Neural Networks II

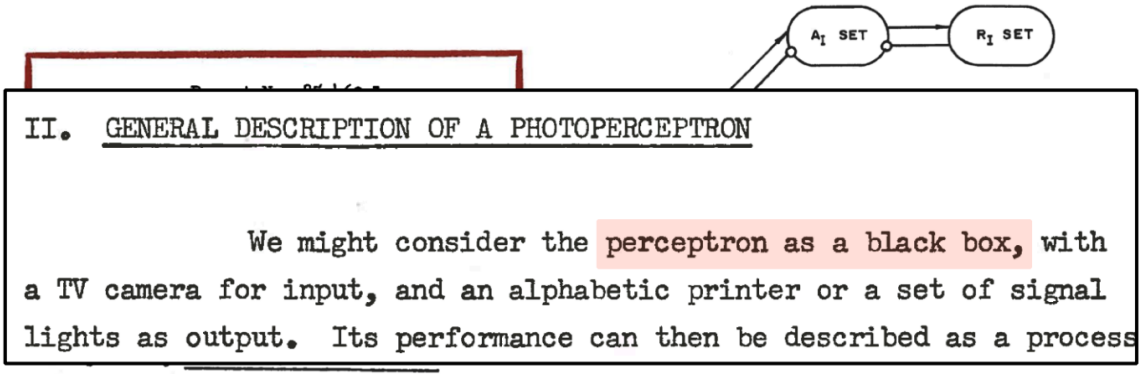
Mengye Ren

NYU

September 26, 2023

Neural Networks Interpretability

ML=Blackbox?



1

[1] Slide credit: Wojciech Samek

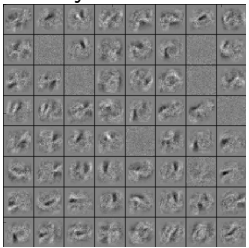
ML Interpretability

- Linear regression: Weights represent feature selection strength
- SVMs: Dual weights represent sample selection
- Bayesian methods: Model the generative process as a probabilistic model, fully transparent
- Decision trees: If-else decision making process
- Neural networks: ?

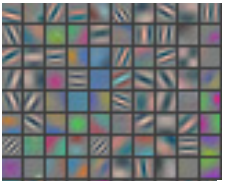
Feature Visualization

- Recall: we can understand what first-layer features are doing by visualizing the weight matrices.
- Higher-level weight matrices are hard to interpret.

Fully connected



Convolutional



Zeiler and Fergus, Visualizing and understanding convolutional networks, ECCV 2014.

- The better the input matches these weights, the more the feature activates.
  - Obvious generalization: visualize higher-level features by seeing what inputs activate them.

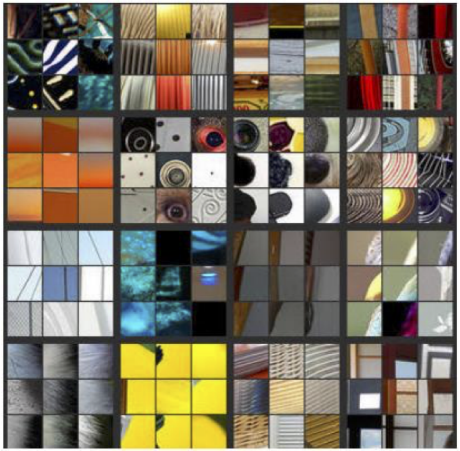
Feature Visualization

- One way to formalize: pick the images in the training set which activate a unit most strongly.
- Here's the visualization for layer 1:

Zeiler and Fergus, Visualizing and understanding convolutional networks, ECCV 2014.

Feature Visualization

- Layer 3:



Zeiler and Fergus, Visualizing and understanding convolutional networks, ECCV 2014.

Feature Visualization

- Layer 4:

Feature Visualization

- Layer 5:

Feature Visualization

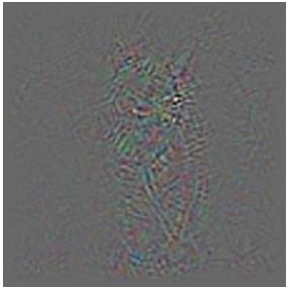
- Higher layers seem to pick up more abstract, high-level information.
- Problems?
  - Can't tell what the unit is actually responding to in the image.
  - We may read too much into the results, e.g. a unit may detect red, and the images that maximize its activation will all be stop signs.
- Can use input gradients to diagnose what the unit is responding to.

