Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization

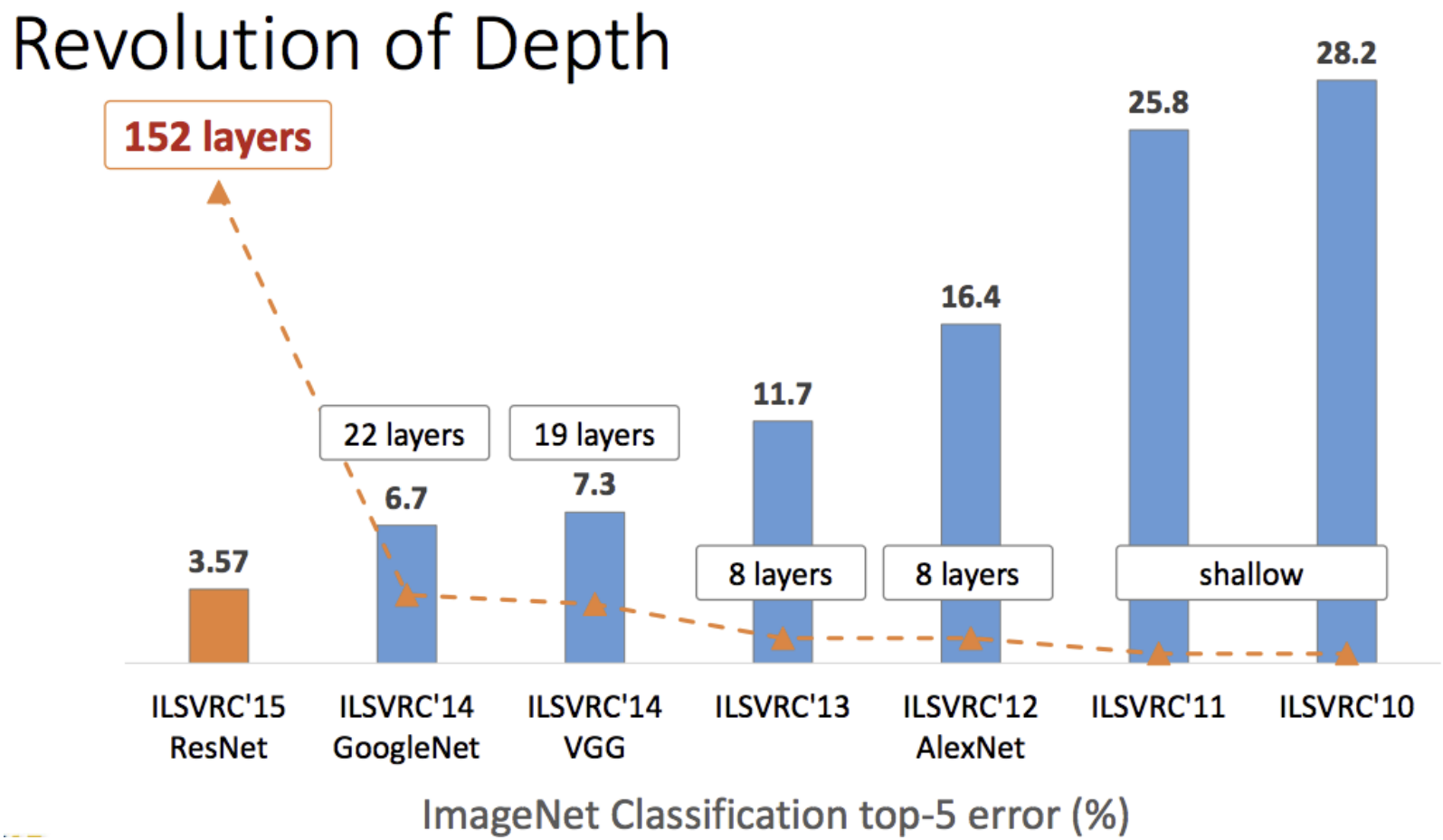
Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, Dhruv Batra

Pedro Herruzo

July 1, 2017

Introduction (I)

Nowadays deeper means better:



Introduction (II)

By using deep models, we **sacrifice interpretable modules** for uninterpretable ones that achieve **greater performance** through greater abstraction (more layers) and tighter integration (end-to-end training).

When models fail, they fail spectacularly disgracefully, without warning or explanation, leaving a user staring at an incoherent output, wondering why.

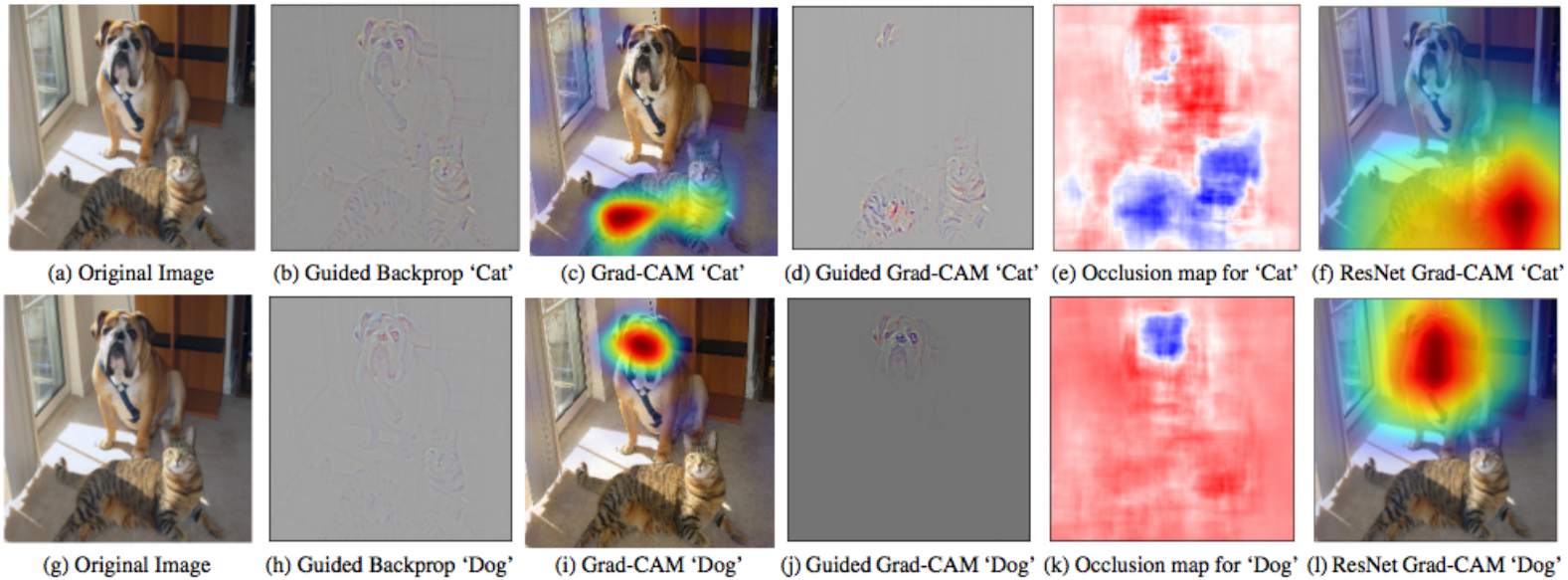
Introduction (III)

Interpretability Matters:

- When AI is **significantly weaker than humans** and not yet reliably 'deployable' (e.g., visual question answering), the goal of transparency and explanations is to **identify the failure modes**, thereby helping researchers focus their efforts on the most fruitful research directions.
- When AI is **on par with humans** and reliably 'deployable' (e.g., image classification), the goal is to **establish appropriate trust and confidence in users**.
- When AI is **significantly stronger than humans** (e.g., chess or Go), the goal of explanations is in **machine teaching** (i.e., a machine teaching a human about how to make better decisions).

