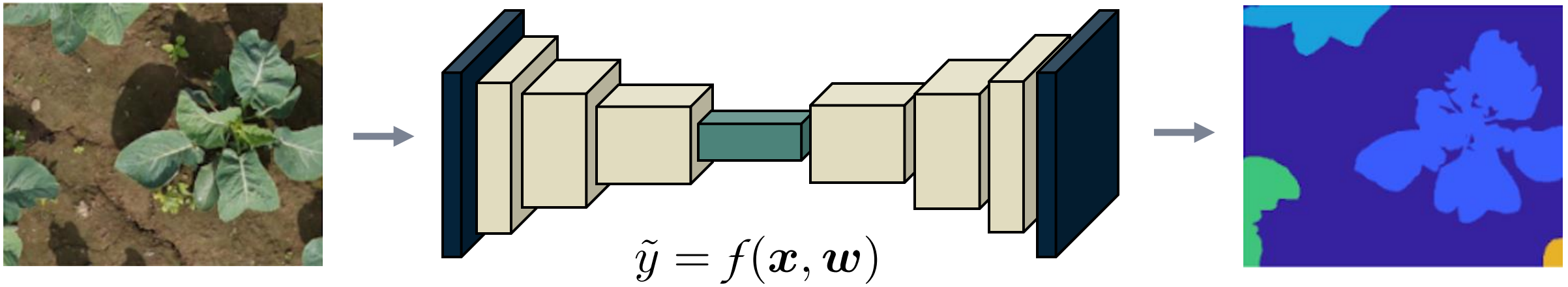
Explainable machine learning

Looking into a neural network - visualizing what was learned

Ribana Roscher

These slides have been created by Ribana Roscher.

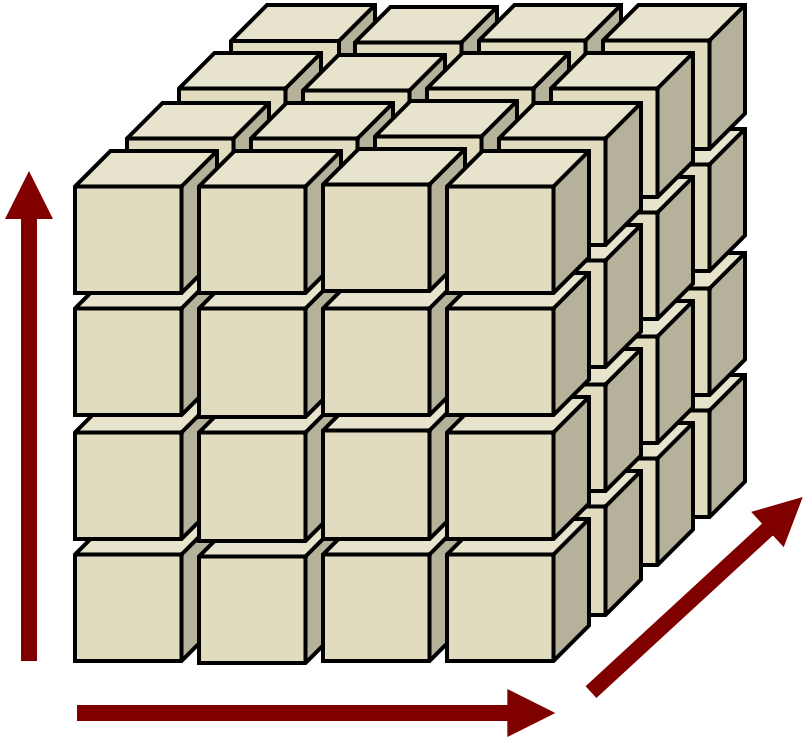
Deep neural networks visualization



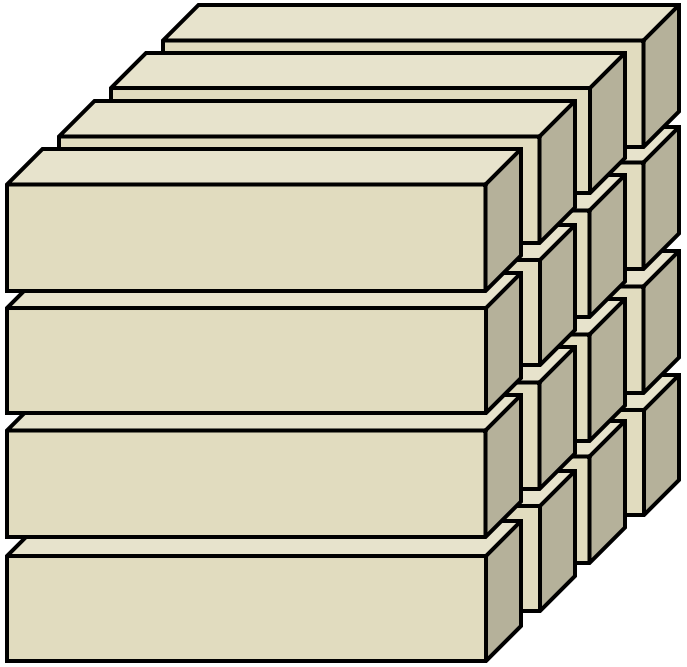
One block illustrates the **activation maps**, i.e., the outcome after applying one (or several) mathematical operations → new representation

Images from: Kierdorf, J., Junker-Frohn, L.V., Delaney, M., Donoso-Olave, M., Burkhart, A., Jeanicke, H., Müller, O. & Roscher, R. (2022). GrowliFlower: A time series dataset for GROWth analysis of cauliflower, accepted.

