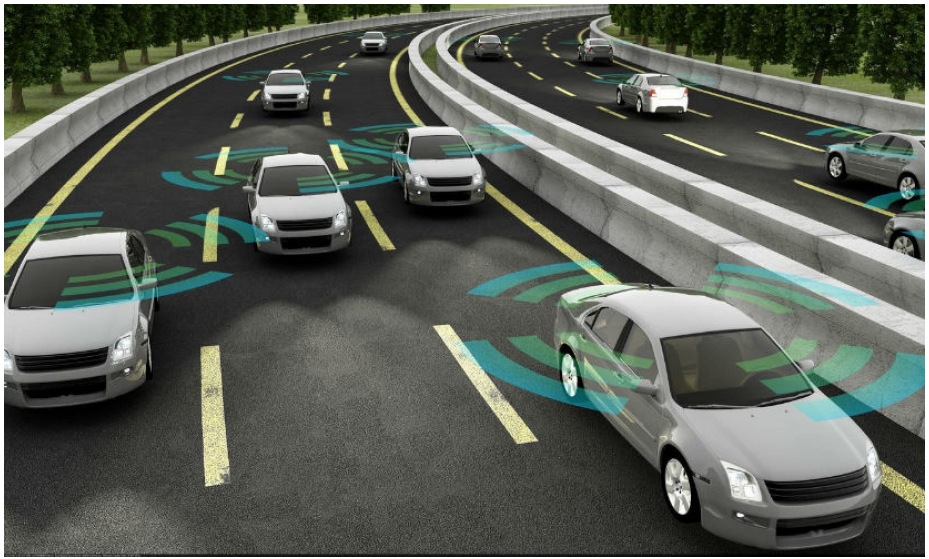
Videolectures

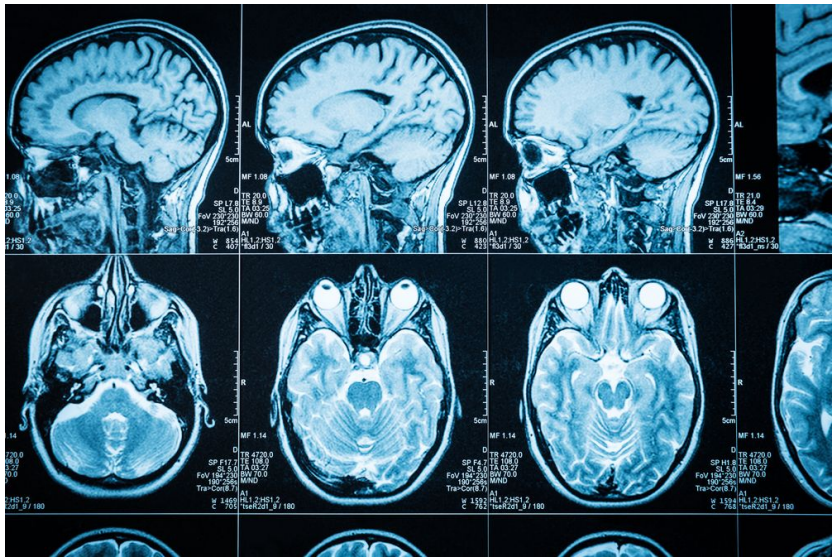
[UPC DLCV 2016]

[UPC DLCV 2018]

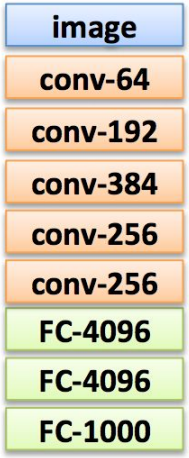
Motivation

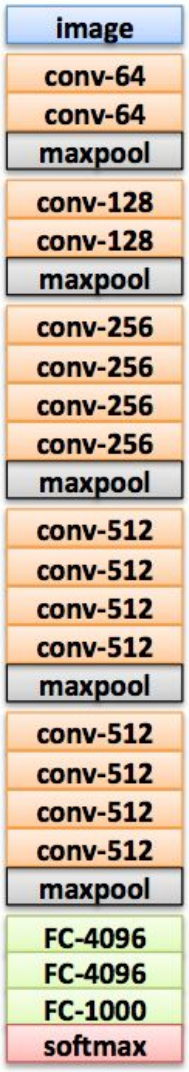
In many cases, an explanation for NN prediction is required.

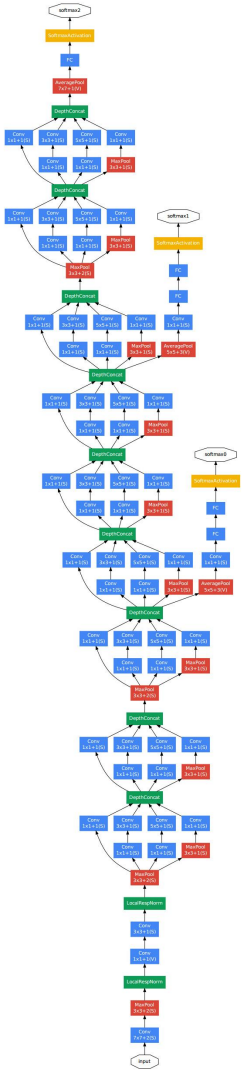


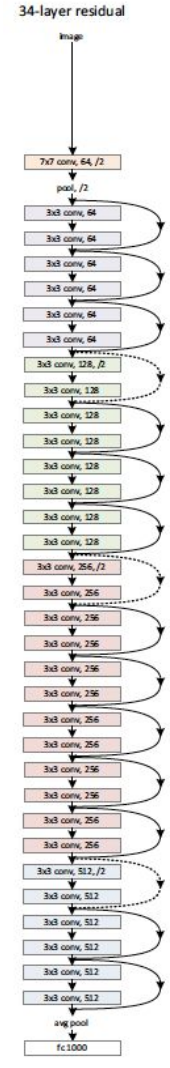


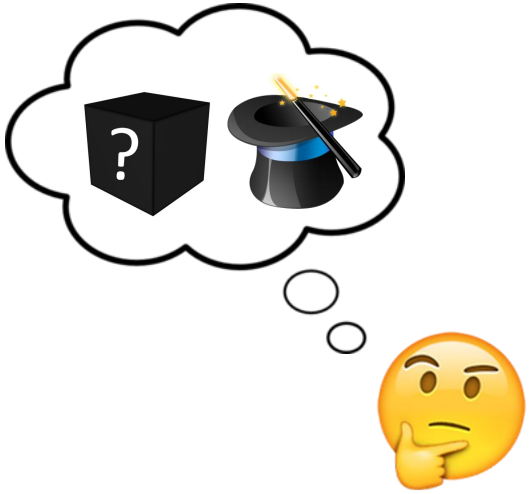
AlexNet



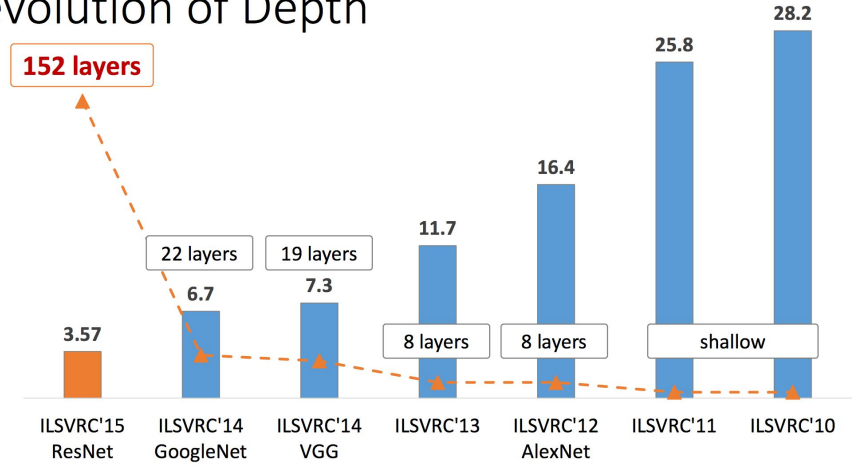








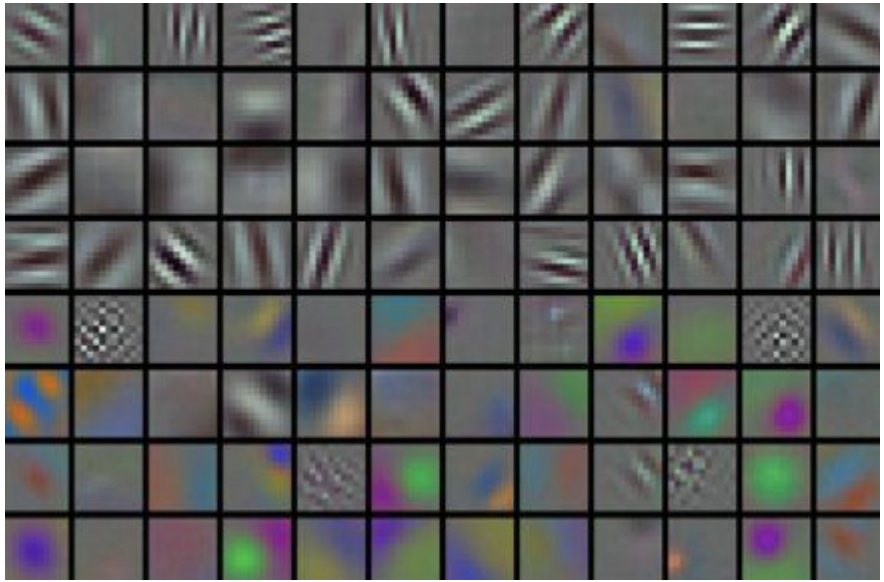
Revolution of Depth

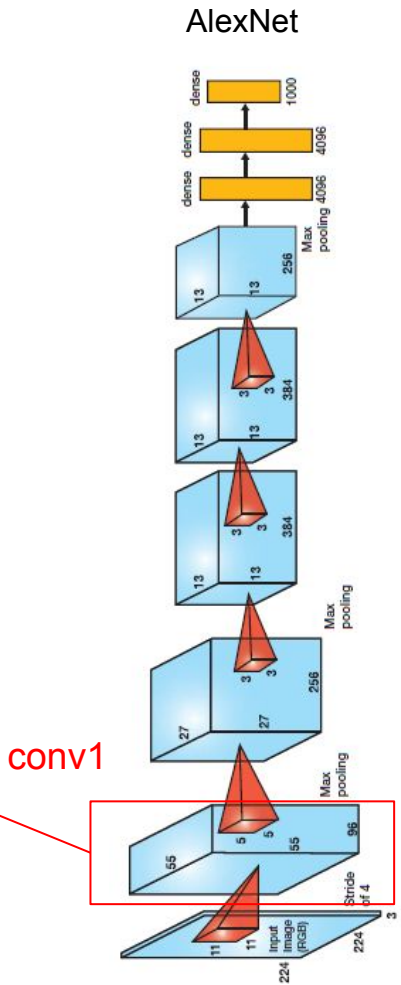


Interpretability

- **Visualization**
  - **Learned Weights**
  - Feature Maps
- Attribution
- Feature visualization