Feature Visualization

- Input gradients can be hard to interpret.
- Take a good object recognition conv net (Alex Net) and compute the gradient of $\log p(y = \text{“cat”}|\mathrm{x})$:

Original image

Gradient for “cat”

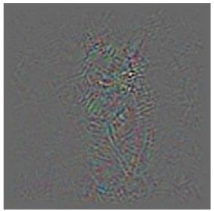


Feature Visualization

- **Guided backprop** is a total hack to prevent this cancellation.
- Do the backward pass as normal, but apply the ReLU nonlinearity to all the activation error signals.

$$y = \mathrm{ReLU}(z) \qquad \bar{z} = \begin{cases} \bar{y} & \text{if } z > 0 \text{ and } \bar{y} > 0 \\ 0 & \text{otherwise} \end{cases}$$

- We want to visualize what excites given unit, not what suppresses it.

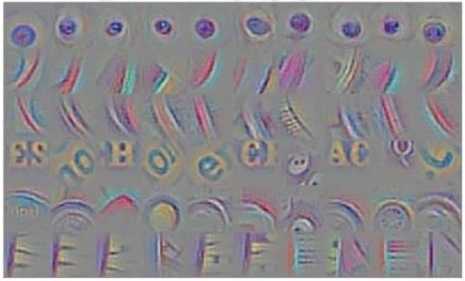


Backprop

Guided Backprop

Guided Backprop

guided backpropagation



corresponding image crops

guided backpropagation

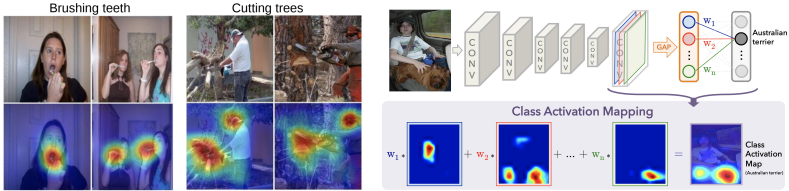
corresponding image crops

Class activation map (CAM)

Classification networks typically use global avg pooling before the final layer.

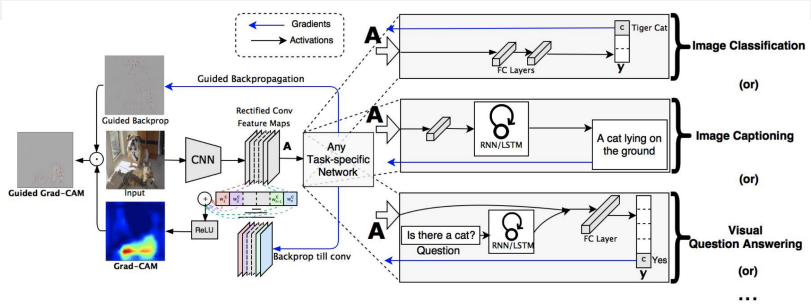
This pooling layer can already contain semantic information.

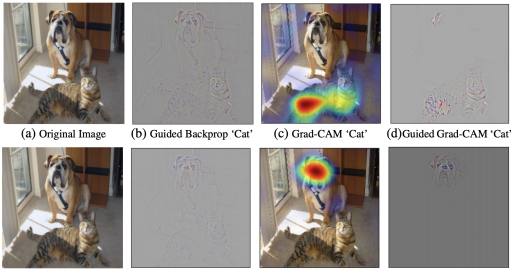
We can visualize a heat map



Zhou et al. Learning deep features for discriminative localization. CVPR 2016.

GradCAM





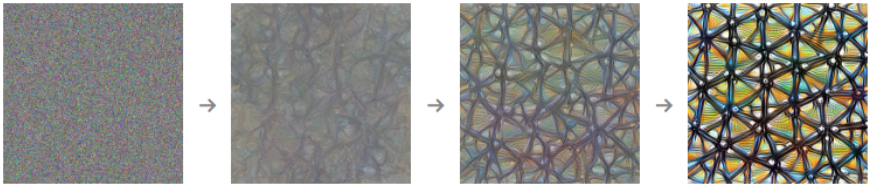
(a) Original Image (b) Guided Backprop 'Cat' (c) Grad-CAM 'Cat' (d)Guided Grad-CAM 'Cat'

(g) Original Image (h) Guided Backprop 'Dog' (i) Grad-CAM 'Dog' (j)Guided Grad-CAM 'Dog'

Gradient Ascent on Images

- Can do gradient ascent on an image to maximize the activation of a given neuron.