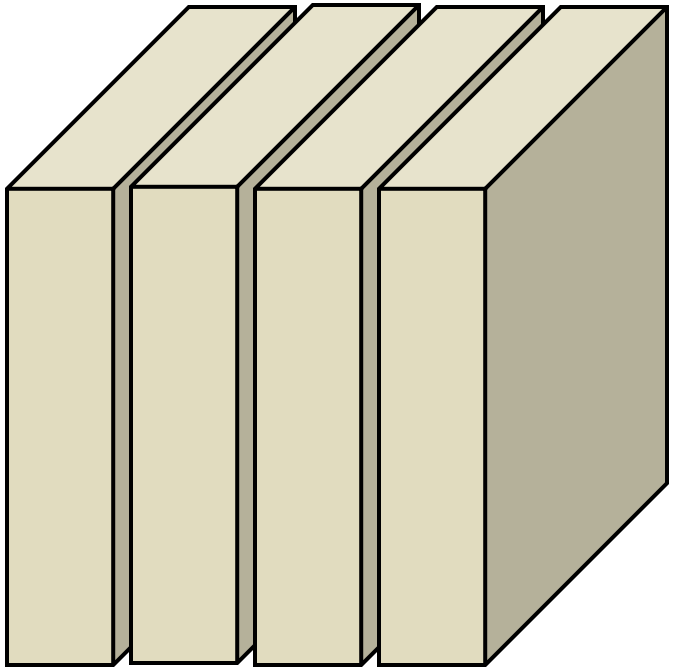
individual activations



spatial activations



channel activations

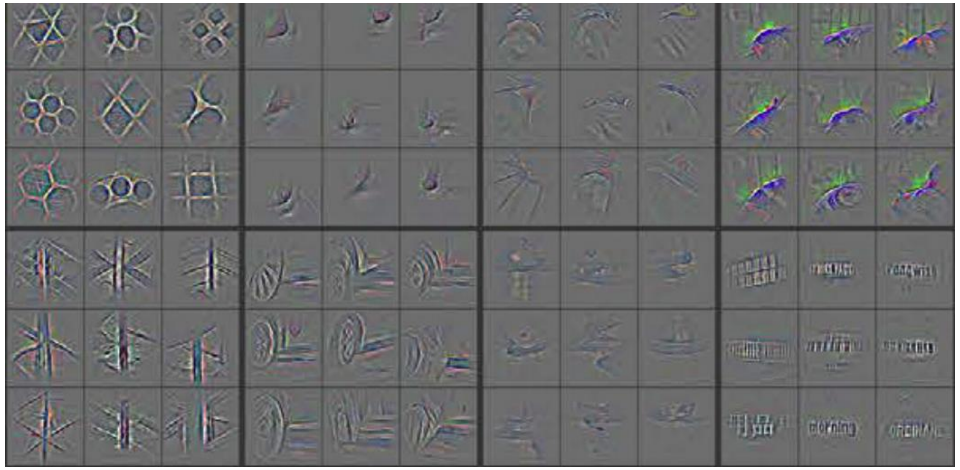


How can we look into a network?

Feature visualization

What does a network look for?

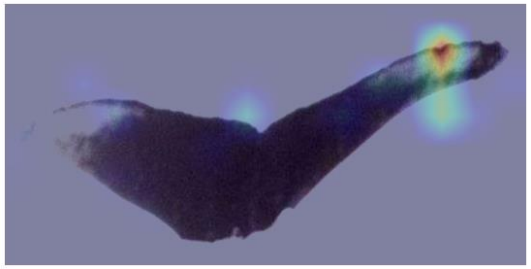
- Visualize model components
- Selecting samples
- Generating examples



Feature attribution

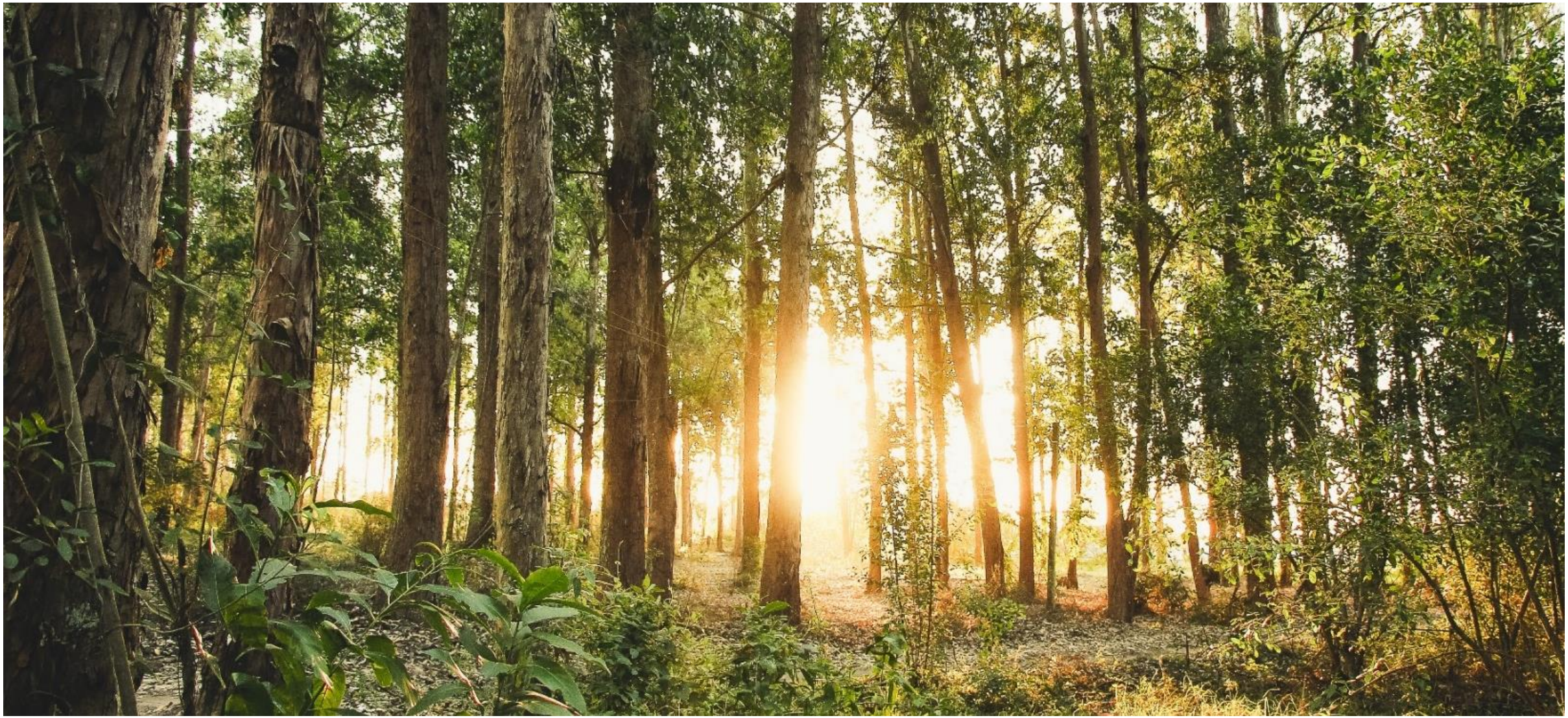
Which part of an example is responsible for the network behaving in a particular way?