Visualization of Learned Weights

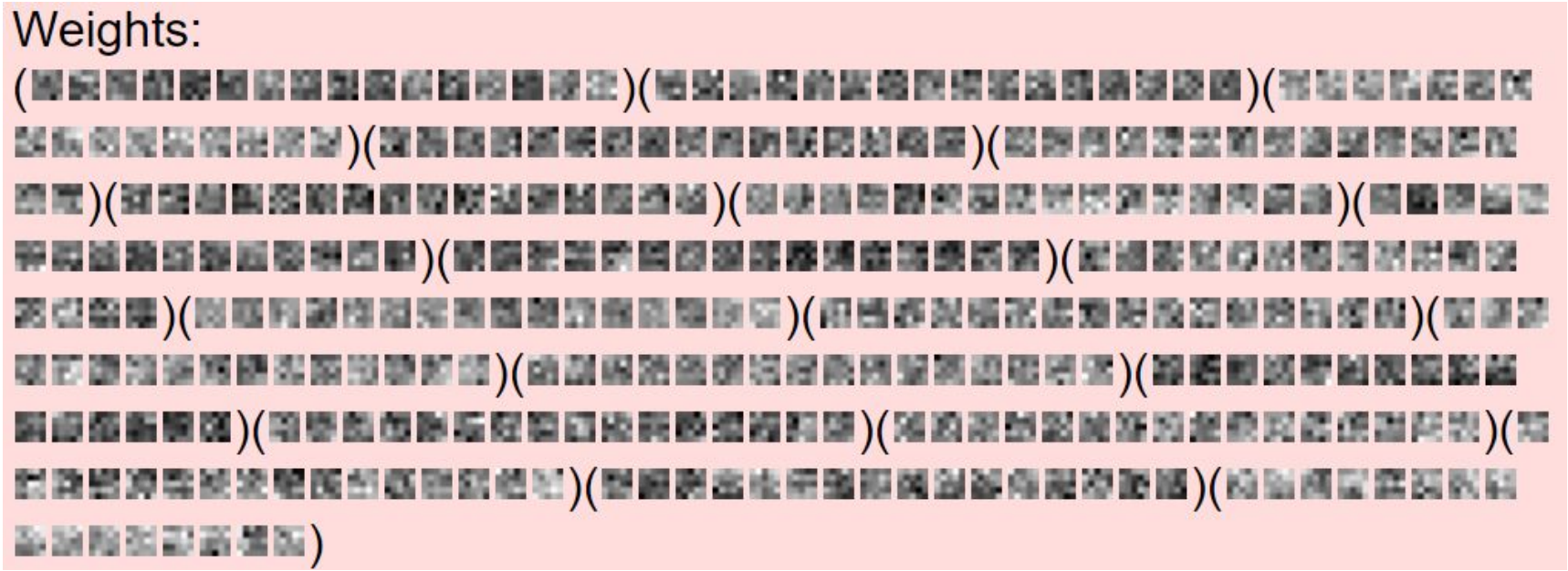




Only convolutional filters from the first layer can be “visualized”, because their depth of 3 matches the input RGB channels.

Visualization of Learned Weights

Filters with depth larger than 3 can be visualized showing a gray-scale image of each depth, one next to the other.



2D Convolutional filters learned in Layer #2 from ConvnetJS

Visualization of Learned Weights

Demo: [Classify MNIST digits with a Convolutional Neural Network]

ConvNetJS

Deep Learning in your browser

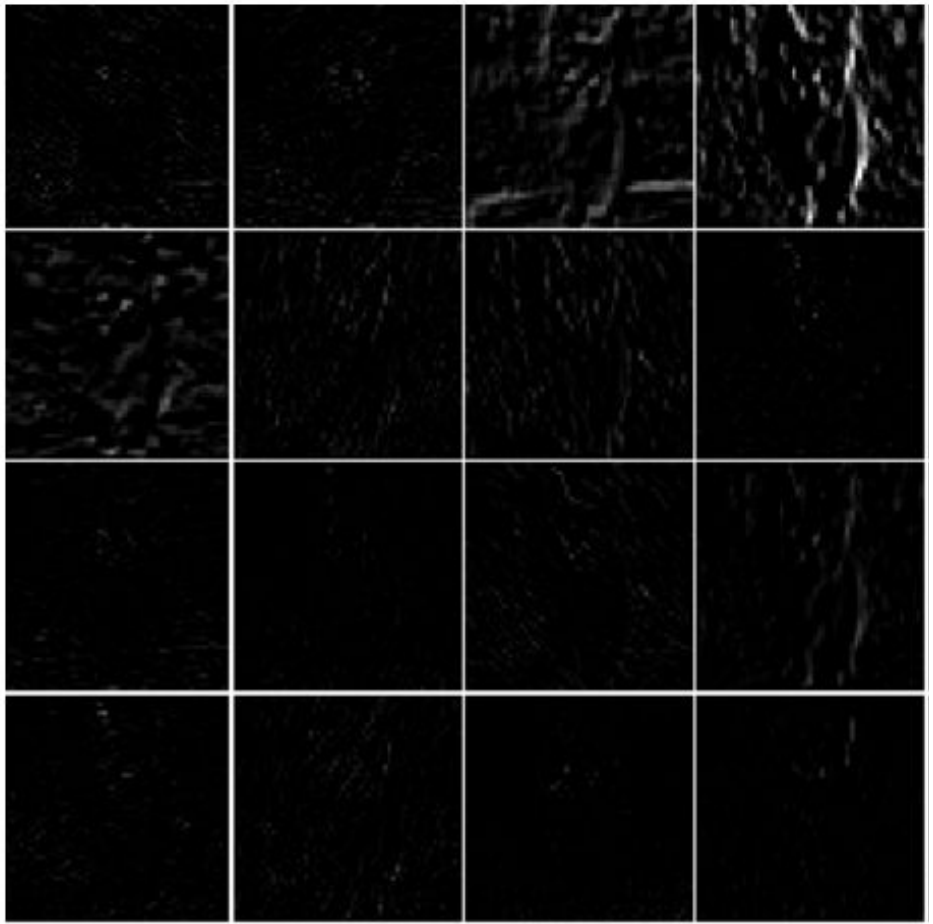
“ConvNetJS is a Javascript library for training Deep Learning models (mainly Neural Networks) entirely in your browser. Open a tab and you're training. No software requirements, no compilers, no installations, no GPUs, no sweat.”

Interpretability

- **Visualization**
  - Learned Weights
  - **Feature Maps**
- Attribution
- Feature visualization

Visualization of Feature Activations (2D)

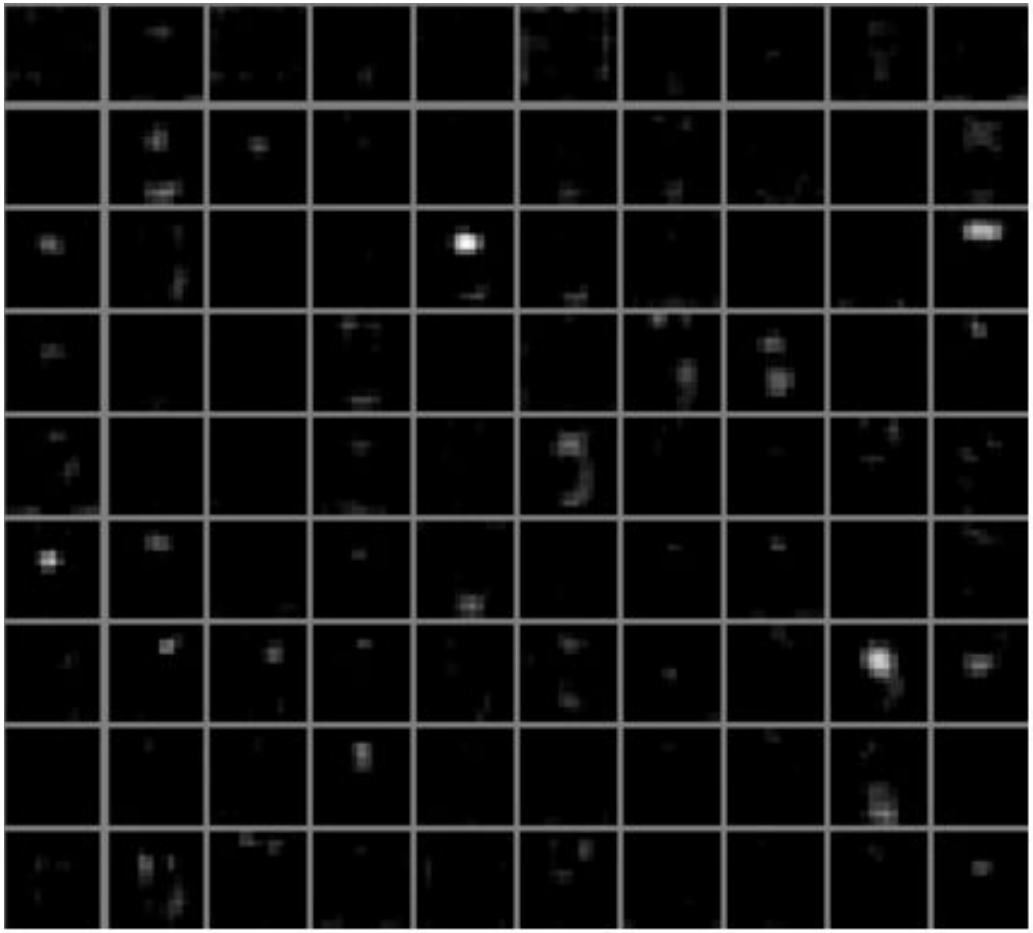
Input
image



conv1

Visualization of Feature Activations (2D)

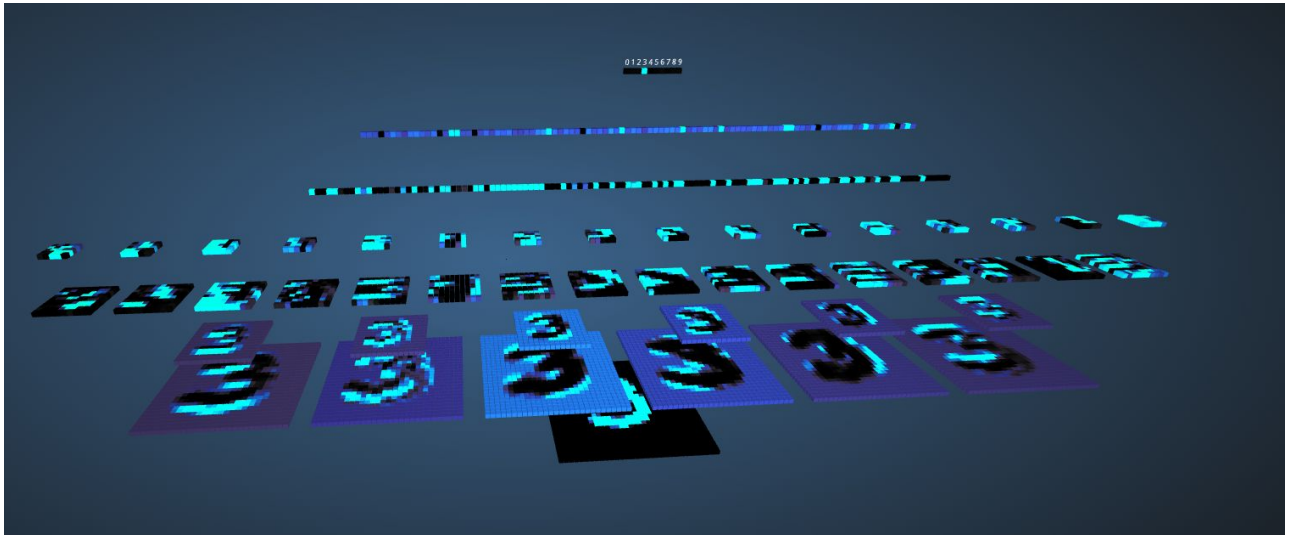
Input
image



conv5

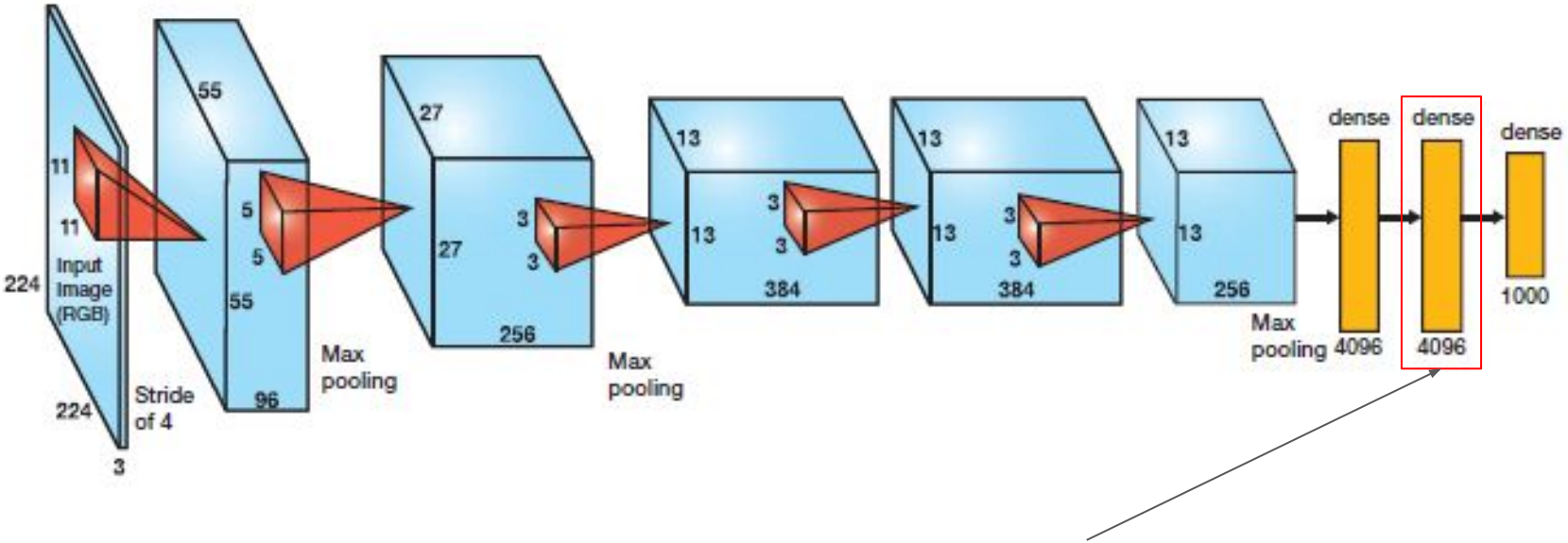
Visualization of Feature Activations (2D)

Demo: 3D Visualization of a Convolutional Neural Network



Harley, Adam W. "An Interactive Node-Link Visualization of Convolutional Neural Networks." In Advances in Visual Computing, pp. 867-877. Springer International Publishing, 2015.