Introduction (IV)

What do we want?



(a) Original Image (b) Guided Backprop 'Cat' (c) Grad-CAM 'Cat' (d) Guided Grad-CAM 'Cat' (e) Occlusion map for 'Cat' (f) ResNet Grad-CAM 'Cat'

(g) Original Image (h) Guided Backprop 'Dog' (i) Grad-CAM 'Dog' (j) Guided Grad-CAM 'Dog' (k) Occlusion map for 'Dog' (l) ResNet Grad-CAM 'Dog'

Understanding Representations Learned in Deep Architectures

Dumitru Erhan, Aaron Courville, and Yoshua Bengio

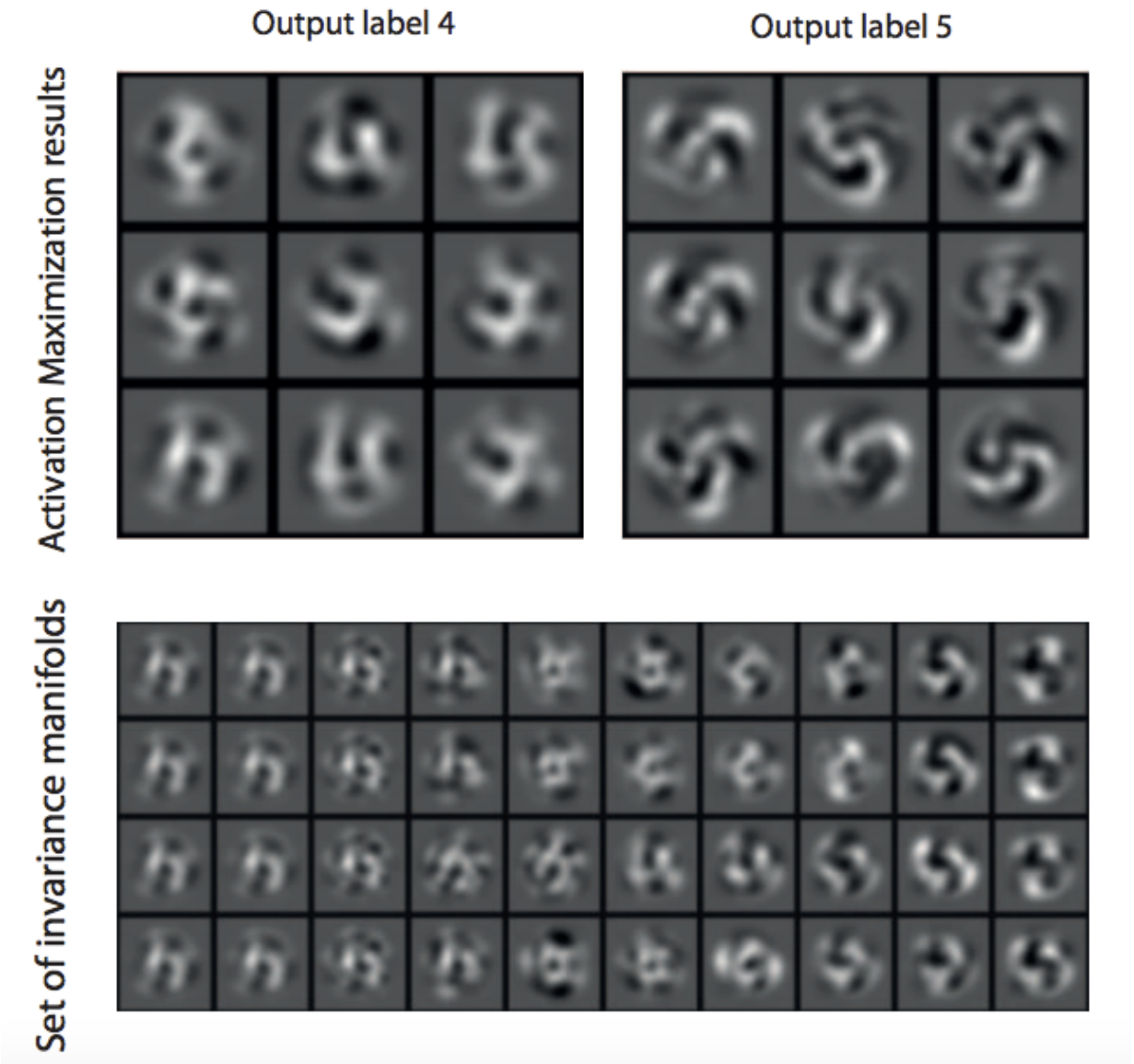
They propose two methods: First, Sampling from top to down using that layers $j-1$ and j form an RBM from which they can sample using block Gibbs sampling. Second, their new idea: maximizing the activation of a unit as an optimization problem fixing the parameters after training the network and learning the input image:

$$\mathbf{x}^* = \arg \max_{\mathbf{x} \; s.t. \; ||\mathbf{x}||=\rho} h_{ij}(\theta, \mathbf{x})$$

Beyond that, they also explore the invariance manifolds for each of the hidden units w.r.t the target class:

$$\mathbf{x}^*_\varepsilon = \arg \max_{\mathbf{x} \; s.t. \; ||\mathbf{x}||=\rho \text{ and } ||\mathbf{x}-\mathbf{x}_{opt}||=\varepsilon\rho} h_{ij}(\theta, \mathbf{x})$$

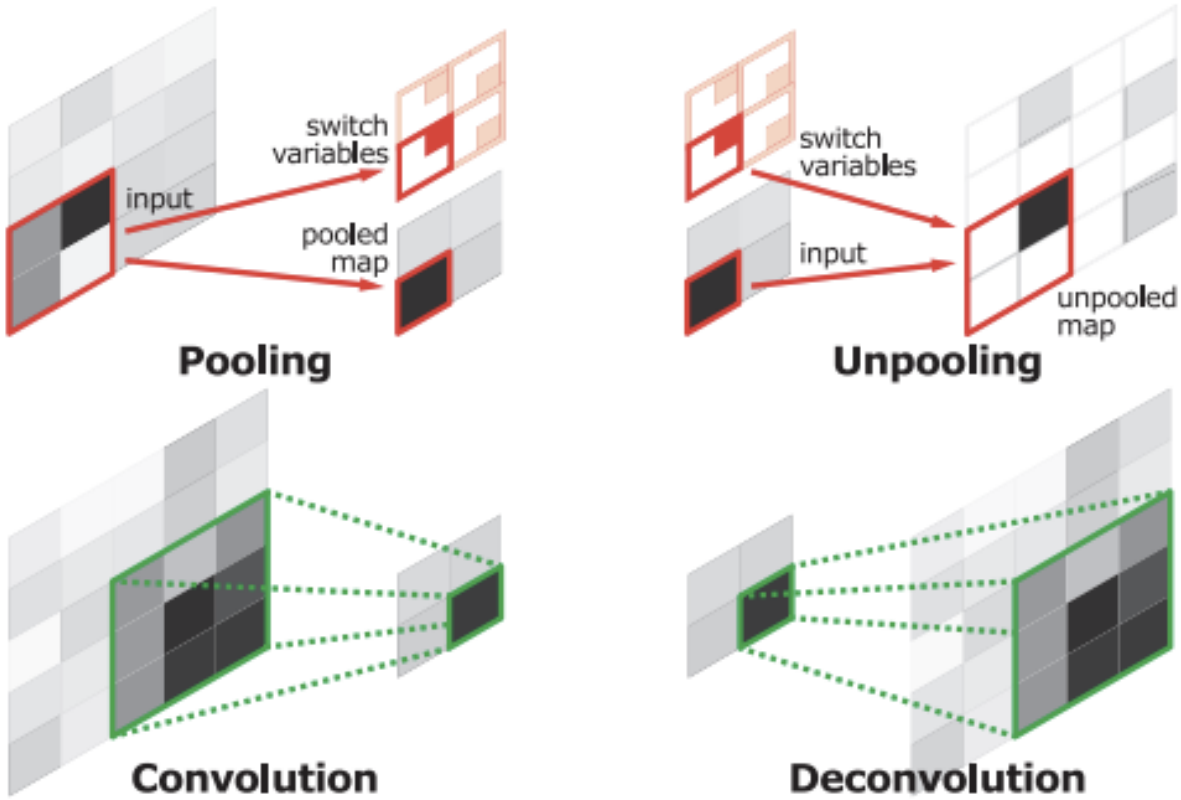
2010: Freeze weights and learn the input (II)