- Highlighting examples or parts of them



Visualizing model components

Discover wilderness characteristics



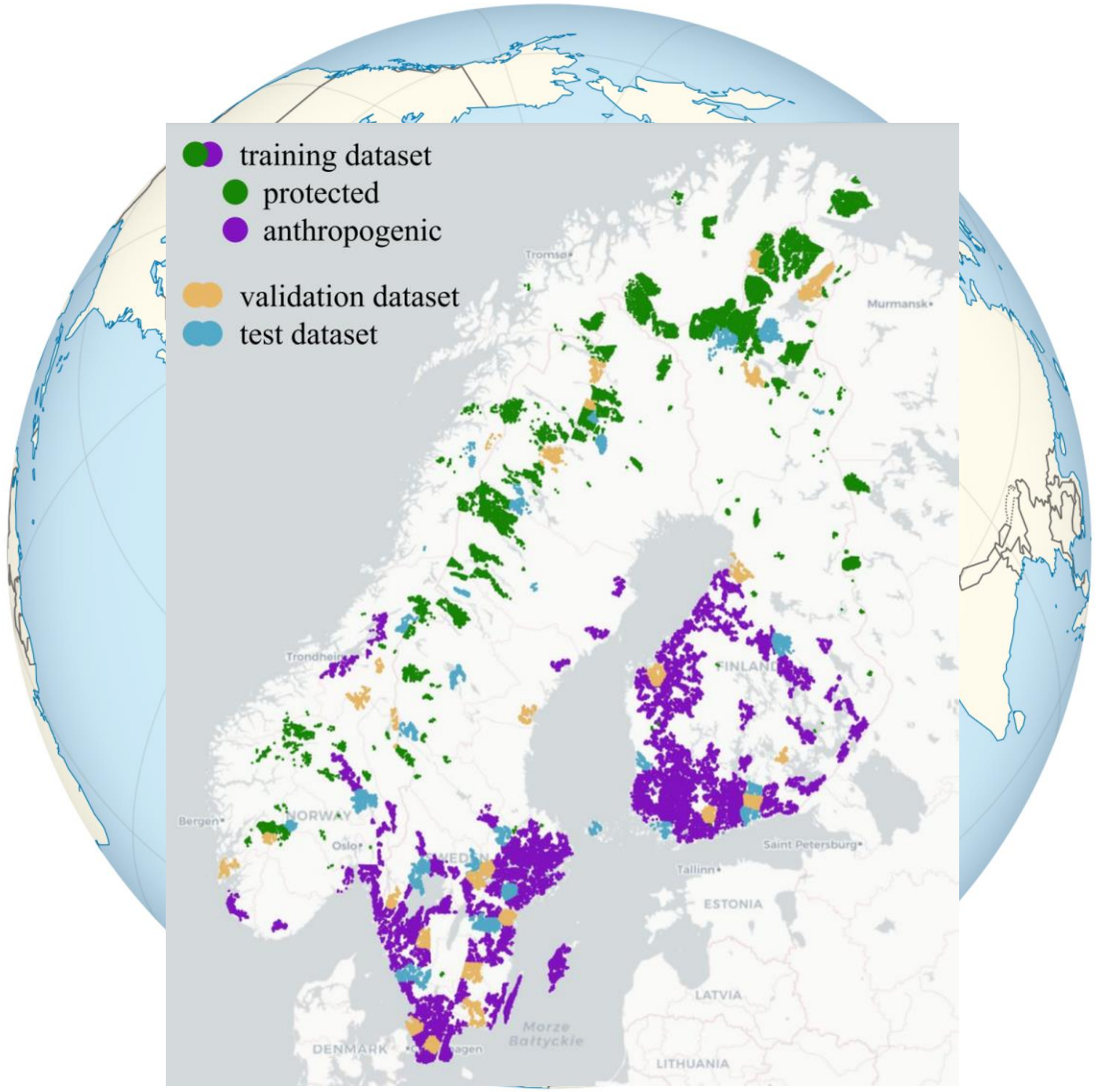
Study site

World Database on Protected Areas (WDPA)

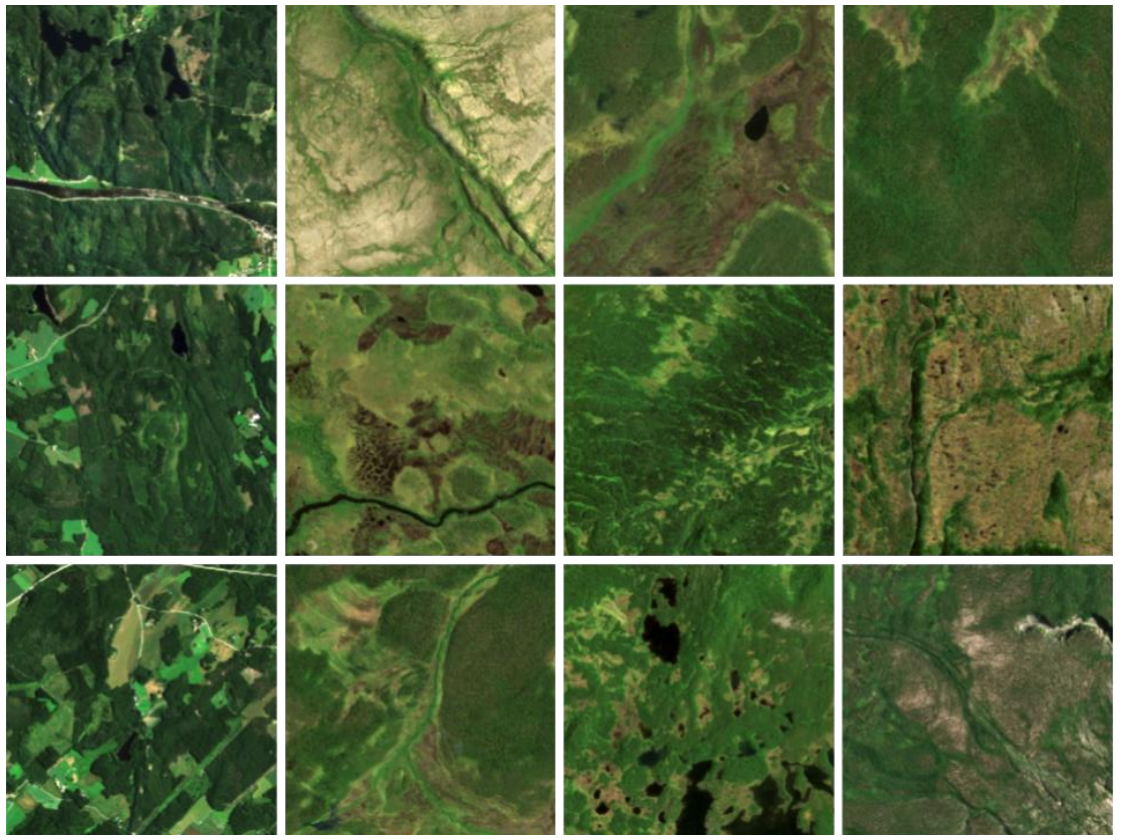
vs.