Visualization of Feature Activations (any dimension)

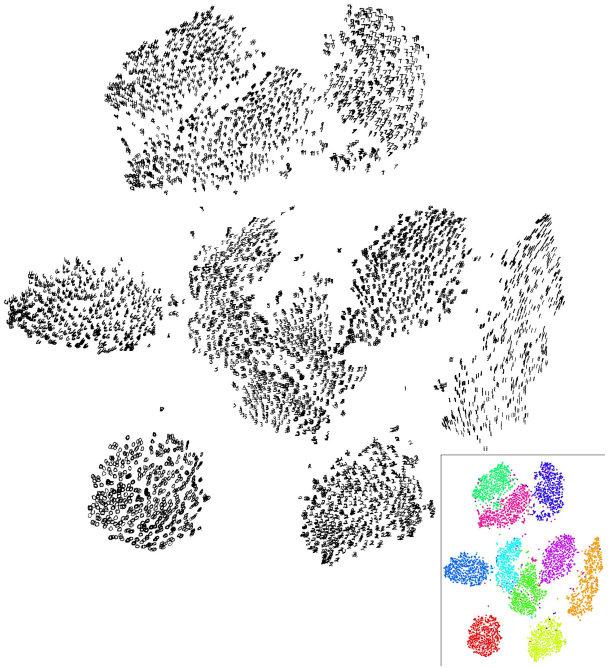


Can be used with features from layer before classification

Visualization of Feature Activations (any dimension)

t-SNE:
Embeds high dimensional data points (i.e. feature maps) so that pairwise distances are preserved in a local 2D neighborhoods.

Example: 10 classes from MNIST dataset



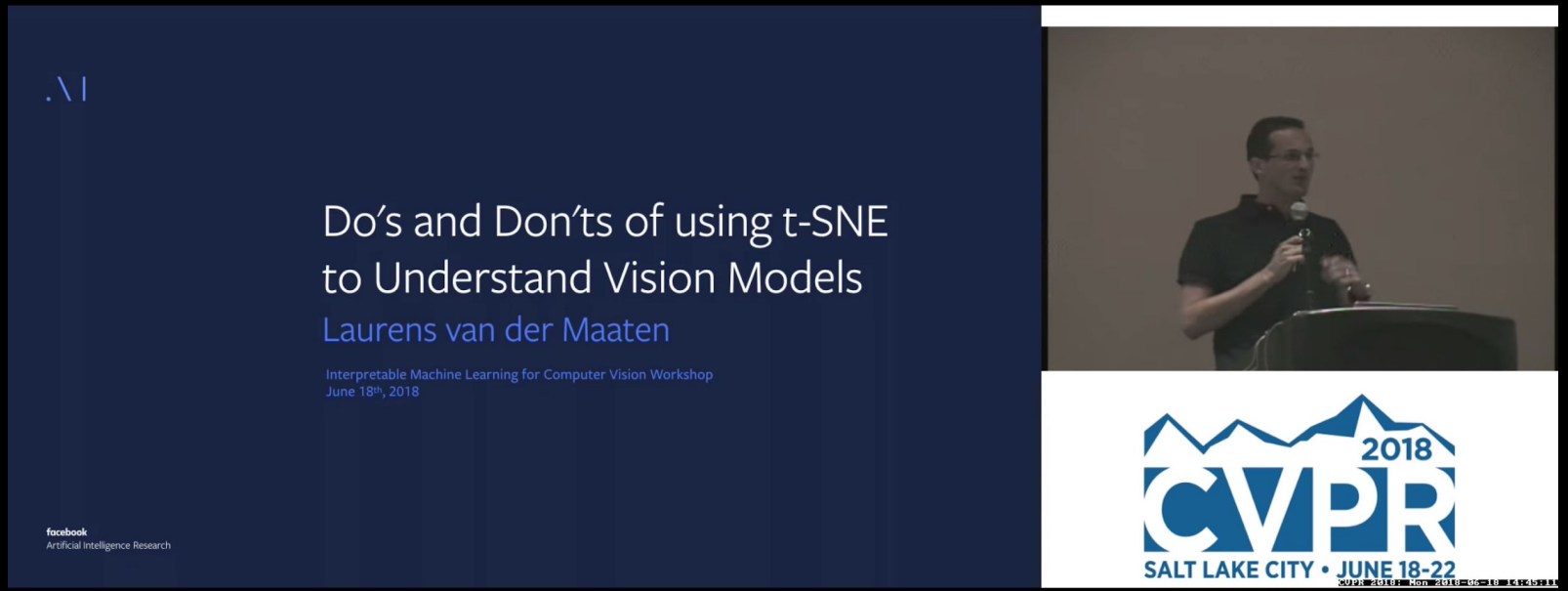
#t-sne Maaten, Laurens van der, and Geoffrey Hinton. "Visualizing data using t-SNE." Journal of machine learning research 9, no. Nov (2008): 2579-2605. [Python scikit-learn]

Visualization of Feature Activations (any dimension)

t-SNE on fc7 features from AlexNet.

Source: Andrey Karpath (Stanford University)

#t-sne Maaten, Laurens van der, and Geoffrey Hinton. "Visualizing data using t-SNE." Journal of machine learning research 9, no. Nov (2008): 2579-2605. [Python scikit-learn]



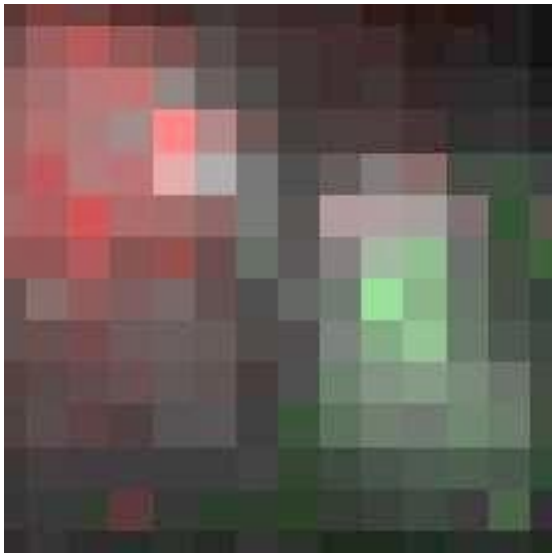
Maaten, Laurens van der @ CVPR 2018 Tutorial on Interpretable Machine Learning for Computer Vision.

Interpretability: Attribution

- Visualization
- **Attribution**
  - Receptive Field of the Highest Activations
  - Activation Changes after Occlusions
  - Class Activation Maps (CAMs)
  - Gradient-based
- Feature visualization

Attribution

Attribution studies what part of an example is responsible for the network activating a particular way.

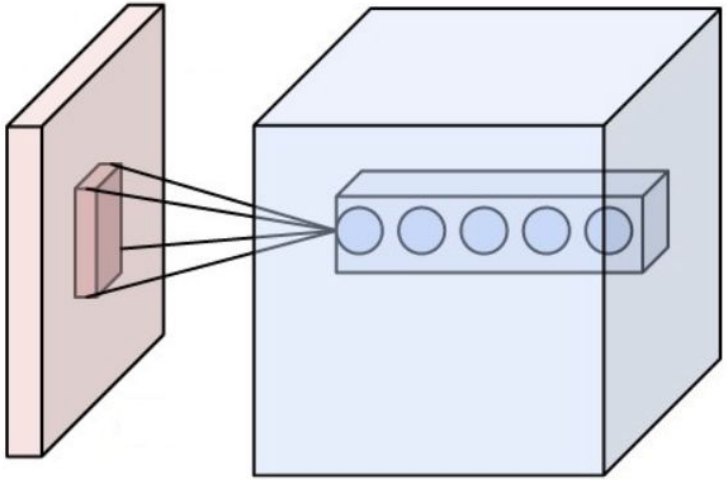


Chris Olah, Alexander Mordvintsev, Ludwig Schubert, "Feature Visualization". Distill.pub (November 2017)

Interpretability: Attribution

- Visualization
- **Attribution**
  - **Receptive Field of the Highest Activations**
  - Activation Changes after Occlusions
  - Class Activation Maps (CAMs)
  - Gradient-based
- Feature visualization

Reminder: Receptive Field



Receptive field: Part of the input that is visible to a neuron. It increases as we stack more convolutional layers (i.e. neurons in deeper layers have larger receptive fields).

Receptive Field of Highest Activations

Visualize the receptive field of a neuron on those images that activate it the most

Girshick et al. Rich feature hierarchies for accurate object detection and semantic segmentation. CVPR 2014

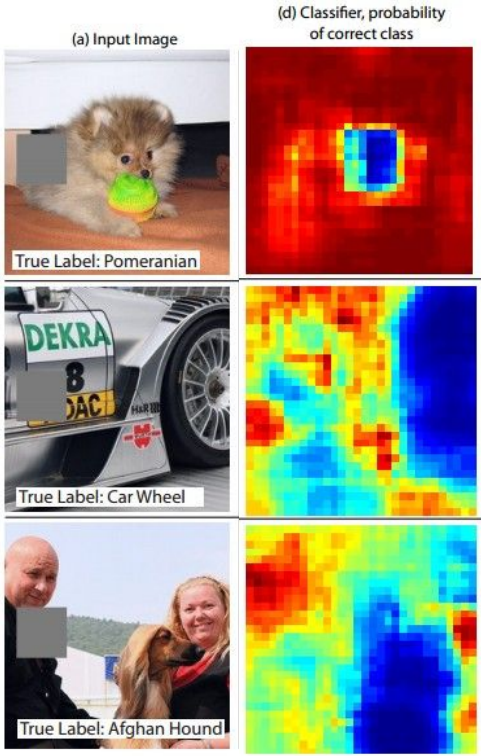
Interpretability: Attribution

- Visualization
- **Attribution**
  - Receptive Field of the Highest Activations
  - **Activation Changes after Occlusions**
  - Class Activation Maps (CAMs)
  - Gradient-based
- Feature visualization

Activation Changes after Occlusions (global scale)