Visualizing and Understanding Convolutional Networks

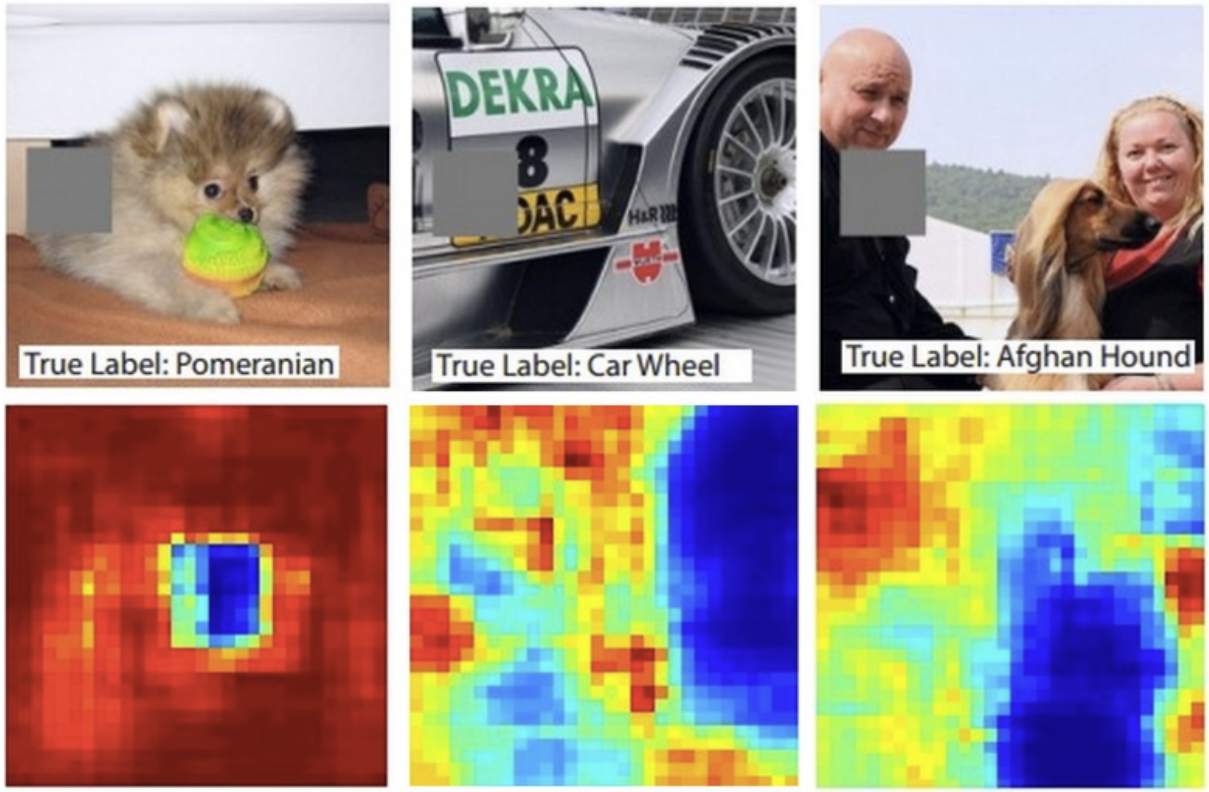
Matthew D. Zeiler, Rob Fergus

They propose two methods: First, Up sampling top-down and computing the transpose of a convolution (deconvolution):



2013: Deconvolutions and patch occlusions (II)

Second, they perturb inputs by occluding patches and classifying the occluded image, typically resulting in lower classification scores for relevant objects when those objects are occluded:



Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps

Karen Simonyan, Andrea Vedaldi, Andrew Zisserman

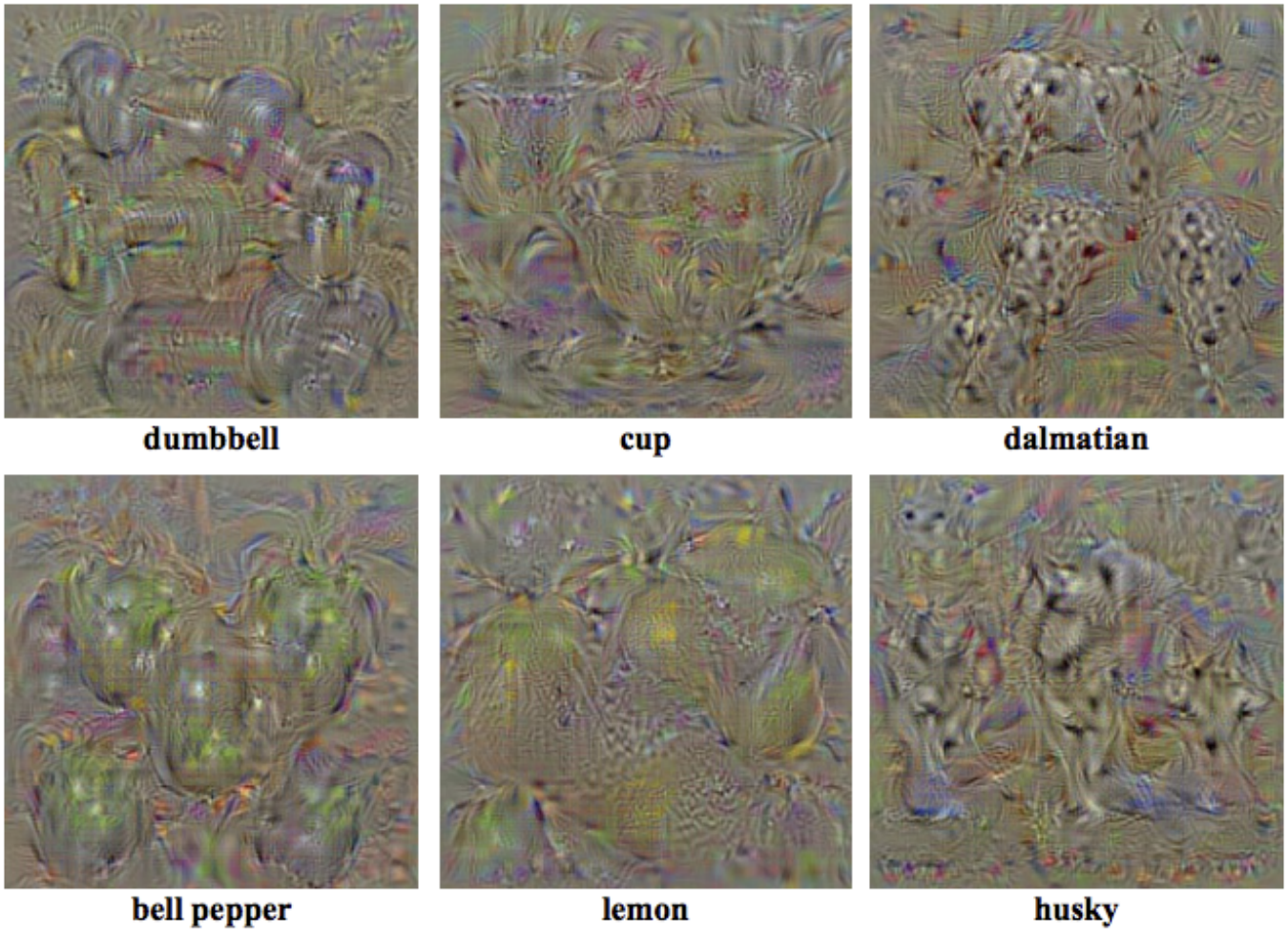
They propose three methods: First, they establish connection between the gradient-based ConvNet visualisation methods and deconvolutional networks. In short, they differ just in how backpropagate the non-liniarities (i.e., ReLu).