Anthropogenic Areas

(artificial and agricultural surfaces)



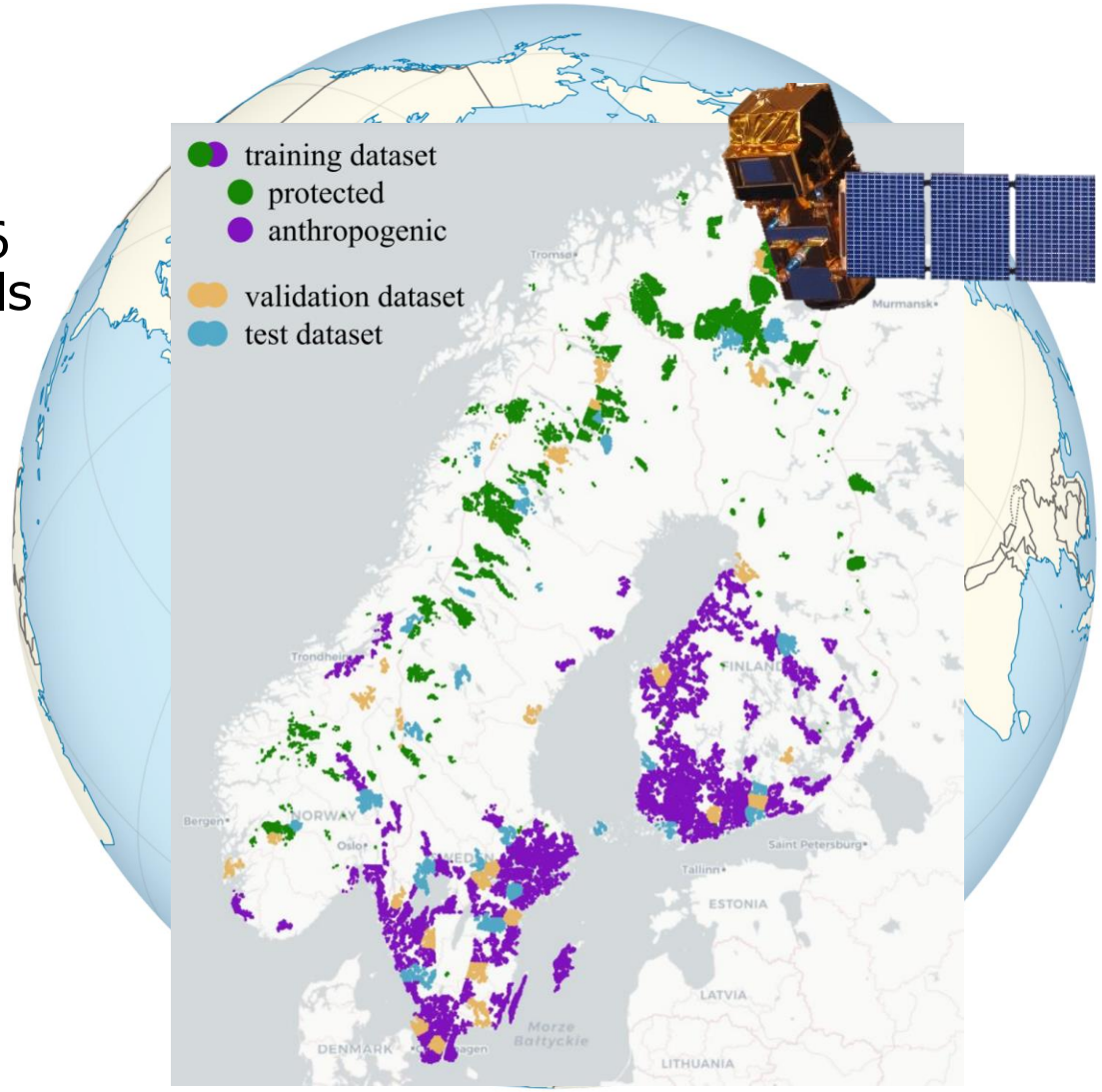
Sentinel-2 data



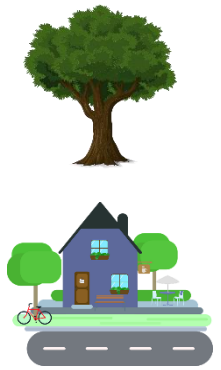
256 pixels

anthropogenic

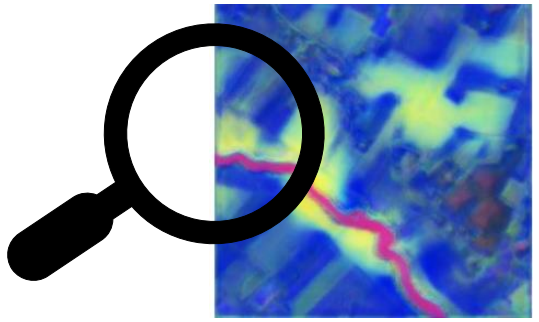
protected



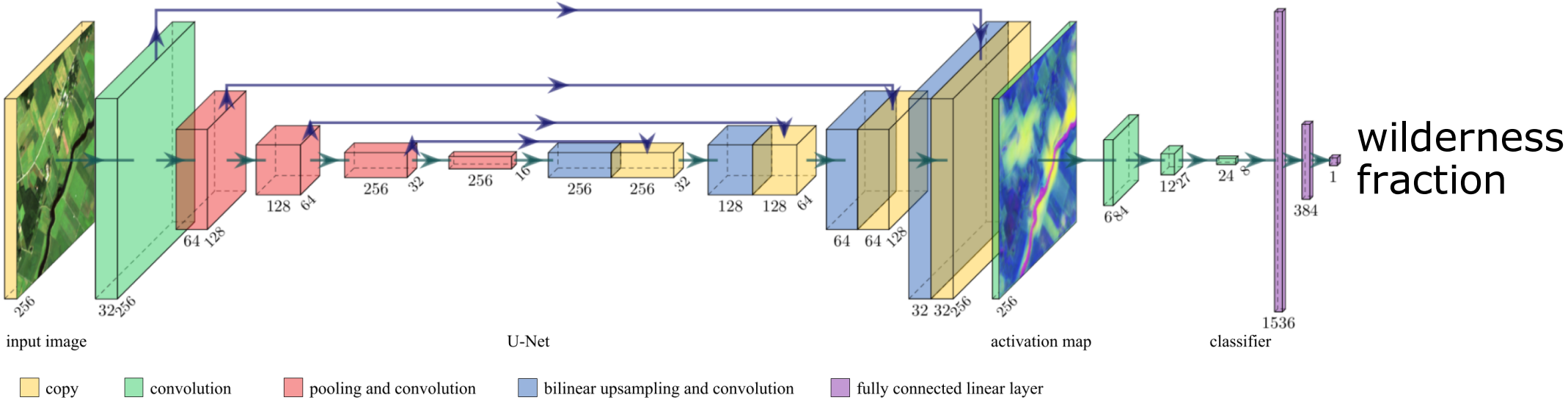
Network to analyze: jUngle-Net



multispectral Sentinel-2 images (mixed with CutMix)



D-dimensional activation maps



wilderness fraction

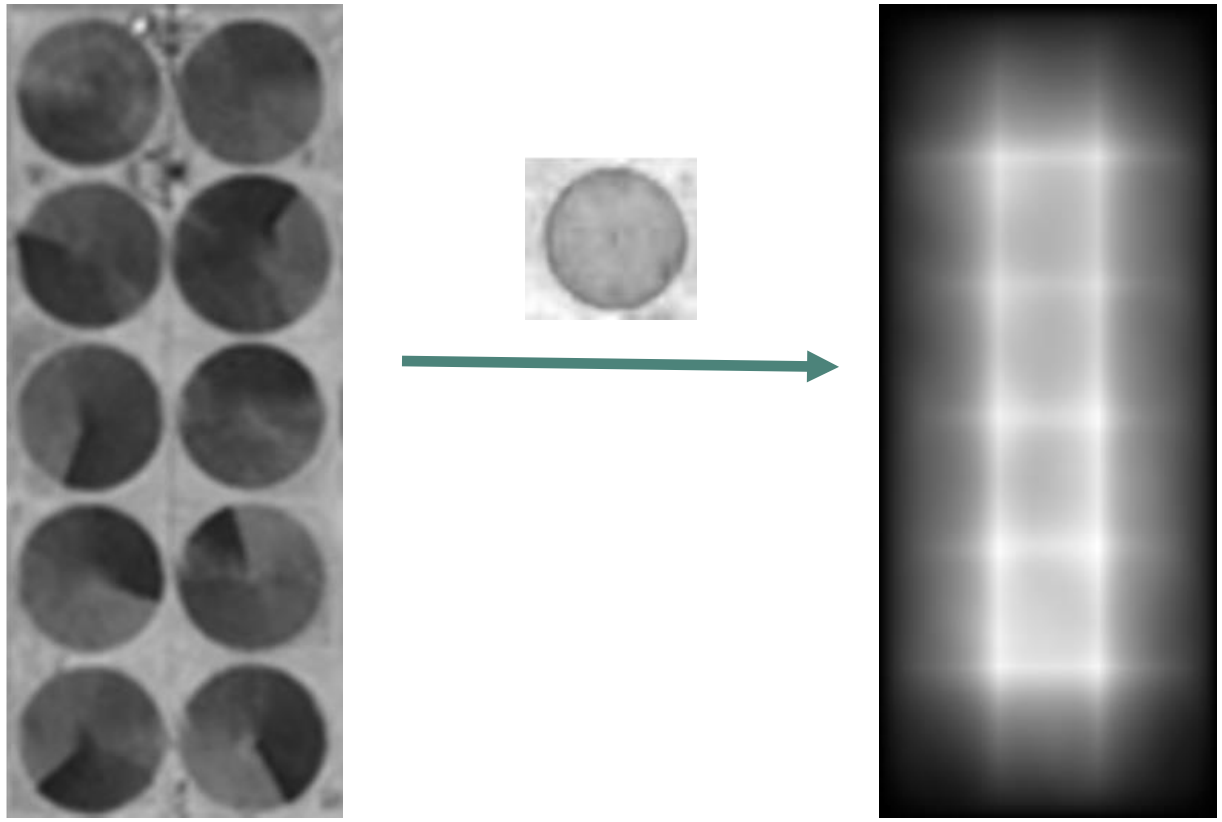
Stomberg, T., Weber, I., Schmitt, M., & Roscher, R. (2021). jUngle-Net: Using explainable machine learning to gain new insights into the appearance of wilderness in satellite imagery. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *3*, 317-324.
Stomberg, T. T., Stone, T., Leonhardt, J., & Roscher, R. (2022). Exploring Wilderness Using Explainable Machine Learning in Satellite Imagery. *arXiv preprint arXiv:2203.00379*.

What can be visualized?

- Intermediate **activations**
  - What the network saw, i.e. what was extracted from the input
- **Weights/filters**
  - Patterns each layer has learned to extract from the input
  - What is used to compute an output
  - Generally used for convolutions (spatially arranged weights)