1. Iteratively forward the same image through the network, occluding a different region at a time.

2. Keep track of the probability of the correct class w.r.t. the position of the occluder



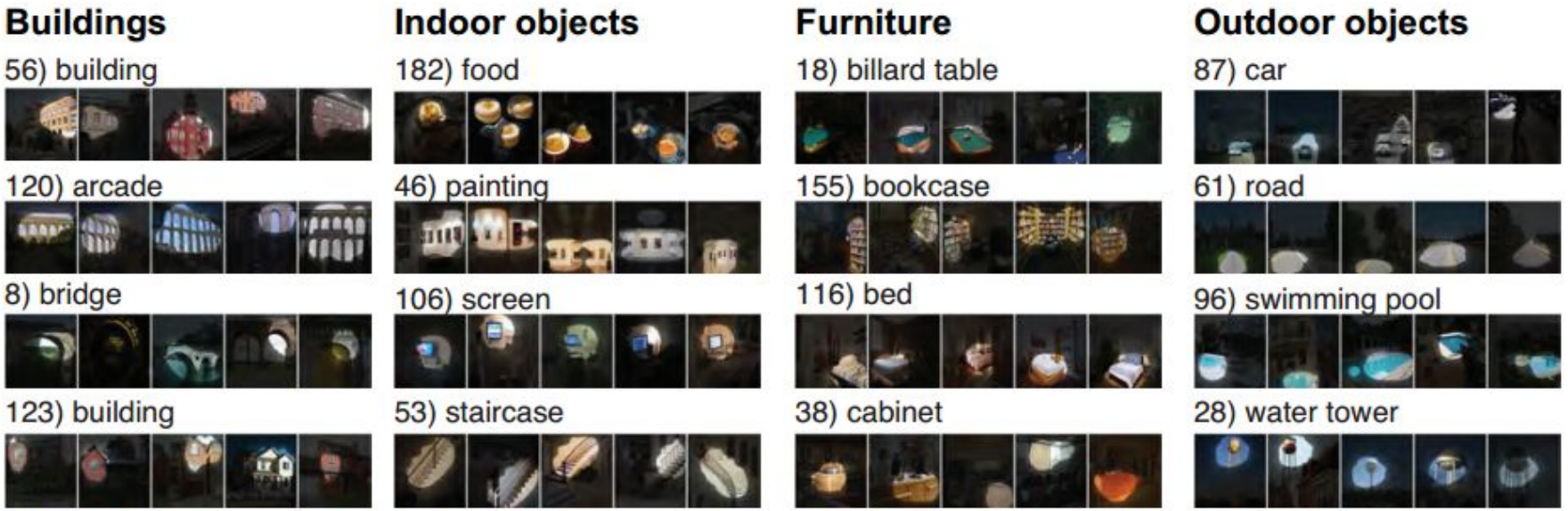
Zeiler and Fergus. Visualizing and Understanding Convolutional Networks. ECCV 2015

Activation Changes after Occlusions (global scale)

Campos, V., Jou, B., & Giro-i-Nieto, X. . From Pixels to Sentiment: Fine-tuning CNNs for Visual Sentiment Prediction. Image and Vision Computing. (2017)

Activation Changes after Occlusions (global scale)

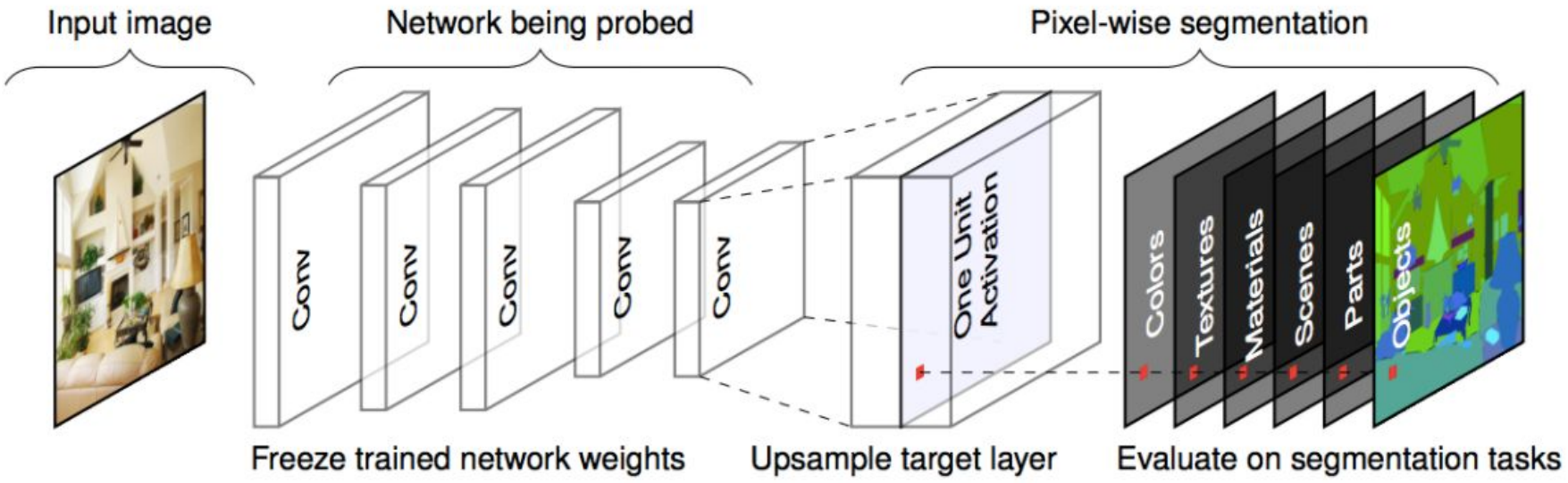
The changes in activations can be observed in any layer. This allowed identifying some filters as weak object detectors, trained with image labels.



Zhou, Bolei, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. "Object detectors emerge in deep scene CNNs." ICLR 2015.

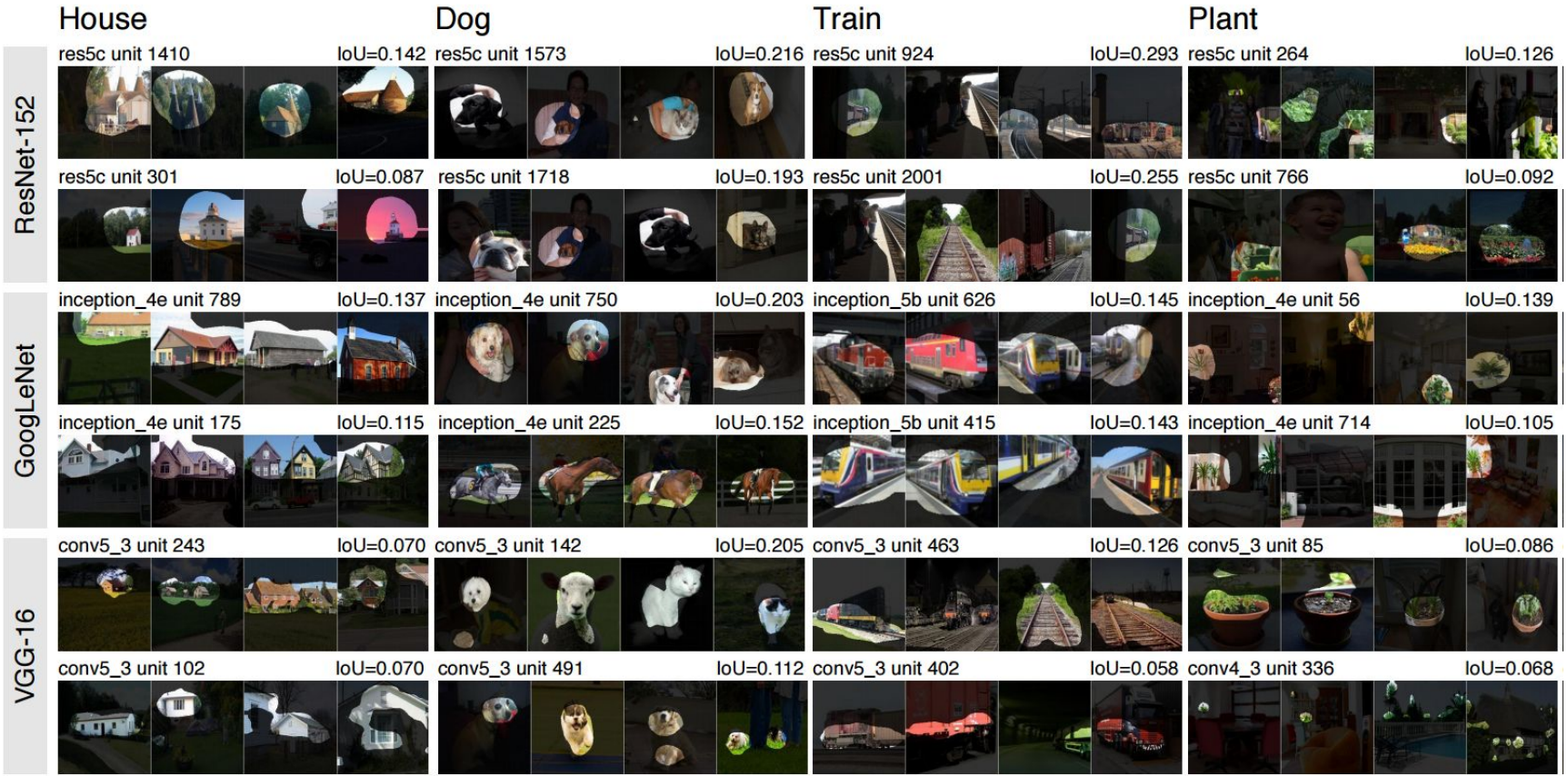
Activation Changes after Occlusions (pixel scale)

Same idea, but automatic unit labeling using densely labeled dataset (pixel-level annotations).
The thresholded activation of each conv unit in the network is evaluated for semantic segmentation.



Bau, Zhou et al. Network Dissection: Quantifying Interpretability of Deep Visual Representations. CVPR 2017

Activation Changes after Occlusions (pixel scale)



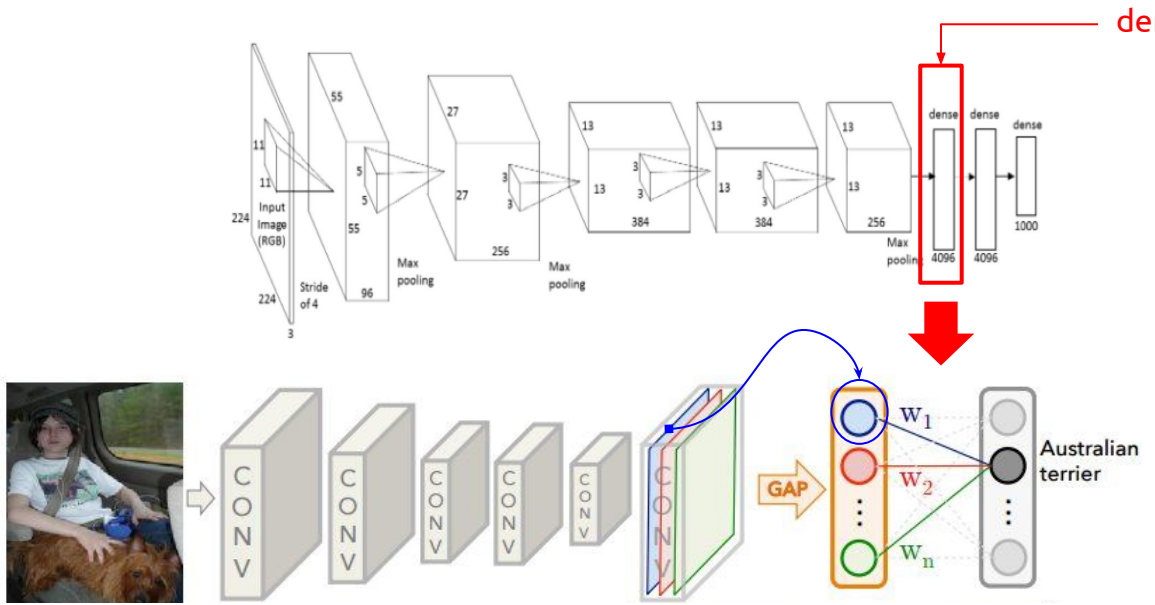
Bau, Zhou et al. Network Dissection: Quantifying Interpretability of Deep Visual Representations. CVPR 2017

Interpretability: Attribution

- Visualization
- **Attribution**
  - Receptive Field of the Highest Activations
  - Activation Changes after Occlusions
  - **Class Activation Maps (CAMs)**
  - Gradient-based
- Feature visualization