Second, they generate an image by maximizing the activation of a unit in the last layer of a CNN very similarly to Bengio in 2010. More formally, let $S_c(I)$ be the score of the class c, computed by the classification layer of the ConvNet for an image I. We would like to find an L_2-regularised image, such that the score S_c is high:

$$\arg\max_I S_c(I) - \lambda \|I\|_2^2$$

2014: Class saliency map using backprop (II)

Creating this results:



2014: Class saliency map using backprop (III)

Third, they presented a new method based on backpropagate the importance of each pixel from a target class. Consider the linear score model for the class c

$$S_c(I) = w_c^T I + b_c$$

Here is easy to see that the magnitude of elements of w defines the importance of the corresponding pixels of I for the class c. In a CNN we can we approximate $S_c(I)$ with a linear function in the neighbourhood of I_0 by computing the first-order Taylor expansion:

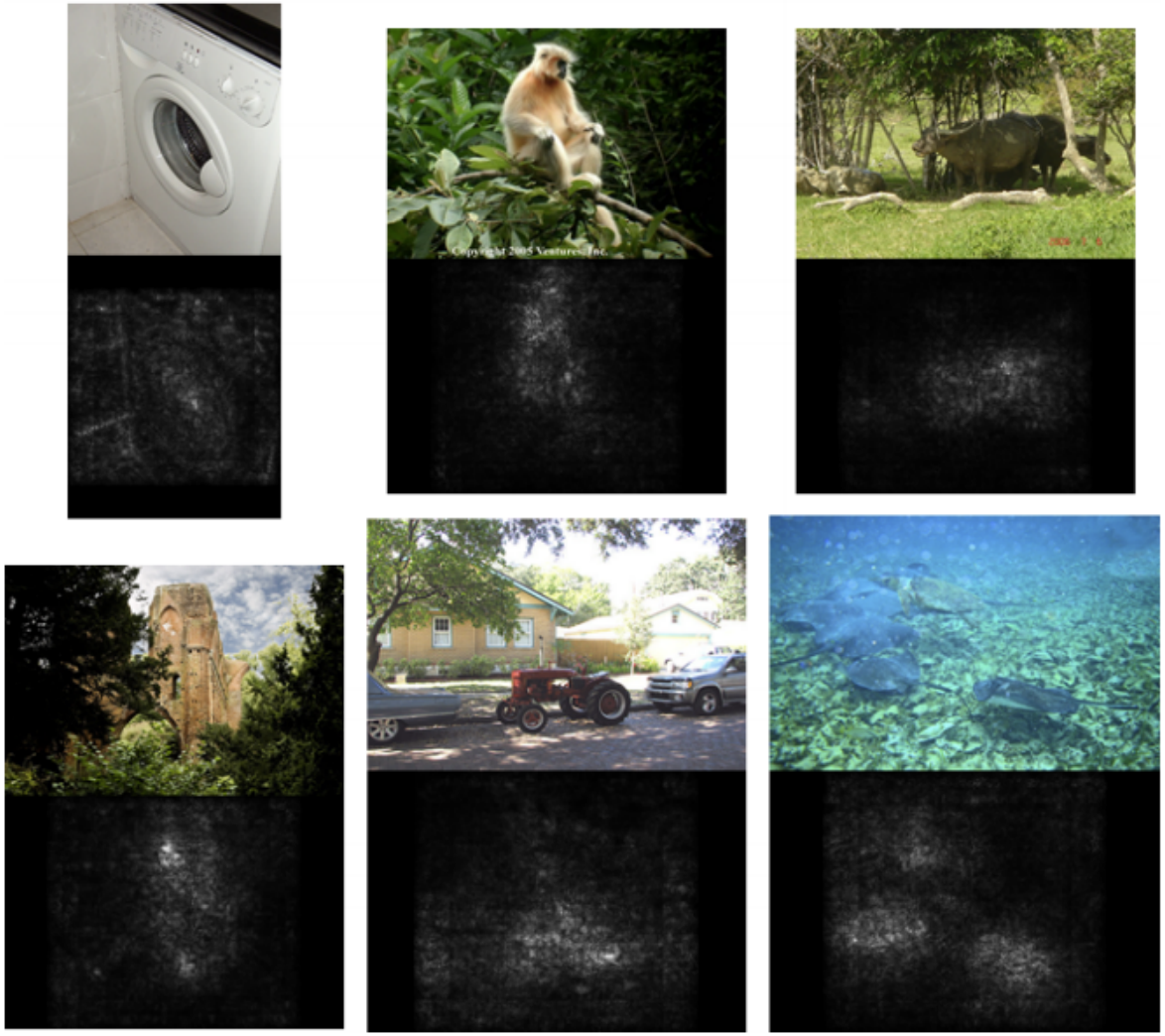
$$S_c(I) \approx w^T I + b$$

Where here the weights are just computed with backprop:

$$w = \left. \frac{\partial S_c}{\partial I} \right|_{I_0}$$

2014: Class saliency map using backprop (IV)

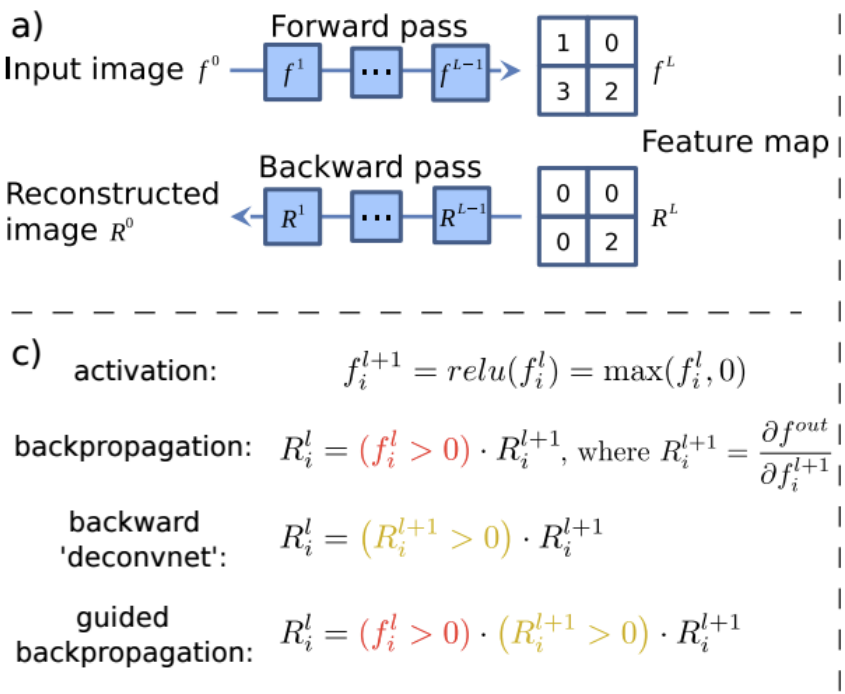
Creating this results:

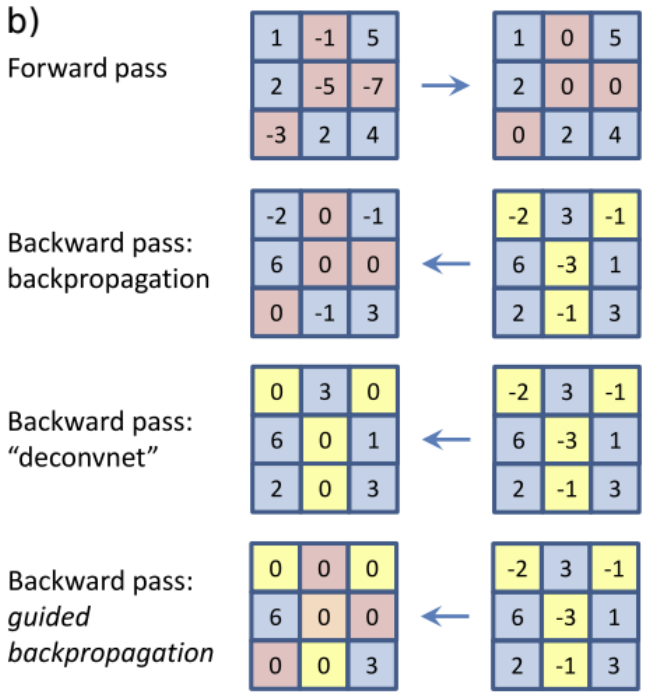


Guided backpropagation

Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, Martin Riedmiller

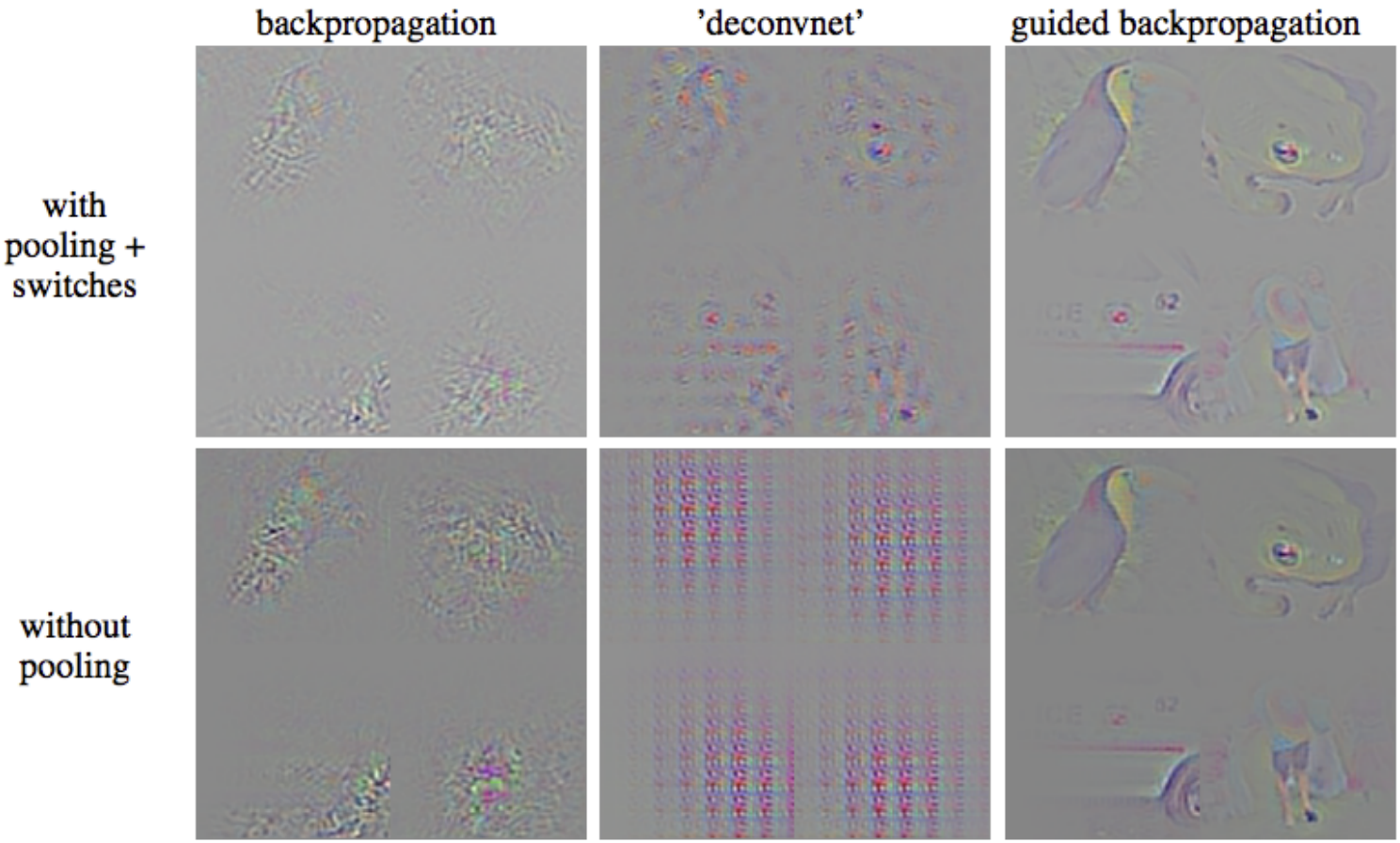
They propose two methods: A new architecture that consists solely of convolutional layers. Second, they add an additional guidance signal from the higher layers to usual backprop:





2015: Guided backpropagation (II)

Creating this results:



DISTINCT CLASS SALIENCY MAPS FOR MULTIPLE OBJECT IMAGES

Wataru Shimoda, Keiji Yanai

They propose three methods: Using CNN derivatives with respect to feature maps of the intermediate convolutional layers with up-sampling, instead of an input image as it is done by Simonyan:

$$v_i^c = \left. \frac{\partial S_c}{\partial L_i} \right|_{L_i^0}$$

Second, aggregating multiple-scale class saliency maps to compensate lower resolution of the feature maps.

2016: Backprop to intermediate layers (II)

Creating this results:

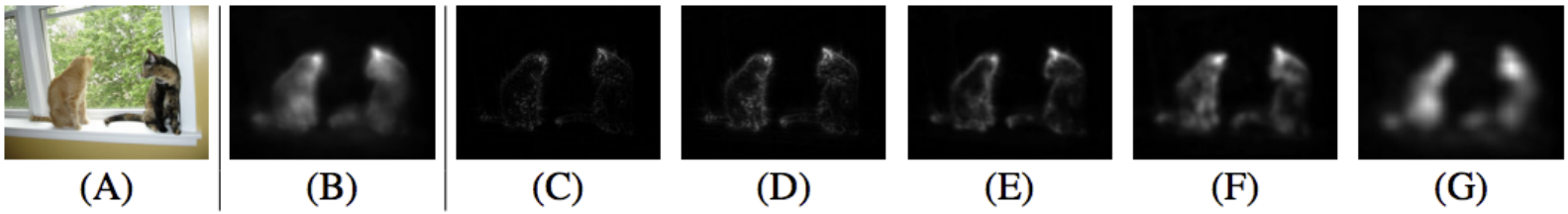


Figure 2: Class saliency maps obtained from the VGG16-net fine-tuned with the PASCAL VOC 2012 dataset. (A) an input image, (B) average of [(E)(F)(G)], (C) conv1_1, (D) conv2_1, (E) conv3_2, (F) conv4_2, (G) conv5_2

Third, subtracting saliency maps of the other classes from saliency maps of the target class to differentiate target objects from other objects:

$$\tilde{M}^c_{i,x,y} = \sum_{c' \in candidates} \max\left(M^c_{i,x,y} - M^{c'}_{i,x,y}, 0\right)[c \neq c']$$