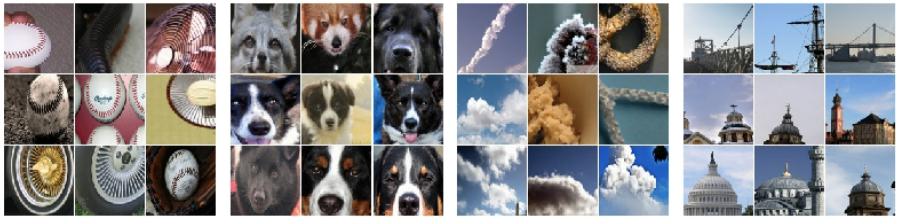
Starting from random noise, we optimize an image to activate a particular neuron (layer mixed4a, unit 11).



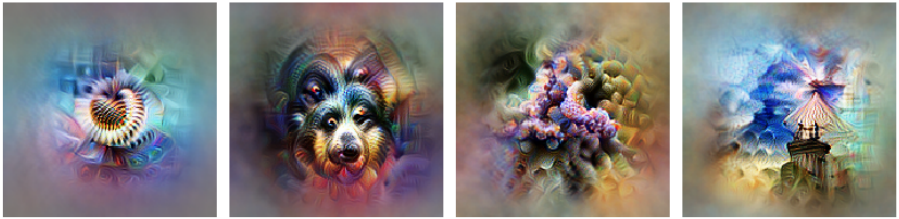
https://distill.pub/2017/feature-visualization/

Gradient Ascent on Images

Dataset Examples show us what neurons respond to in practice



Optimization isolates the causes of behavior from mere correlations. A neuron may not be detecting what you initially thought.



Baseball—or stripes?
mixed4a, Unit 6

Animal faces—or snouts?
mixed4a, Unit 240

Clouds—or fluffiness?
mixed4a, Unit 453

Buildings—or sky?
mixed4a, Unit 492

https://distill.pub/2017/feature-visualization/

Gradient Ascent on Images

- Higher layers in the network often learn higher-level, more interpretable representations

Edges (layer conv2d0) **Textures** (layer mixed3a) **Patterns** (layer mixed4a)

https://distill.pub/2017/feature-visualization/

Gradient Ascent on Images

- Higher layers in the network often learn higher-level, more interpretable representations

Parts (layers mixed4b & mixed4c) **Objects** (layers mixed4d & mixed4e)

https://distill.pub/2017/feature-visualization/

Deep dream

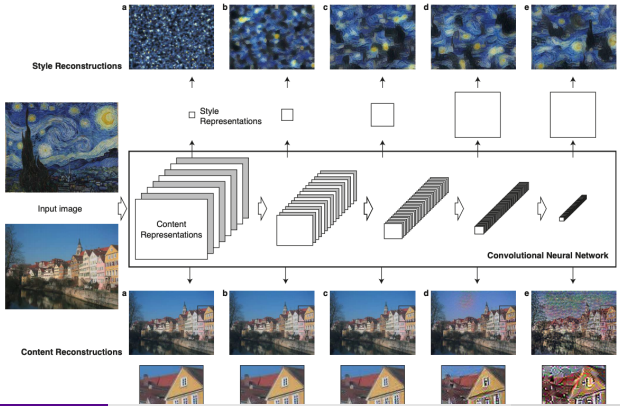
- Start with an image, and run a conv net on it.
- Change the image such that units which were already highly activated get activated even more strongly. "Rich get richer."

Deep dream

Deep dream

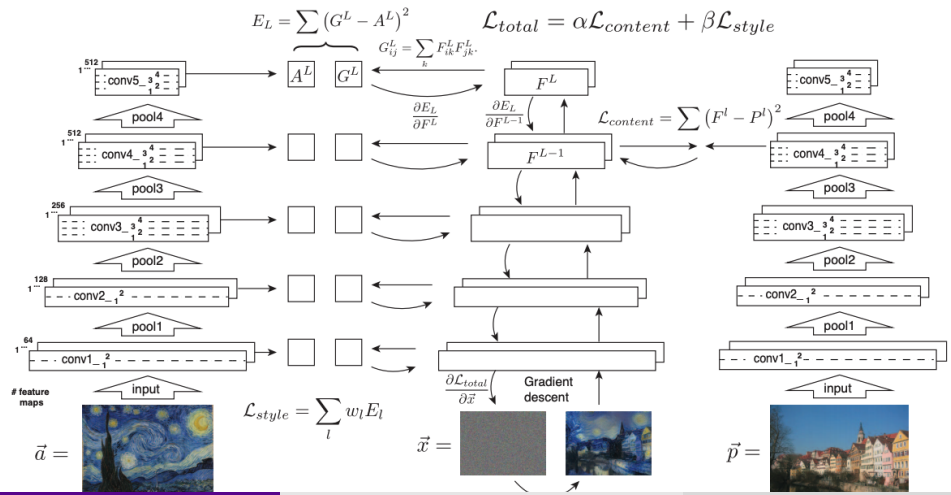
Artistic style transfer

- Activation stores content information
- Activation correlation across space stores style information and discards spatial arrangement