Class Activation Maps

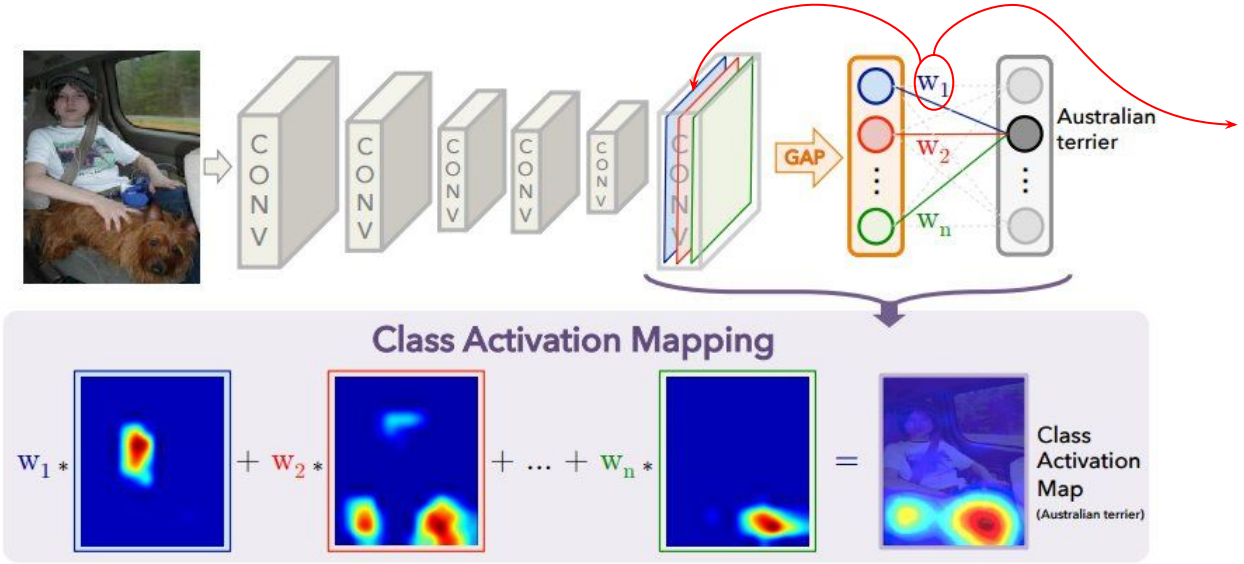
Change in a classic CNN+MLP architecture: Replace FC layer after last conv with Global Average Pooling (GAP), which corresponds to averaging per channel.



Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. "Learning deep features for discriminative localization." CVPR 2016

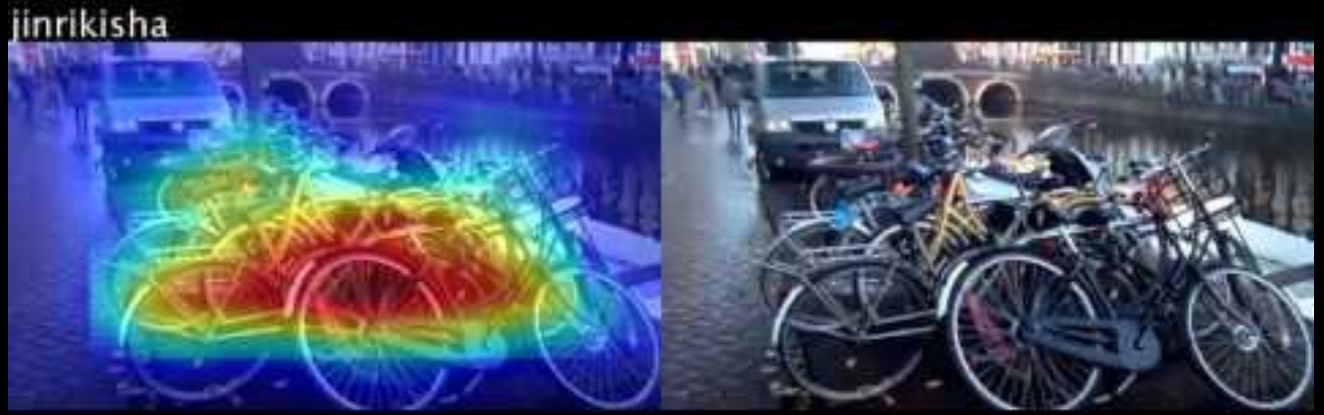
Class Activation Maps

Weighted Fusion of Feature Maps : The classifier weights define the contribution of each channel in the previous layer



contribution of the *blue* channel in the conv layer to the prediction of “australian terrier”

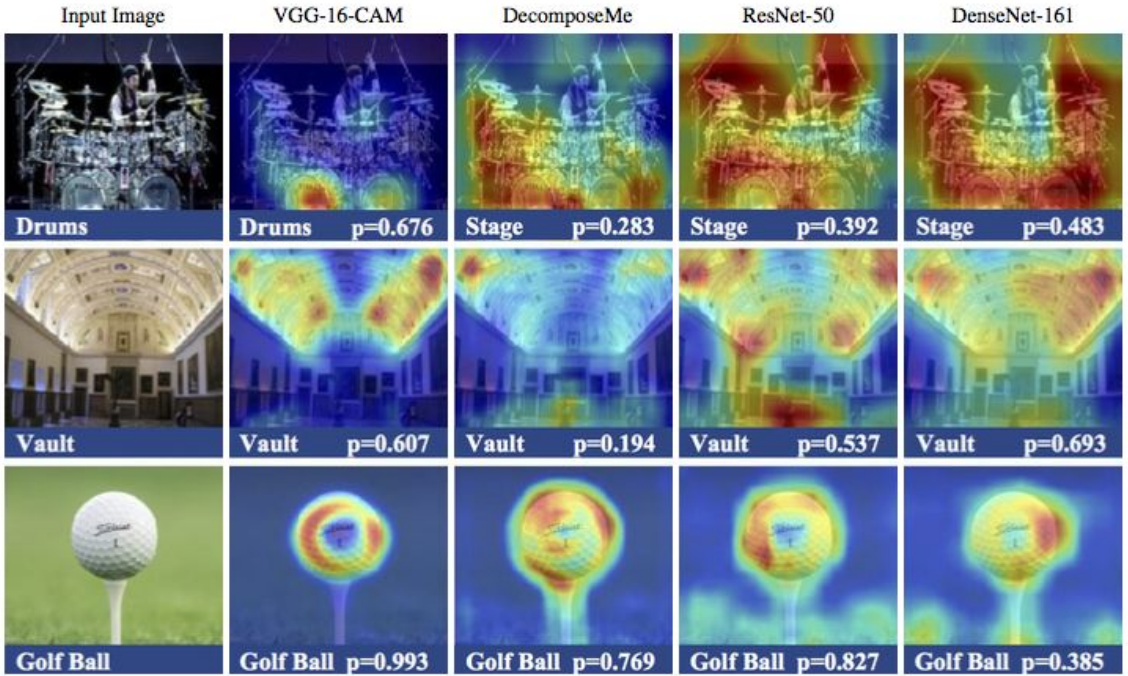
Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. "Learning deep features for discriminative localization." CVPR 2016



Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. "Learning deep features for discriminative localization." CVPR 2016

Class Activation Maps

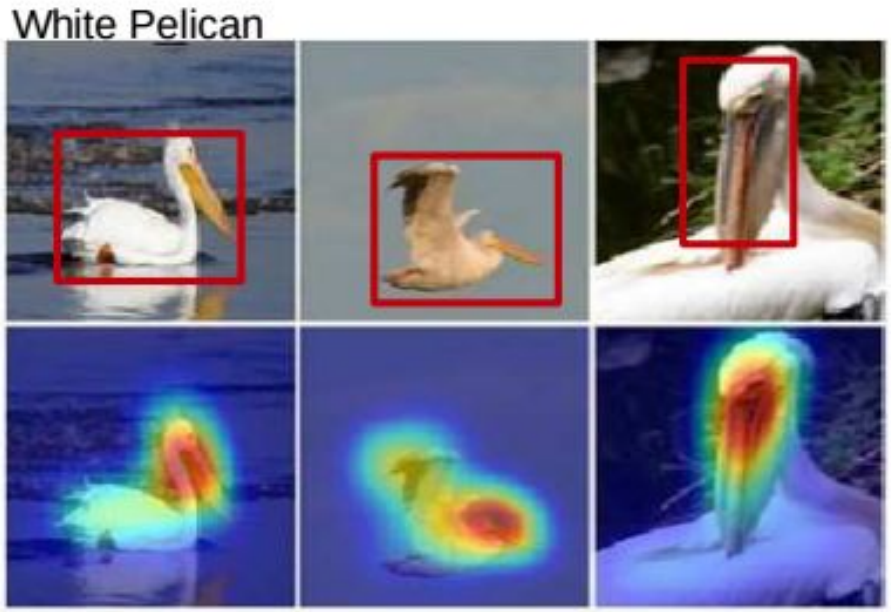
Different architectures generate different CAMs.

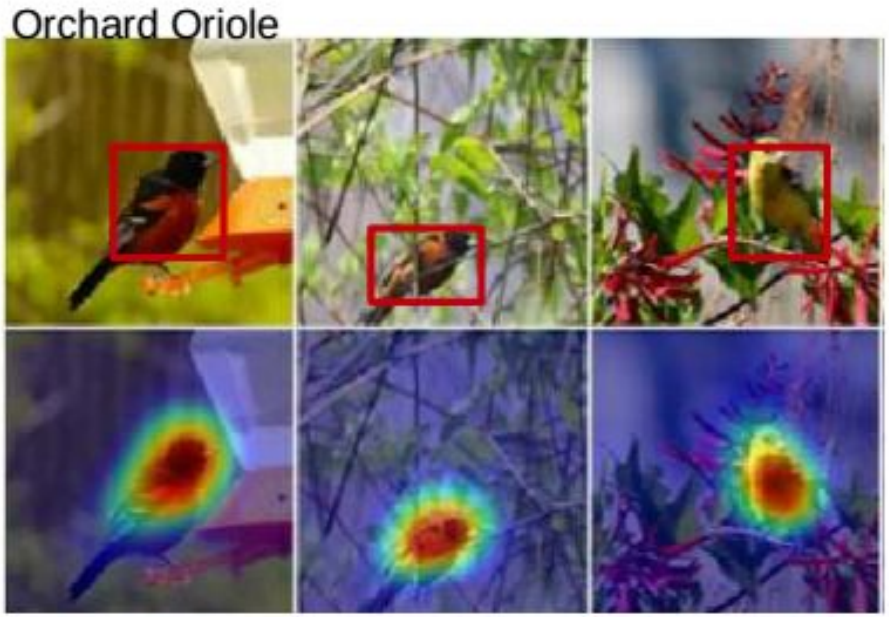


Jimenez, Albert, Jose M. Alvarez, and Xavier Giro-i-Nieto. "Class-weighted convolutional features for visual instance search." BMVC 2017.

Class Activation Maps

Applications of CAMs: Rough object localization without weak labels (image).





Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. "Learning deep features for discriminative localization." CVPR 2016

Class Activation Maps

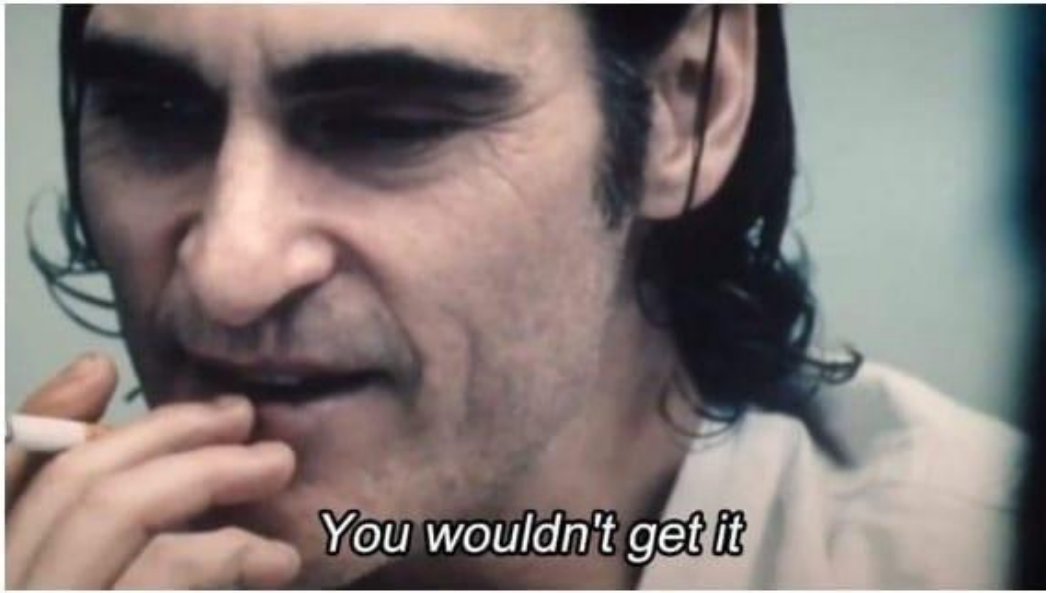
Applications of CAMs: Spatial feature weighting for object retrieval.