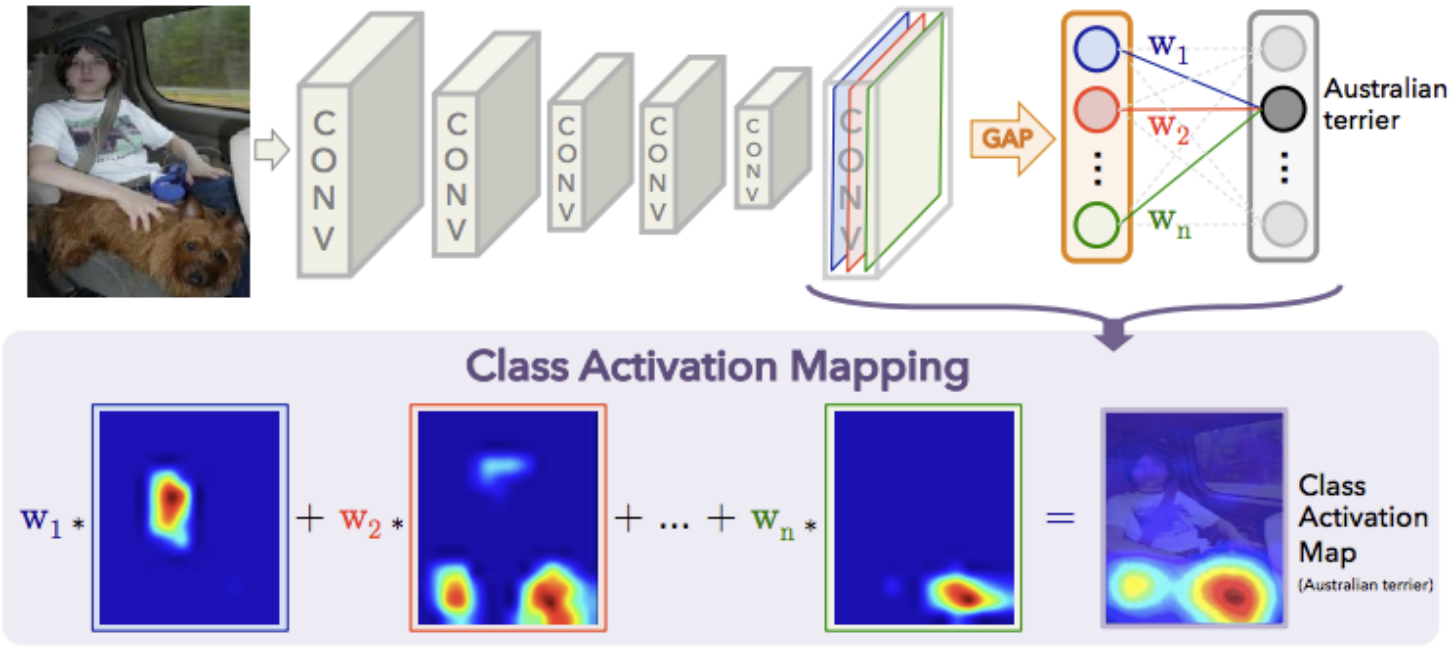
2016: Backprop to intermediate layers (III)

Creating this results:

DISTINCT CLASS SALIENCY MAPS FOR MULTIPLE OBJECT IMAGES

Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, Antonio Torralba

They revisit the global average pooling layer proposed in "Network In Network" (2014), and shed light on how it explicitly enables the CNNs to have remarkable localization ability despite being trained on imagelevel labels:



Methodology (I)

The way I see it, this method is a mix of CAM (from Agata), and backpropagate from a target class y^c to the last (and hence, more abstract) convolutional layer (from Simonyan).

First, they compute the importance of a kernel k in the last convolutional layer from the target class y^c to the activation produced by this kernel A^k:

$$\alpha_k^c = \overbrace{\frac{1}{Z}\sum_i\sum_j}^{\text{global average pooling}} \underbrace{\frac{\partial y^c}{\partial A_{ij}^k}}_{\text{gradients via backprop}}$$

Second, they use the weighted kernels activation, only the positive part, which indeed is the one that could increase the score of y^c:

Methodology (II)

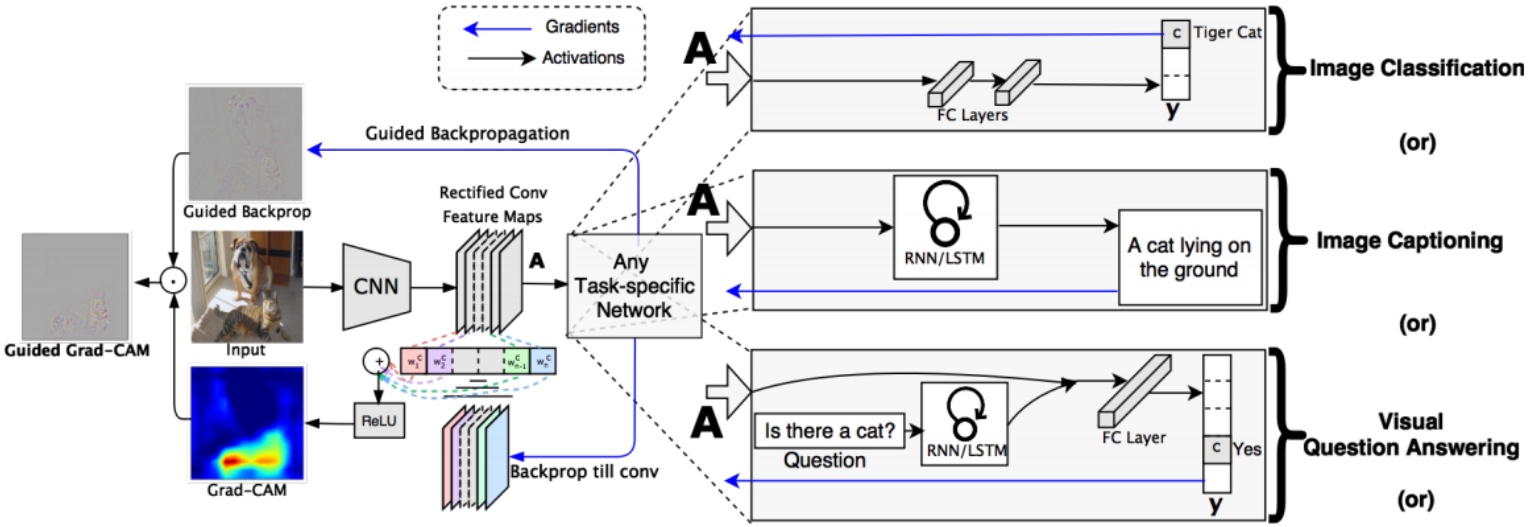
$$L^c_{\text{Grad-CAM}} = ReLU \underbrace{\left(\sum_k \alpha^c_k A^k \right)}_{\text{linear combination}}$$

After that, we just need to up-sampling to the image size.

The authors show that these technique is indeed a generalization of CAM. However, with grad-CAM y^c can be any differentiable activation including words from a caption or the answer to a question. Also, grad-CAM can be combined with guided-backprop in order to get high-definition discriminative visual explanations.

Methodology (III)

Grad-CAM as discriminative and guided-backprop as high-definition for any kind of problem:



Evaluating Localization and classification

Evaluation of the localization capability of Grad-CAM in the context of image classification over the ImageNet localization challenge which requires to provide bounding boxes in addition to classification labels:

Method	Top-1 loc error	Top-5 loc error	Top-1 cls error	Top-5 cls error
Backprop on VGG-16 [44]	61.12	51.46	30.38	10.89
c-MWP on VGG-16 [50]	70.92	63.04	30.38	10.89
Grad-CAM on VGG-16 (ours)	56.51	46.41	30.38	10.89
VGG-16-GAP (CAM) [51]	57.20	45.14	33.40	12.20

Evaluating class discrimination and trustful

In which of the following models will you trust more?