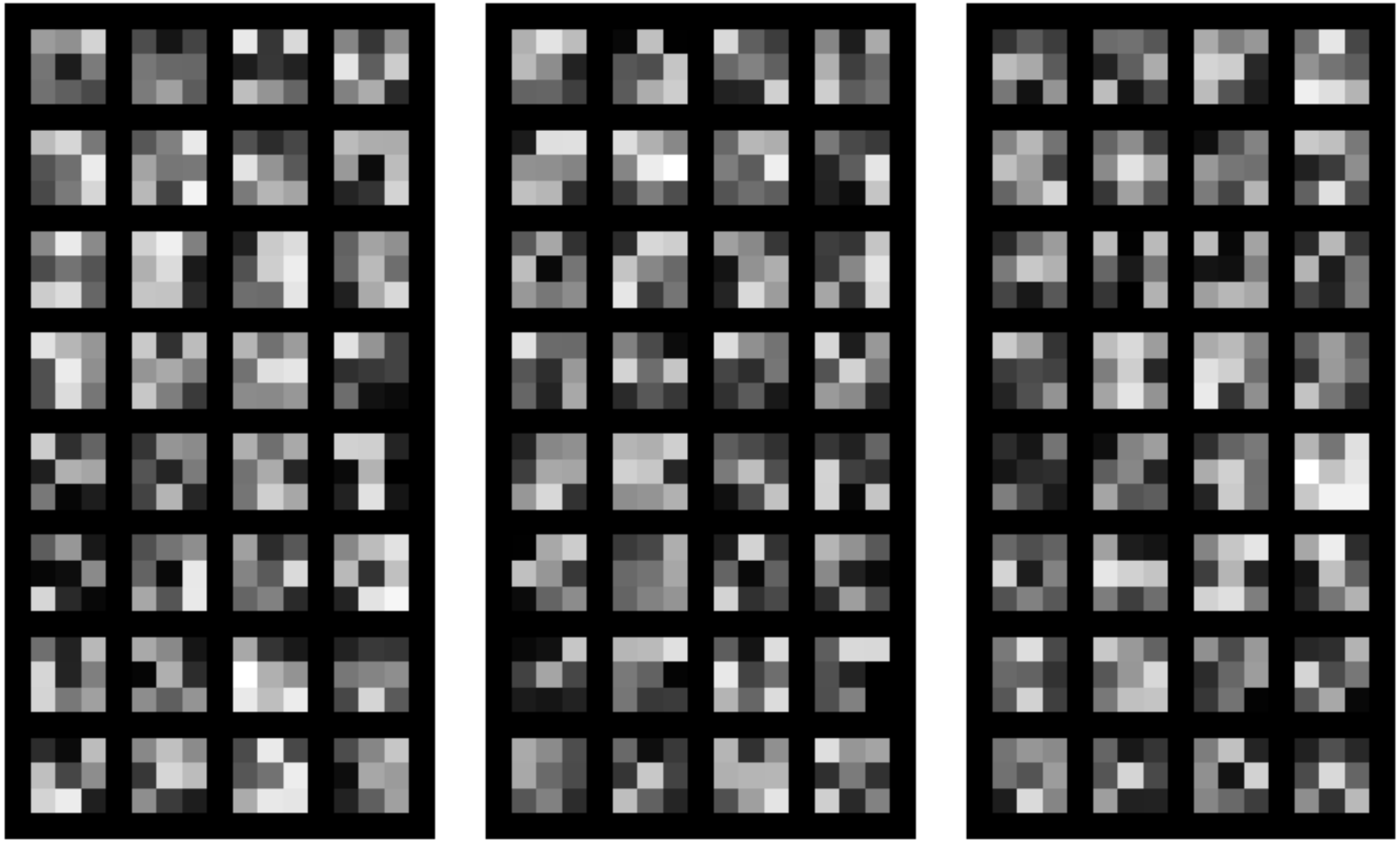
➢ Activations are much easier to study than weights