Jimenez, Albert, Jose M. Alvarez, and Xavier Giro-i-Nieto. "Class-weighted convolutional features for visual instance search." BMVC 2017.

Class Activation Maps

Me: using Class Activation Maps to understand where the model is looking when making the decision

Model:

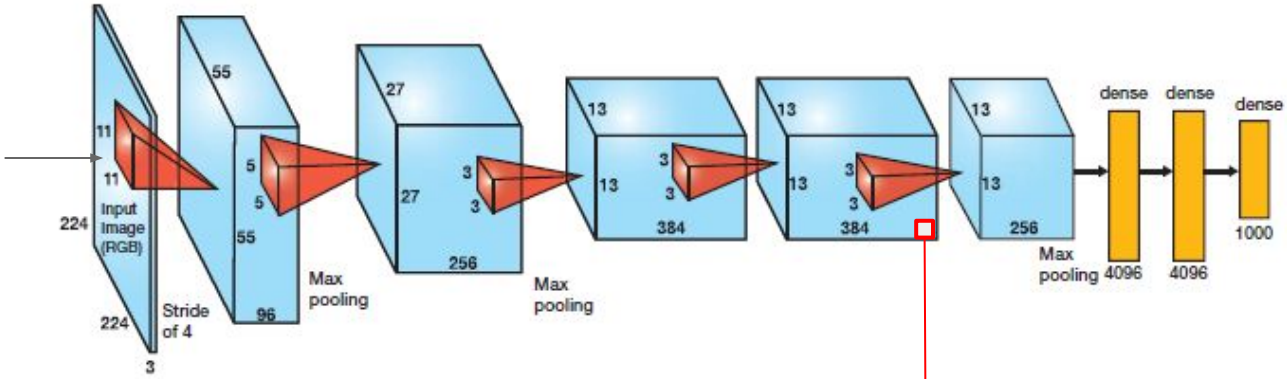


Interpretability: Attribution

- Visualization
- **Attribution**
  - Receptive Field of the Highest Activations
  - Activation Changes after Occlusions
  - Class Activation Maps (CAMs)
  - 2D Projections with t-SNE
  - **Gradient-based**
- Feature visualization

Gradient-based approach

Goal: Visualize the part of an image that mostly activates one of the neurons.



Compute the gradient of any neuron w.r.t. the image

1. Forward image up to the desired layer (e.g. conv5)
2. Set all gradients to 0
3. Set gradient for the neuron we are interested in to 1
4. Backpropagate to get reconstructed image (gradient on the image)

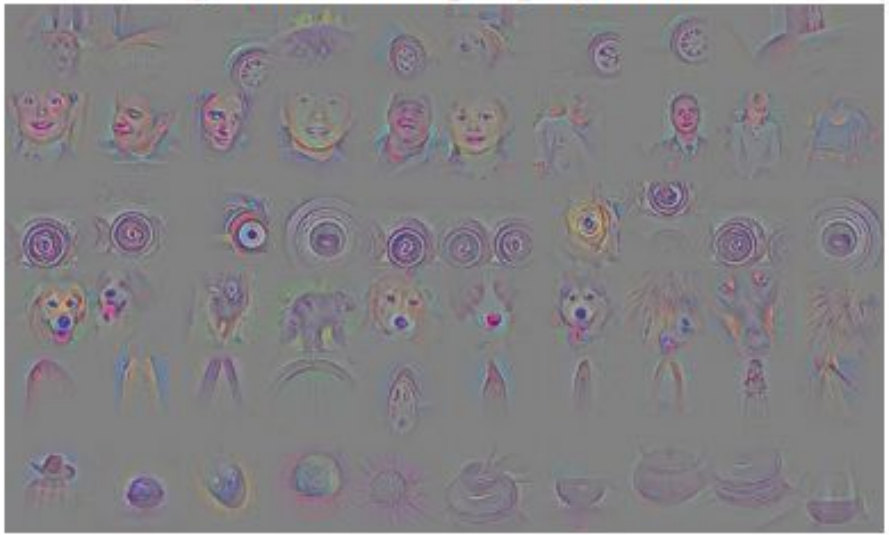
Gradient-based approach

Goal: Visualize the part of an image that mostly activates one of the neurons.

Springenberg, Dosovitskiy, et al. Striving for Simplicity: The All Convolutional Net. ICLR 2015

Gradient-based approach

guided backpropagation



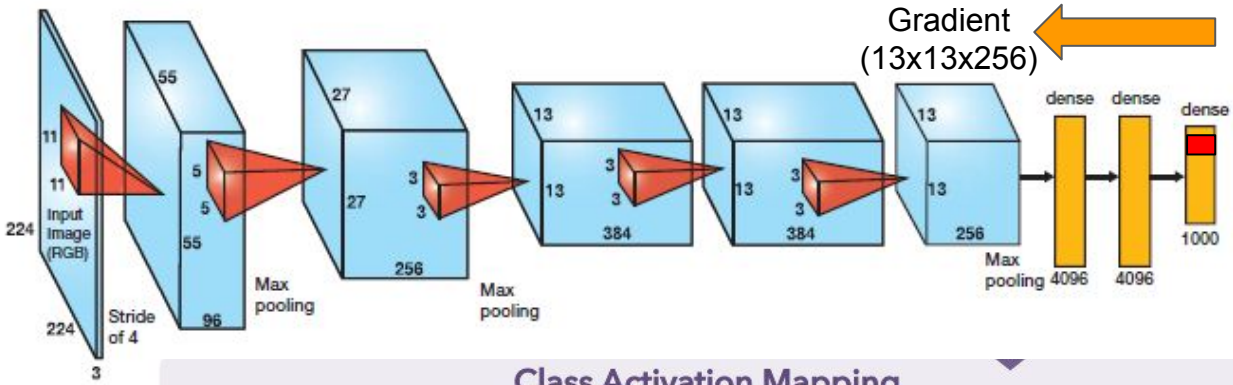
corresponding image crops

Springenberg, Dosovitskiy, et al. Striving for Simplicity: The All Convolutional Net. ICLR 2015

Grad-CAM

Same idea as Class Activation Maps (CAMs), but:

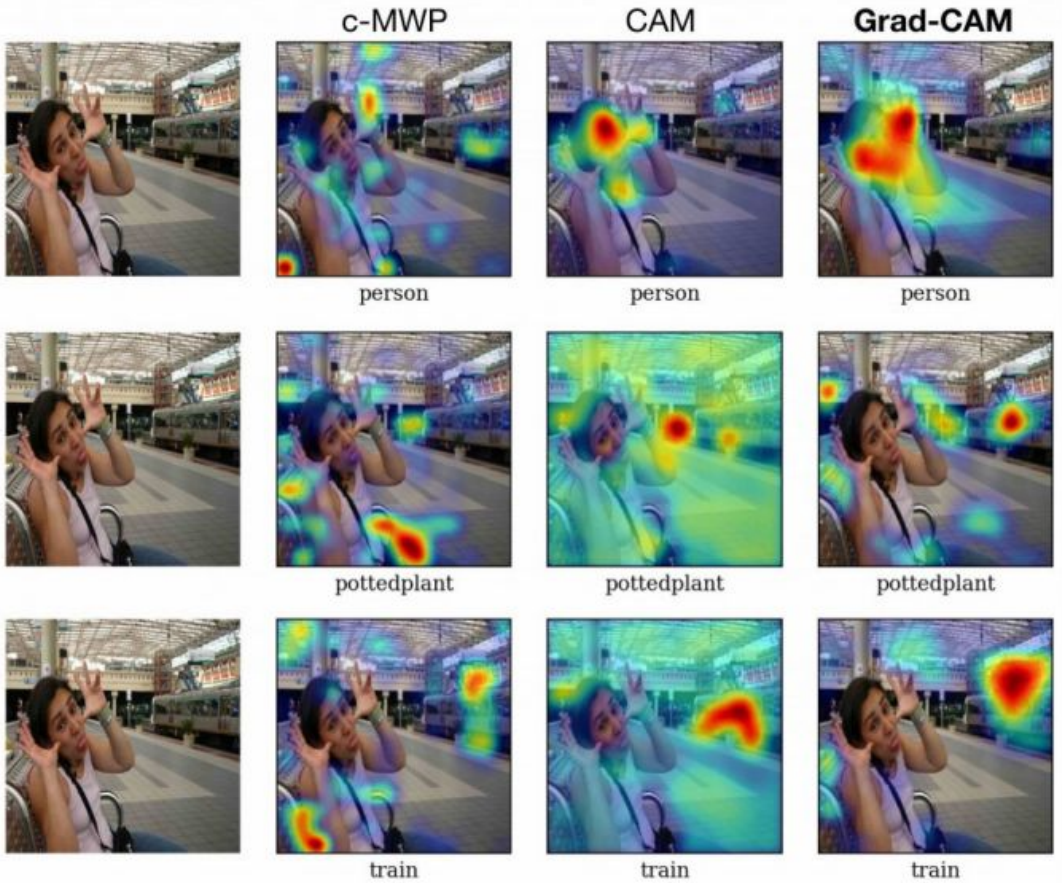
- No modifications required to the network
- Weight feature map by the gradient of the class wrt each channel



Feature maps weighted by the mean gradient.

Ramprasaath et al. Grad-CAM: Why did you say that? Visual Explanations from Deep Networks via Gradient-based Localization. ICCV 2017

Grad-CAM



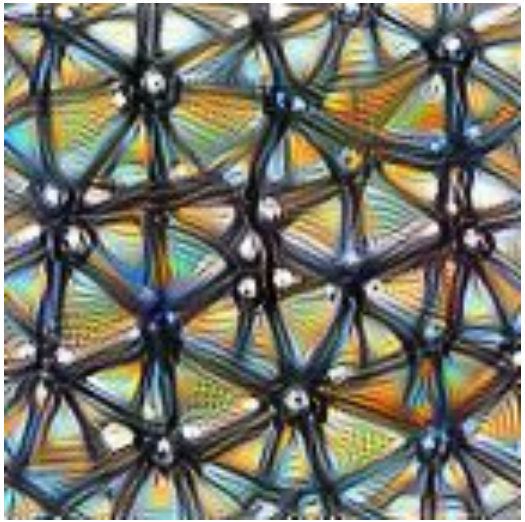
Ramprasaath et al. Grad-CAM: Why did you say that? Visual Explanations from Deep Networks via Gradient-based Localization. ICCV 2017

Interpretability: Attribution

- Visualization
- Attribution
  - Receptive Field of the Highest Activations
  - Activation Changes after Occlusions
  - Class Activation Maps (CAMs)
  - 2D Projections with t-SNE
  - Gradient-based
- **Feature visualization**