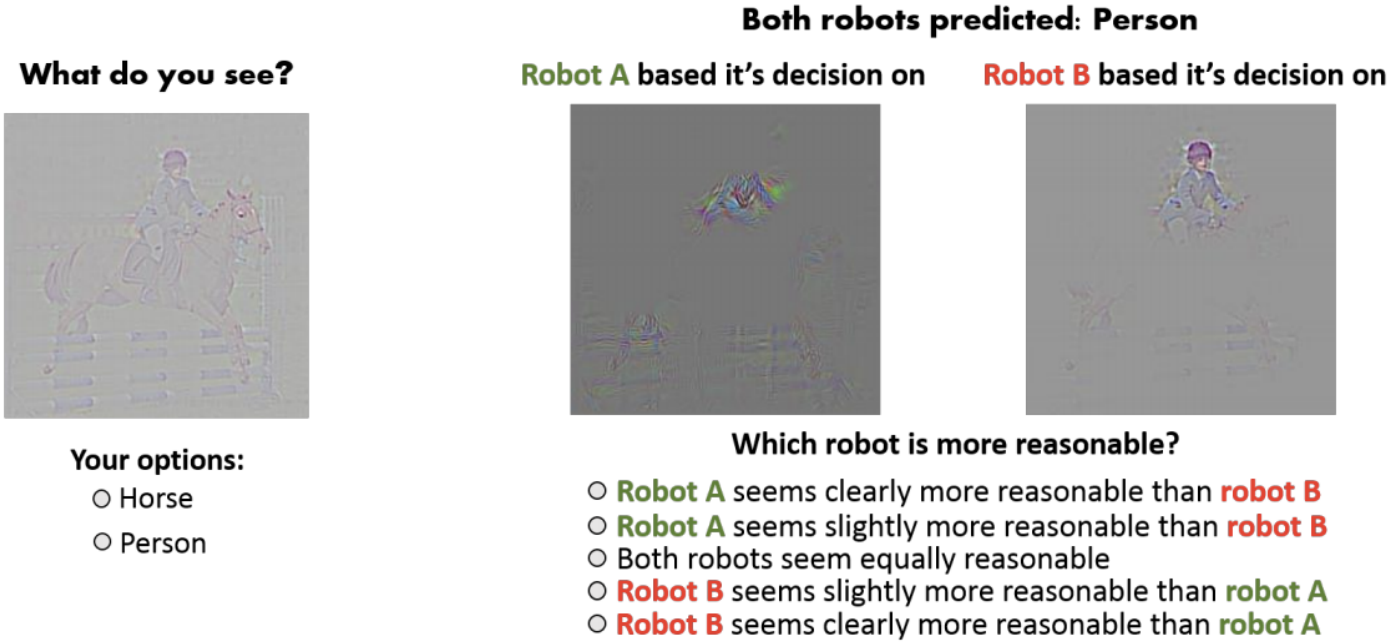


Figure 3: AMT interfaces for evaluating different visualizations for class discrimination (left) and trust worthiness (right). Guided Grad-CAM outperforms baseline approaches (Guided-backprop and Deconvolution) showing that our visualizations are more class-discriminative and help humans place trust in a more accurate classifier.

Discovering bias in dataset

They finetune an ImageNet trained VGG-16 model for the task of classifying "doctor" vs. "nurse" which did not generalize as well (82%). Grad-CAM visualizations of the model predictions revealed that the model had learned to look at the person's face/hairstyle to distinguish nurses from doctors, thus learning a gender stereotype. Indeed, in the training dataset 78% of images for doctors were men, and 93% images for nurses were women. Grad-CAM discovered it:

Ground-Truth: Doctor

(g) Original Image

Predicted: Nurse

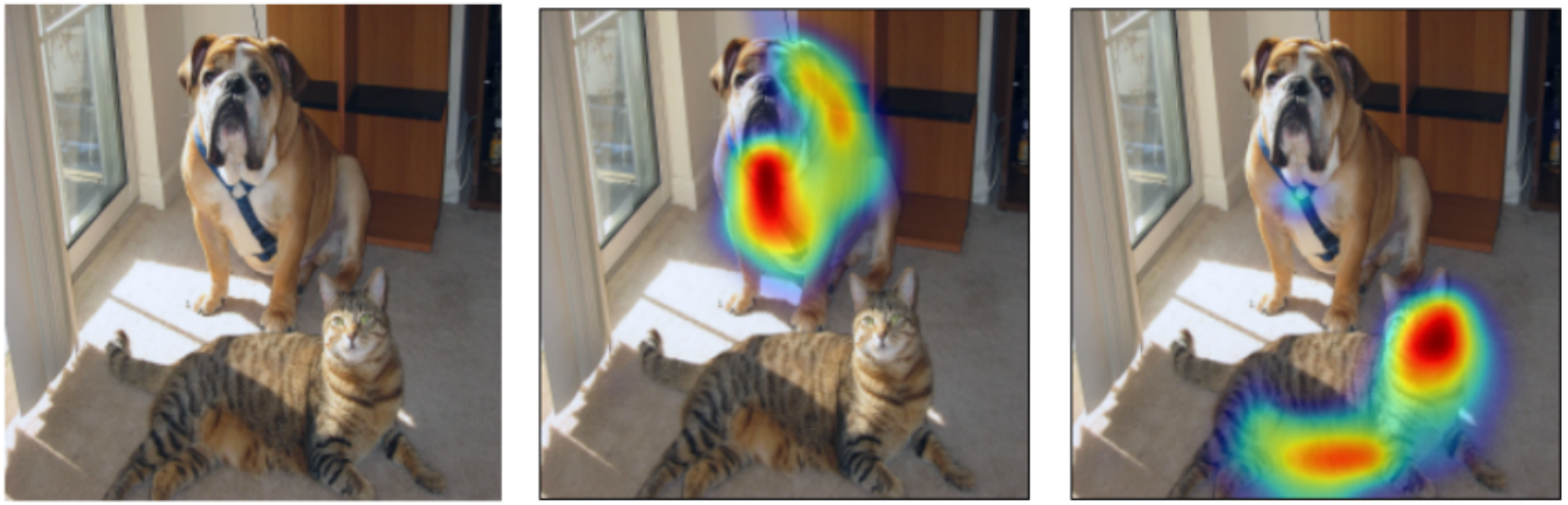
(h) Grad-CAM for biased model

Predicted: Doctor

(i) Grad-CAM for unbiased model

Counterfactual Explanations

Which are the regions that would make the network change its decision more?



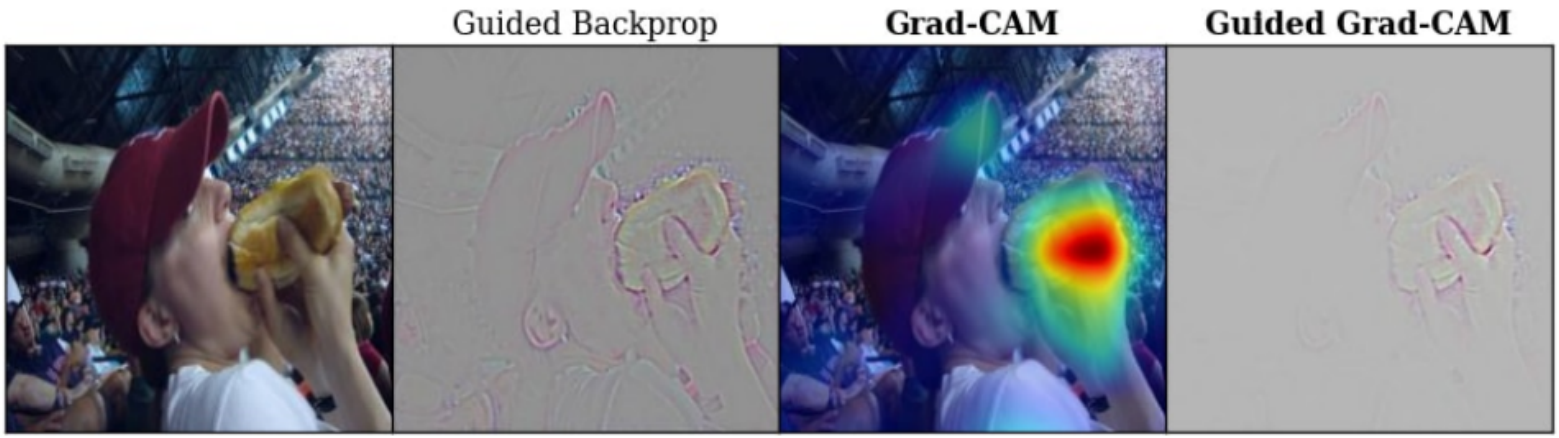
(a) Original Image (b) Cat Counterfactual exp (c) Dog Counterfactual exp