Artistic style transfer

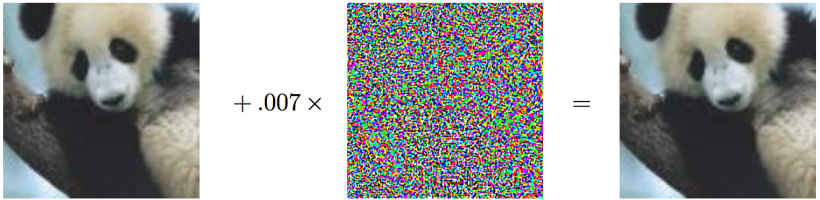
- Optimizing both content & style from random noise



Artistic style transfer

Adversarial Examples

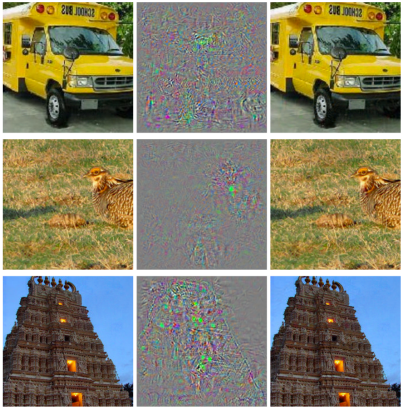
- One of the most surprising findings about neural nets has been the existence of **adversarial inputs**, i.e. inputs optimized to fool an algorithm.

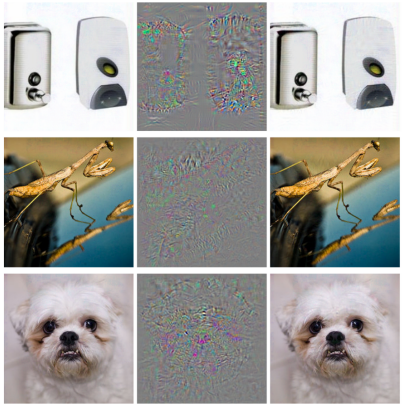


Goodfellow et al., Explaining and harnessing adversarial examples, ICLR 2015.

Adversarial Examples

- The following adversarial examples are misclassified as ostriches. ($10\times$ perturbation visualized in middle.)





Szegedy et al., Intriguing properties of neural networks, ICLR 2014.

Adversarial Examples

- You can print out an adversarial image and take a picture of it, and it still works!

(a) Printout (b) Photo of printout (c) Cropped image

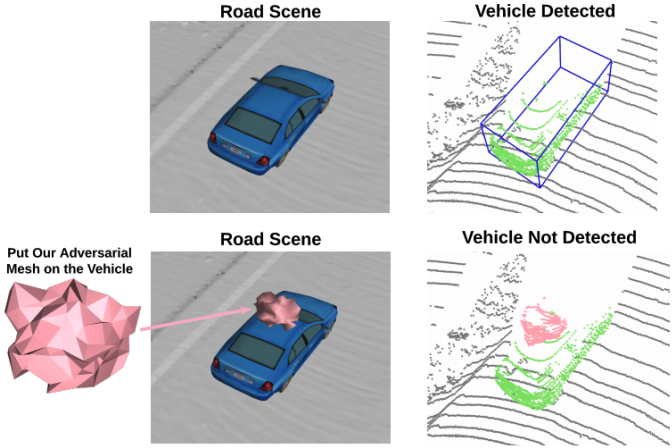
Kurakin et al., Adversarial examples in the physical world, ICLR workshop 2017.

Adversarial Examples

- An adversarial example in the physical world (network thinks it's a gun, from a variety of viewing angles!)

Adversarial Examples

- An adversarial mesh object that can hide cars from LiDAR detector



Tu et al., Physically realizable adversarial examples for LiDAR object detection, CVPR 2020.

Adversarial Defense

- How to defend from adversarial perturbation is still an active research area.
- One common approach is to train with millions of adversarial examples.
- Needs to train much longer, and also suffers a little from normal accuracy.