Feature Visualization

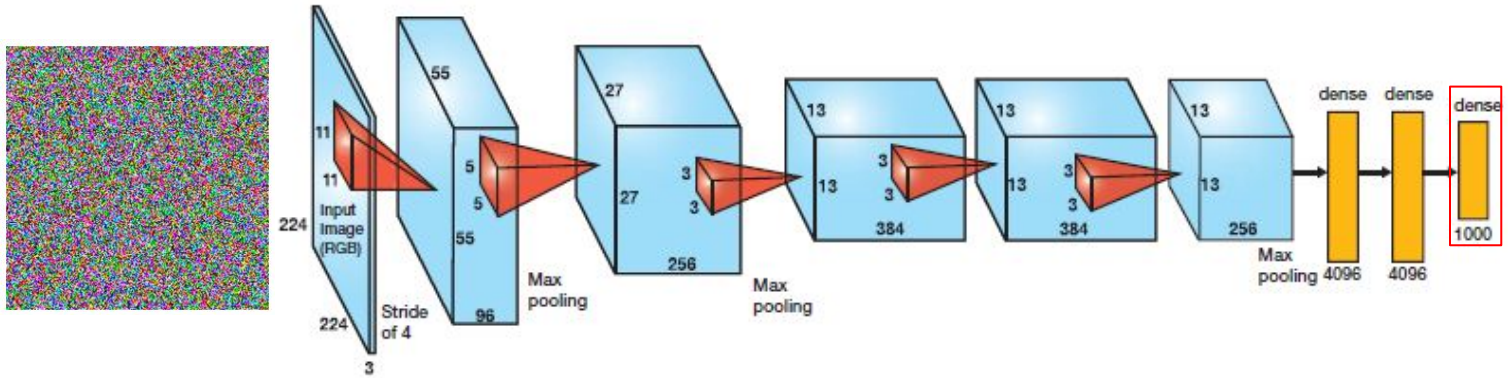
Feature visualization answers questions about what a network — or parts of a network — are looking for by generating examples



Chris Olah, Alexander Mordvintsev, Ludwig Schubert, "Feature Visualization". Distill.pub (November 2017)

Feature Visualization

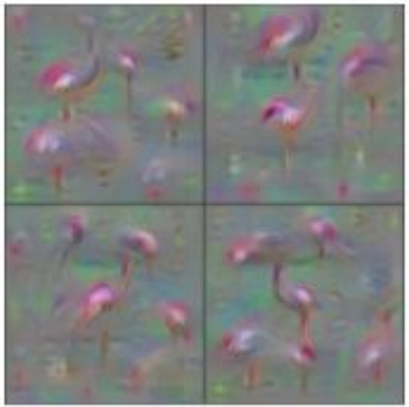
Goal: Generate an image that maximizes the activation of a neuron (or class confidence).



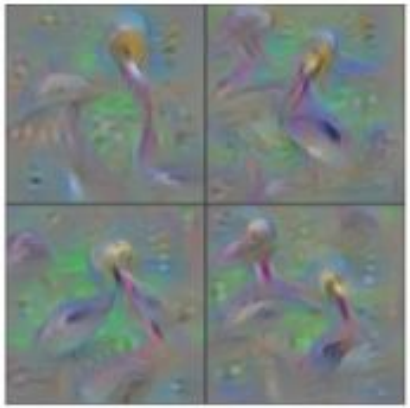
1. Forward random image
2. Set the gradient of the confidence vector to be [0,0,0…,1,...,0,0]
3. Backprop to get gradient on the image
4. Update image (small step in the gradient direction)
5. Repeat

K. Simonyan, A. Vedaldi, A. Ziseramnn, Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps, ICLR 2014

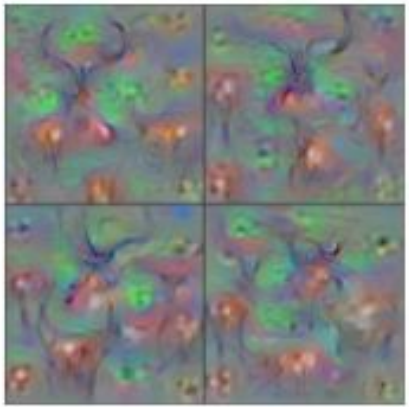
Feature Visualization



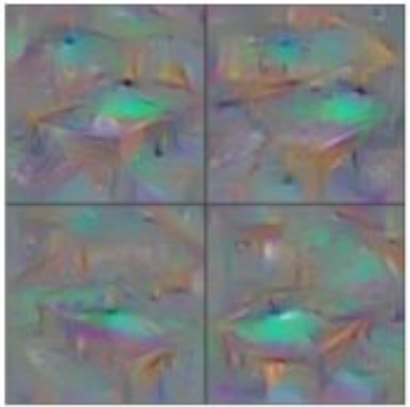
Flamingo



Pelican



Hartebeest



Billiard Table

K. Simonyan, A. Vedaldi, A. Ziseramnn, Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps, ICLR 2014

Feature Visualization

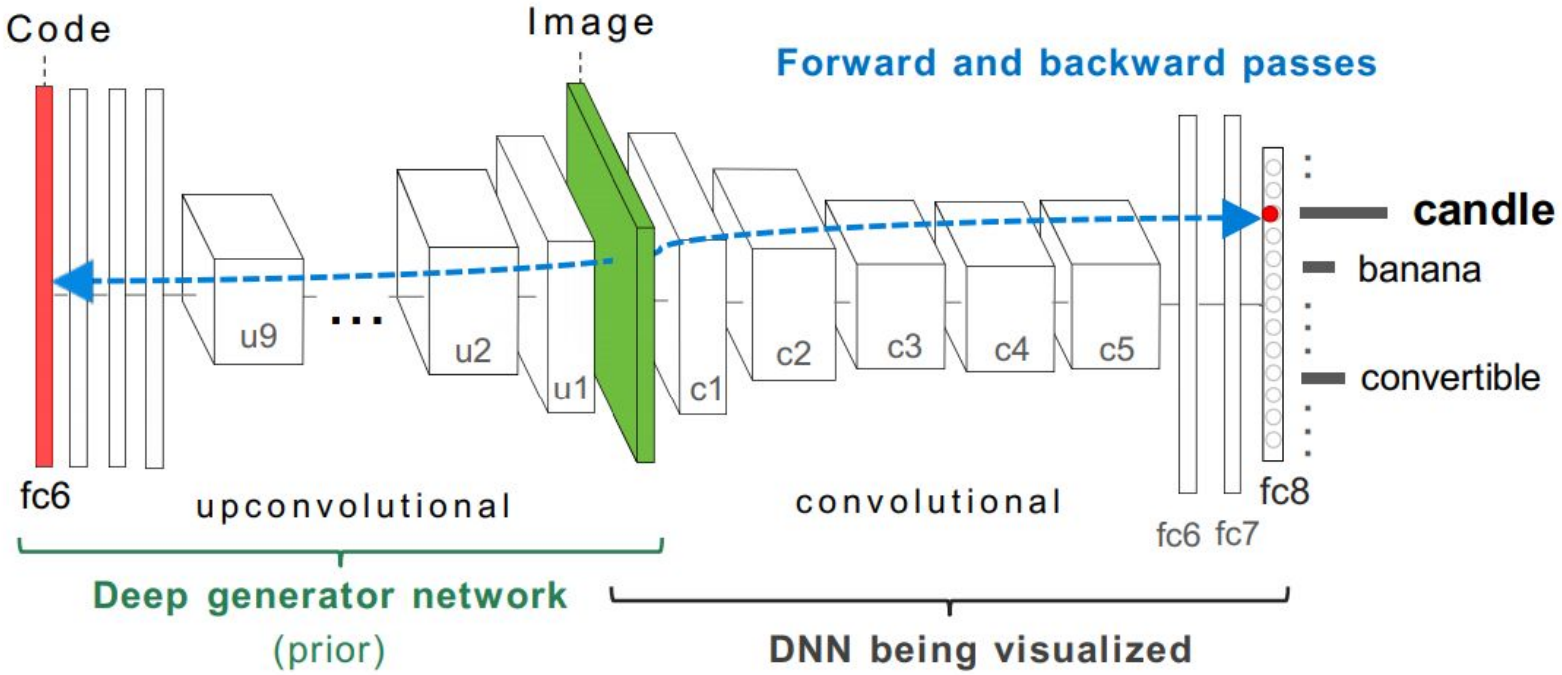
Chris Olah, Alexander Mordvintsev, Ludwig Schubert, "Feature Visualization". Distill.pub (2017) [Google Colab]

Feature Visualization: Regularizations

	Unregularized	Frequency Penalization	Transformation Robustness	Learned Prior	Dataset Examples
 **Erhan, *et al.*, 2009** [3] Introduced core idea. Minimal regularization.	■				
 **Szegedy, *et al.*, 2013** [11] Adversarial examples. Visualizes with dataset examples.	■				■
 **Mahendran & Vedaldi, 2015** [7] Introduces total variation regularizer. Reconstructs input from representation.		■			
 **Nguyen, *et al.*, 2015** [14] Explores counterexamples. Introduces image blurring.		■			
 **Mordvintsev, *et al.*, 2015** [4] Introduced jitter & multi-scale. Explored GMM priors for classes.			■	■	
 **Øygard, *et al.*, 2015** [15] Introduces gradient blurring. (Also uses jitter.)		■	■		
 **Tyka, *et al.*, 2016** [16] Regularizes with bilateral filters. (Also uses jitter.)		■	■		
 **Mordvintsev, *et al.*, 2016** [17] Normalizes gradient frequencies. (Also uses jitter.)		■	■		
 **Nguyen, *et al.*, 2016** [18] Paramaterizes images with GAN generator.				■	
 **Nguyen, *et al.*, 2016** [10] Uses denoising autoencoder prior to make a generative model.				■	

Feature Visualization: Generative prior

Optimizes over the input code (fc6 in the Fig.) instead of the image.



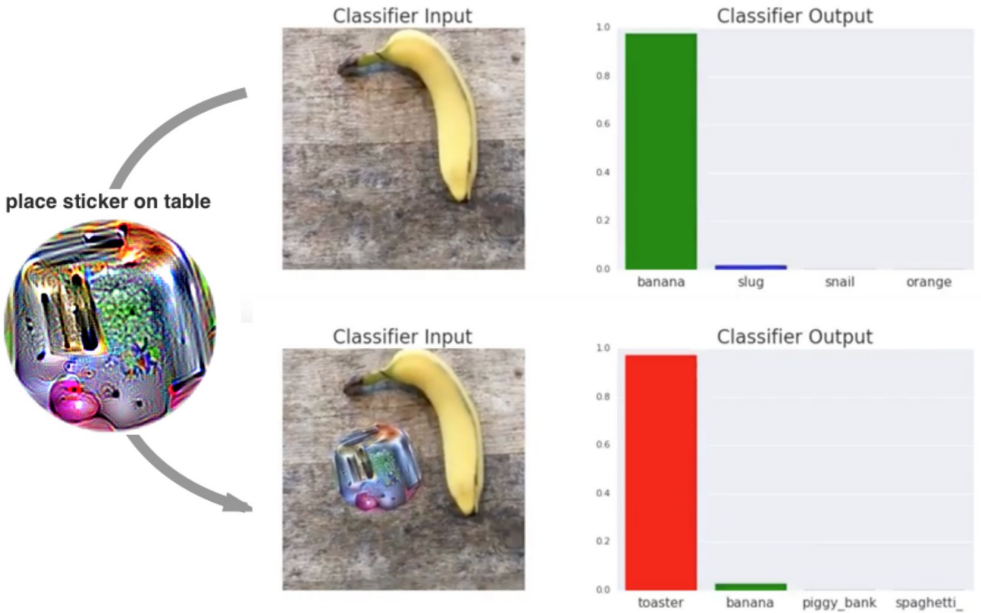
Nguyen, Anh, Alexey Dosovitskiy, Jason Yosinski, Thomas Brox, and Jeff Clune. "Synthesizing the preferred inputs for neurons in neural networks via deep generator networks." NIPS 2016.

Feature Visualization: Generative prior

Nguyen, Anh, Alexey Dosovitskiy, Jason Yosinski, Thomas Brox, and Jeff Clune. "Synthesizing the preferred inputs for neurons in neural networks via deep generator networks." NIPS 2016.