Figure 6: Negative Explanations with Grad-CAM

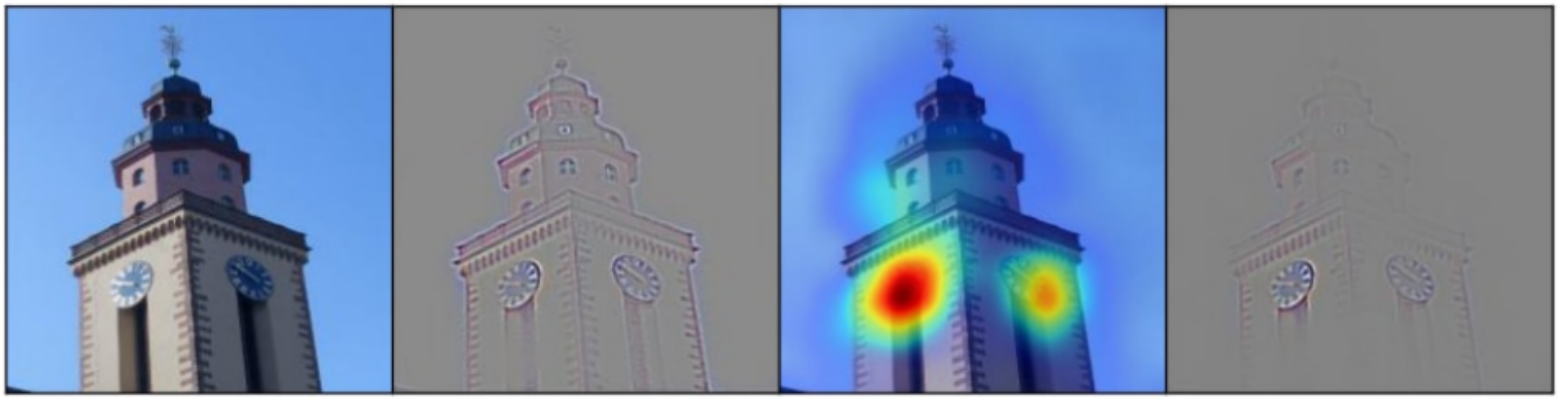
Gradient negations finds locations that drops the score:

$$\alpha_k^c = \overbrace{\frac{1}{Z}\sum_i\sum_j}^{\text{global average pooling}} \underbrace{-\frac{\partial y^c}{\partial A_{ij}^k}}_{\text{Negative gradients}}$$

Image captioning

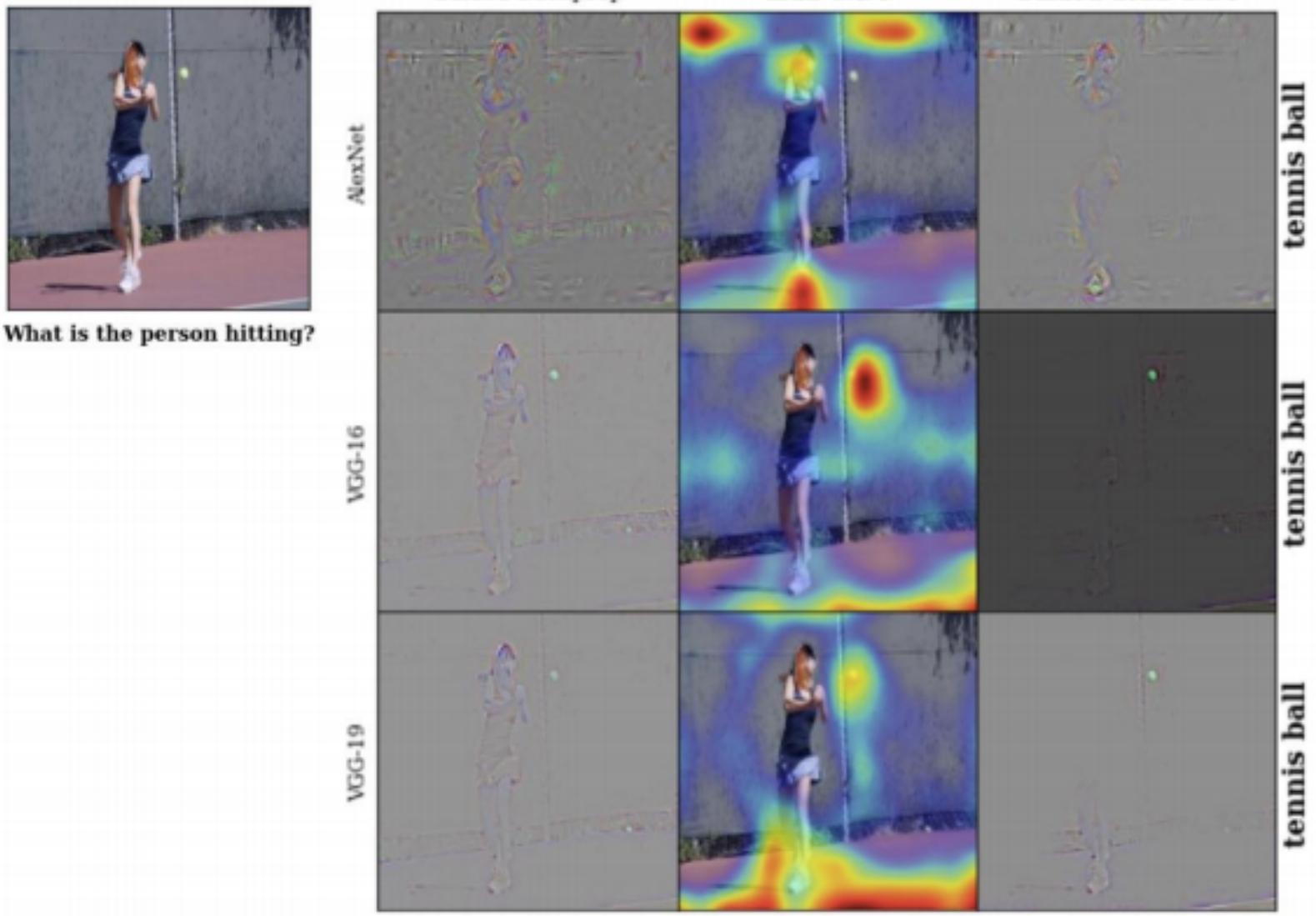


A man is holding a hot dog in his hand



A large clock tower with a clock on the top of it

Visual Question Answering



Pros/cons & Future Work

Pros: It is a very general model which uses almost everything I mentioned today and applies to almost all kind of problems. It is easy to use, cheap, and transparent.

Cons: To really understand this method there are a bunch of previous method that you should master before.

Future work: Following the lines of the papers shown today, I will try to improve the current grad-CAM adding:

- Aggregating Grad-CAM visualizations over different layers, not just relying in the last one.
- Subtracting Grad-CAM of the other classes from Grad-CAM of the target class to differentiate target objects from other objects.