Visualizing filters

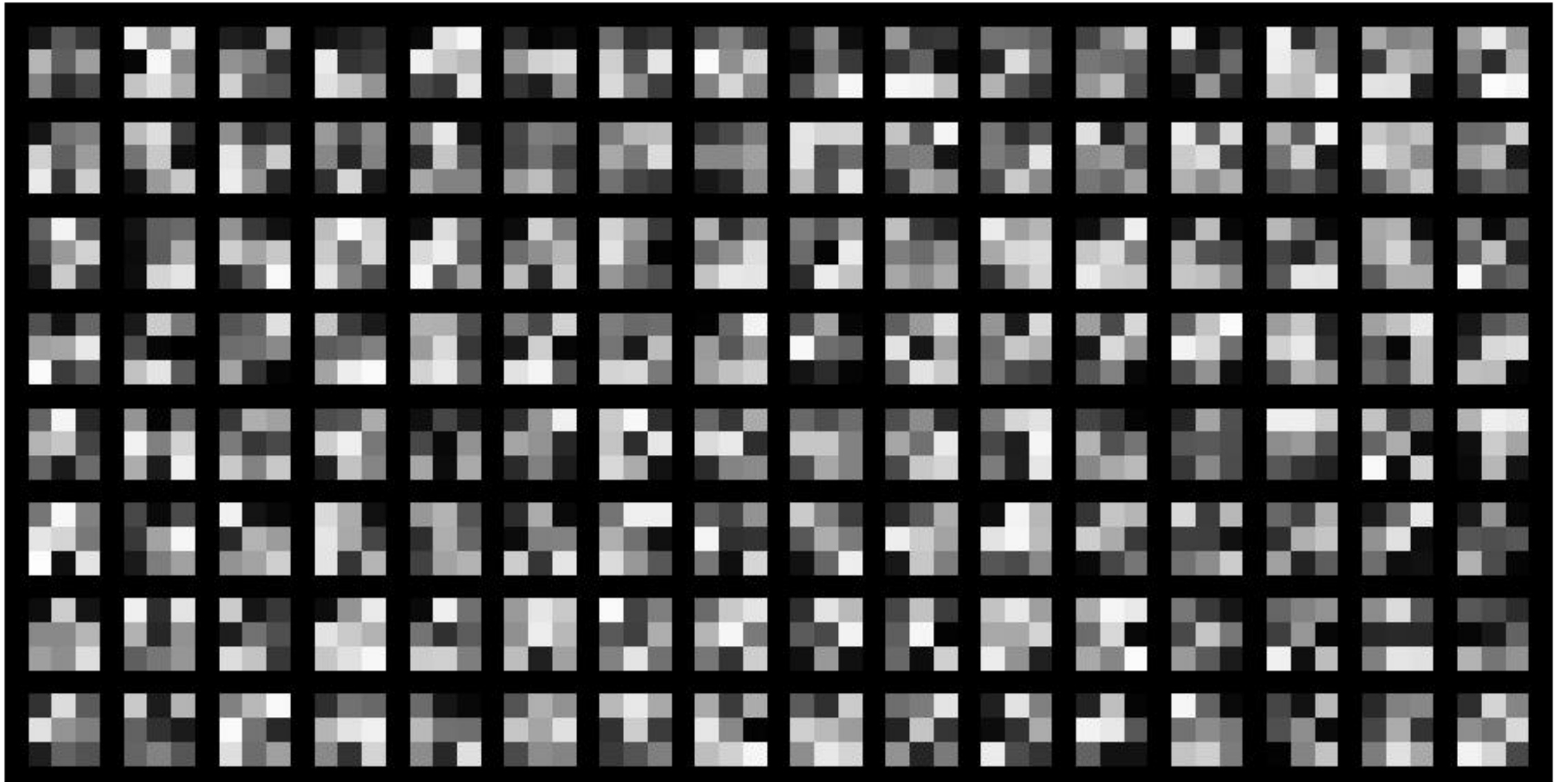


high values in filters result in high activations if the pattern is similar

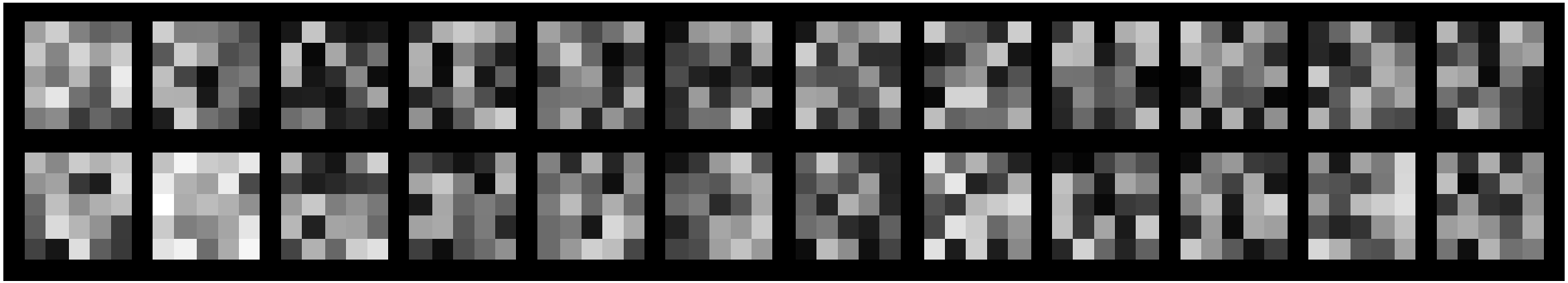
First layer

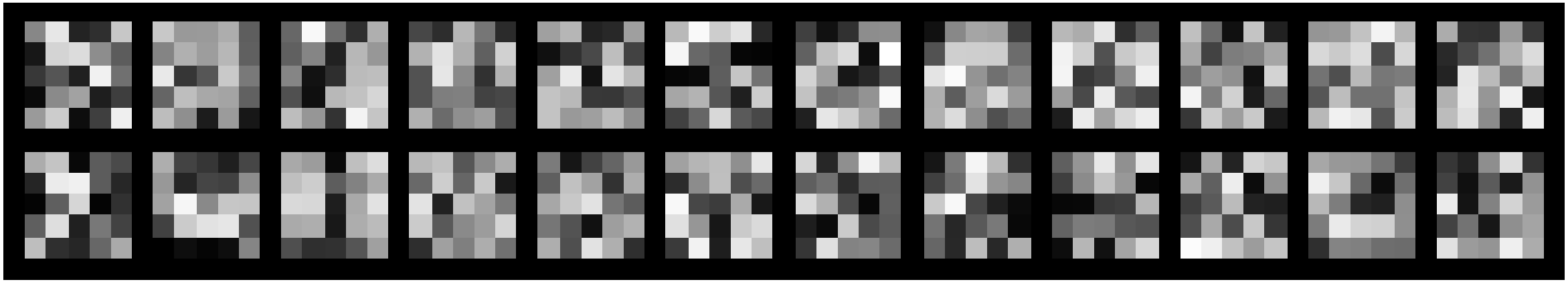


Layer 4



Layer 12





Visualizing weights and filters

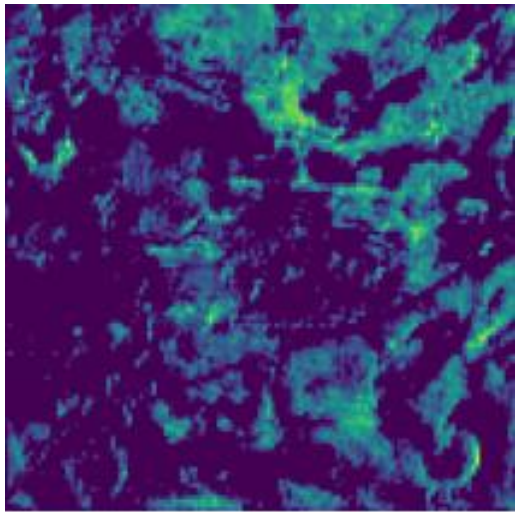
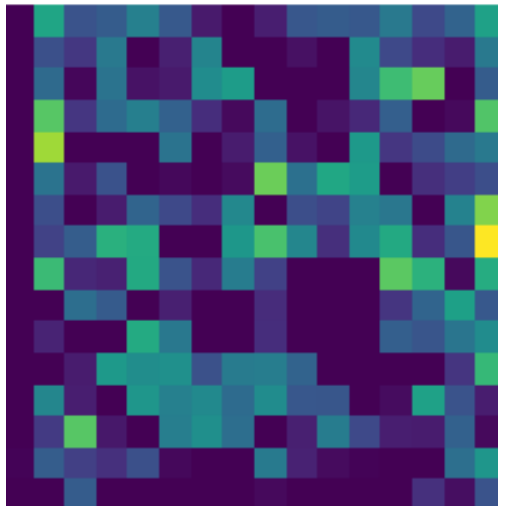
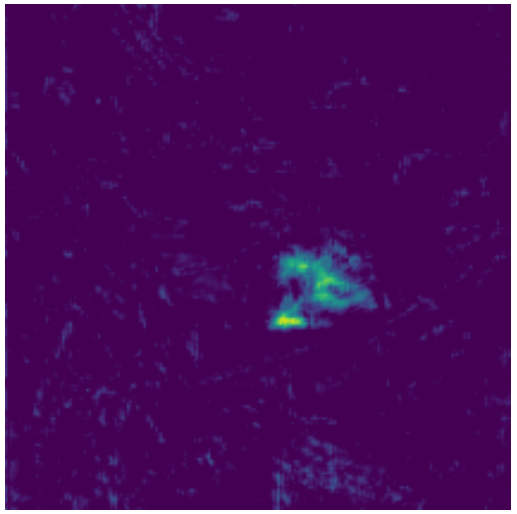
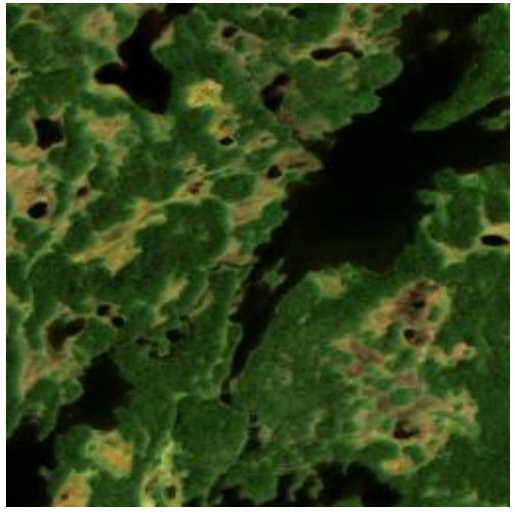
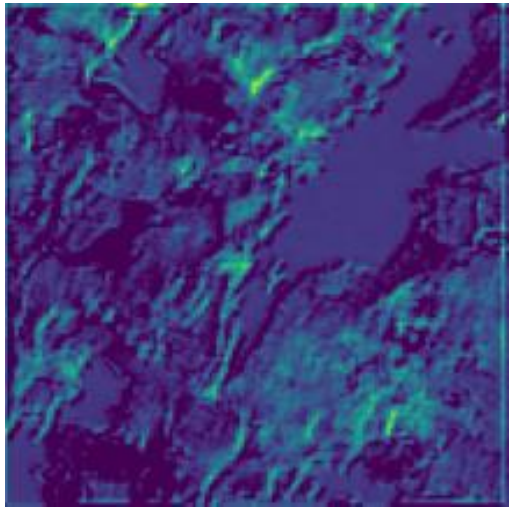
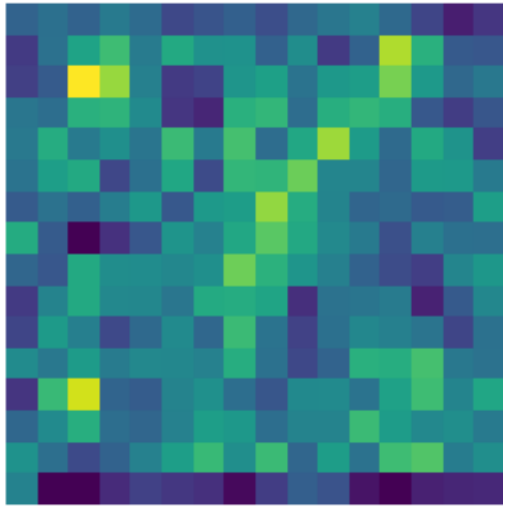
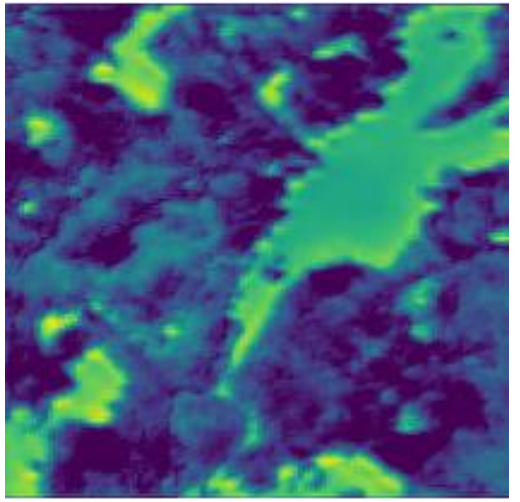
Lack of contextualization
Visualization is meaningful when linked to an input (e.g. RGB values)

Indirect interaction
Neurons can interact on a complex way throughout the network

Dimensionality and scale
Number of weights is generally very high, therefore visualizing single neurons is overwhelming