Bonus: Adversarial Attacks (eg. patch on sticker)

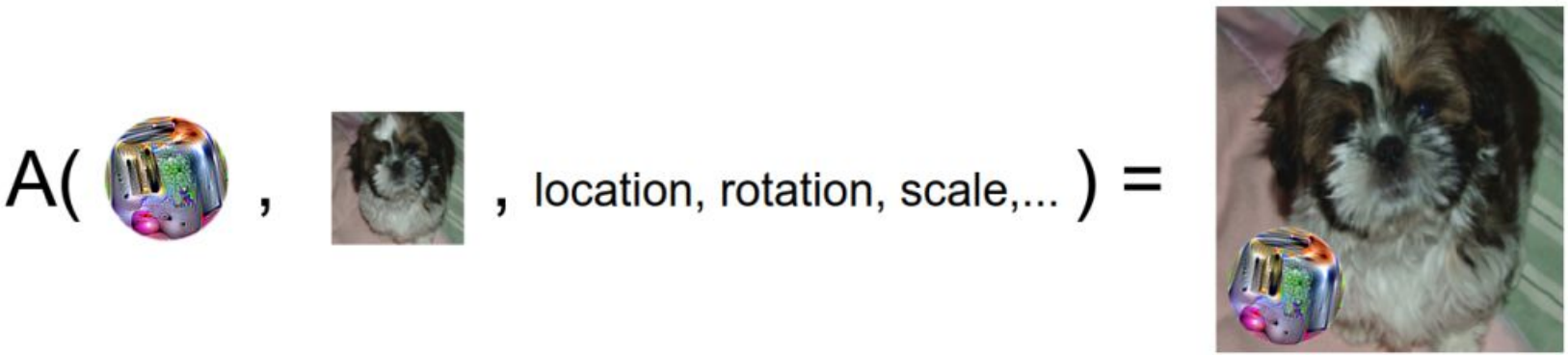
Goal: Generate a patch P that will make a Neural Network N always predict class C for any image containing the patch.



Brown, Tom B., Dandelion Mané, Aurko Roy, Martín Abadi, and Justin Gilmer. "Adversarial patch." arXiv preprint arXiv:1712.09665 (2017). [Google Colab]

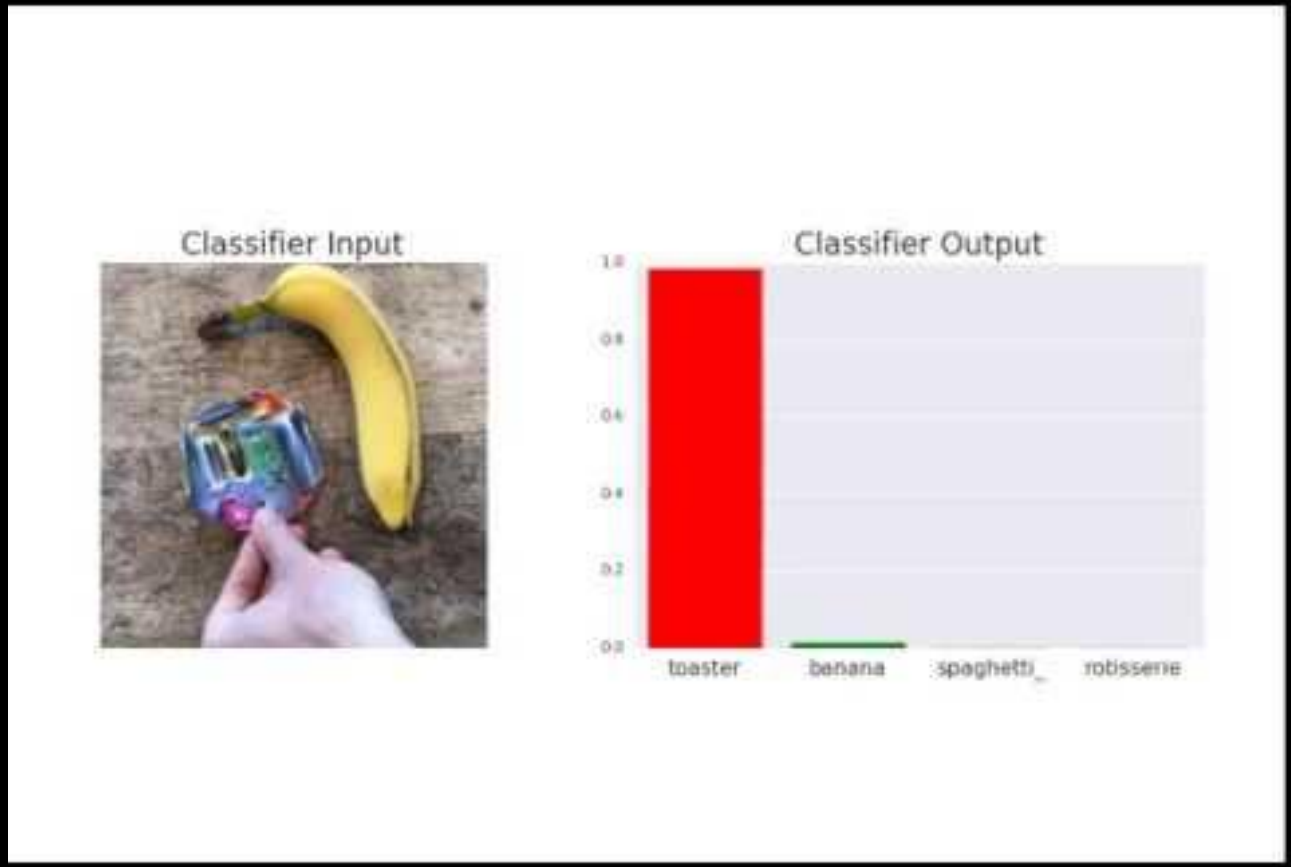
Bonus: Adversarial Attacks (eg. patch on sticker)

How: Backpropagate the gradient to images where the patch has been inserted.



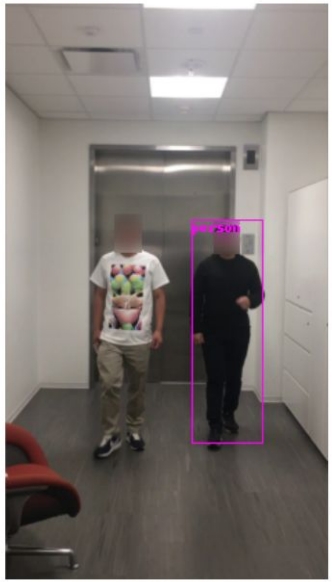
Brown, Tom B., Dandelion Mané, Aurko Roy, Martín Abadi, and Justin Gilmer. "Adversarial patch." arXiv preprint arXiv:1712.09665 (2017).

Sachin Joglekar, "Adversarial patches for CNNs explained" (2018)

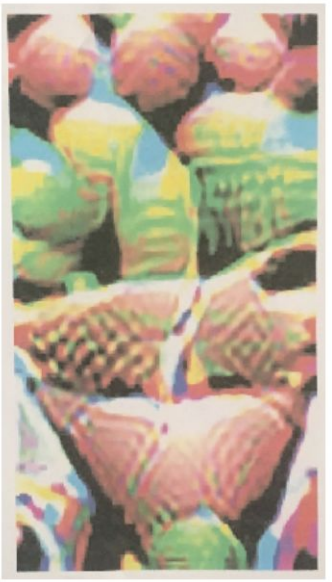


Brown, Tom B., Dandelion Mané, Aurko Roy, Martín Abadi, and Justin Gilmer. "Adversarial patch." arXiv preprint arXiv:1712.09665 (2017).

Bonus: Adversarial Attacks (eg. T-shirts for Privacy)







Xu, K., Zhang, G., Liu, S., Fan, Q., Sun, M., Chen, H., ... & Lin, X. (2019). Evading Real-Time Person Detectors by Adversarial T-shirt. arXiv preprint arXiv:1910.11099.

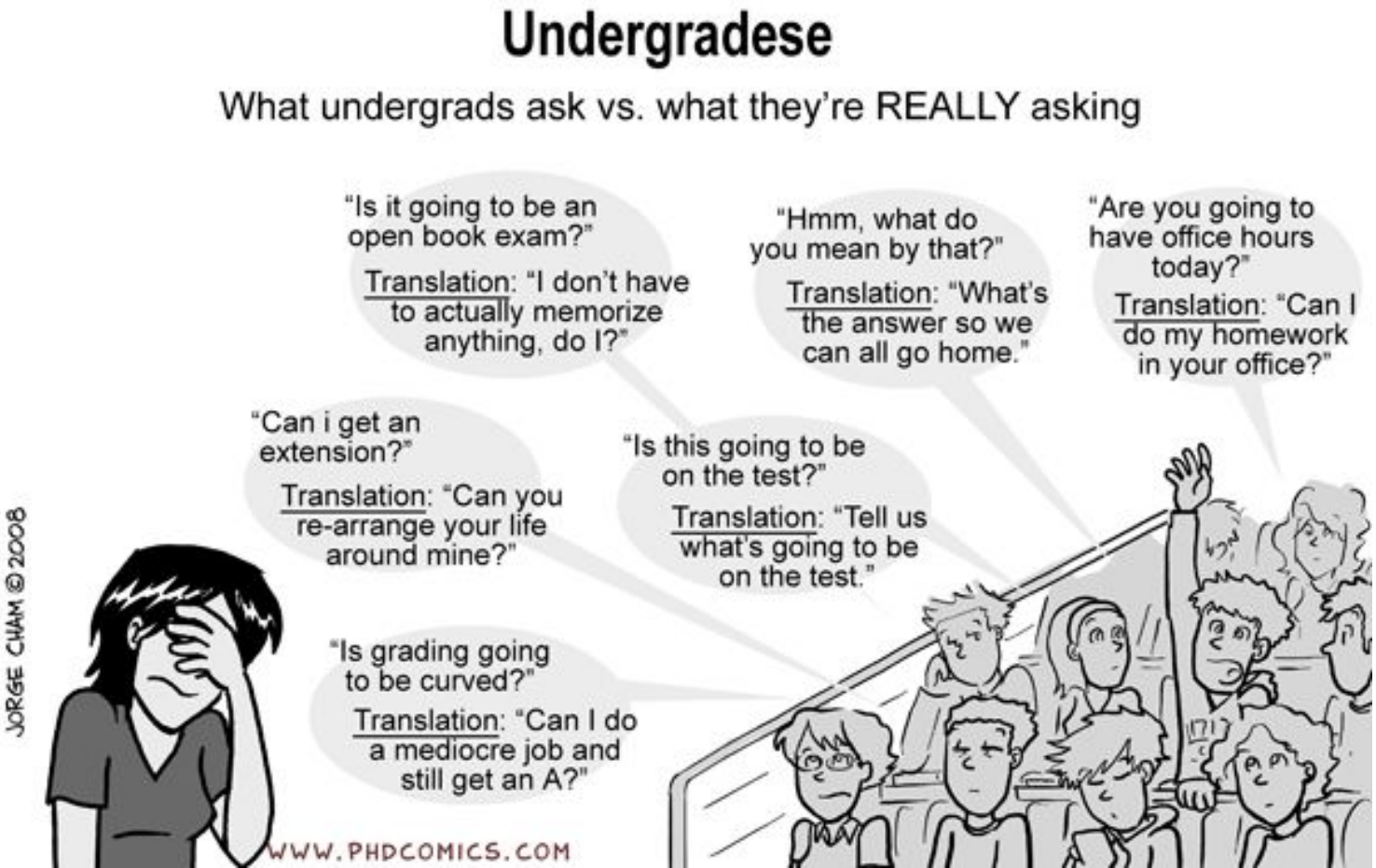
Further readings

Shan Carter, Zan Armstrong, Ludwig Schubert, Ian Johnson, Chris Olah, “Exploring Neural Networks with Activation Atlases”. Distill.pub (March 2019)

Adebayo, Julius, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. "Sanity checks for saliency maps." NeurIPS 2018.

Software: Captum, model interpretability for PyTorch

Questions ?


Undergradese
What undergrads ask vs. what they're REALLY asking
"Is it going to be an open book exam?"
Translation: "I don't have to actually memorize anything, do I?"
"Hmm, what do you mean by that?"
Translation: "What's the answer so we can all go home."
"Are you going to have office hours today?"
Translation: "Can I do my homework in your office?"
"Can i get an extension?"
Translation: "Can you re-arrange your life around mine?"
"Is this going to be on the test?"
Translation: "Tell us what's going to be on the test."
"Is grading going to be curved?"
Translation: "Can I do a mediocre job and still get an A?"
JORGE CHAM © 2008
WWW.PHDCOMICS.COM