Voss, et al., "Visualizing Weights", Distill, 2021.

Visualizing intermediate activations

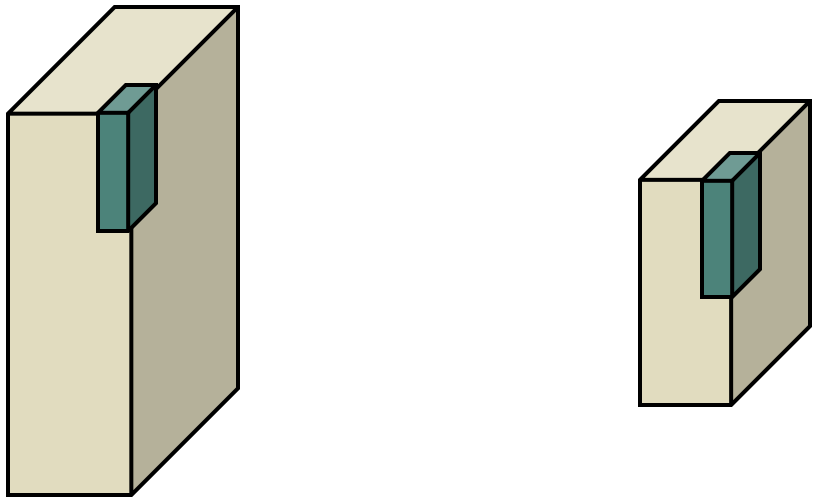
original image | shallow layer | bottleneck layer | deep layer

Visualizing intermediate activations

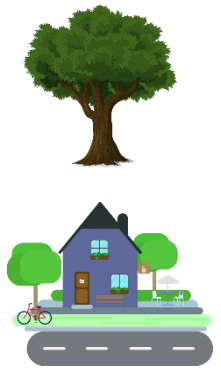
- Earlier layer focus on general features like edges
- Deeper layers are difficult to interpret visually
  - They focus on discriminating features when uses for classification
  - Only important information is retained – information distillation
- Blank activations (with zero value) indicate that these features could not be found in the inputs

Receptive field

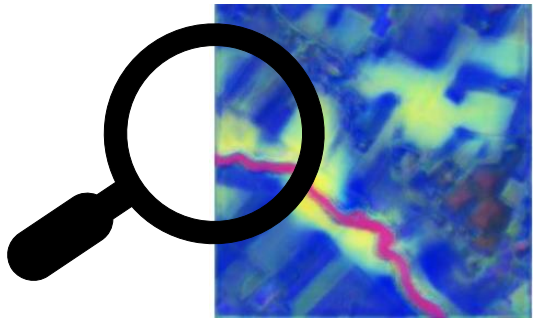
- The convolutional filter size stays the same, but due to downsampling the receptive field size increase
- An activation in deeper layers captures more spatial information



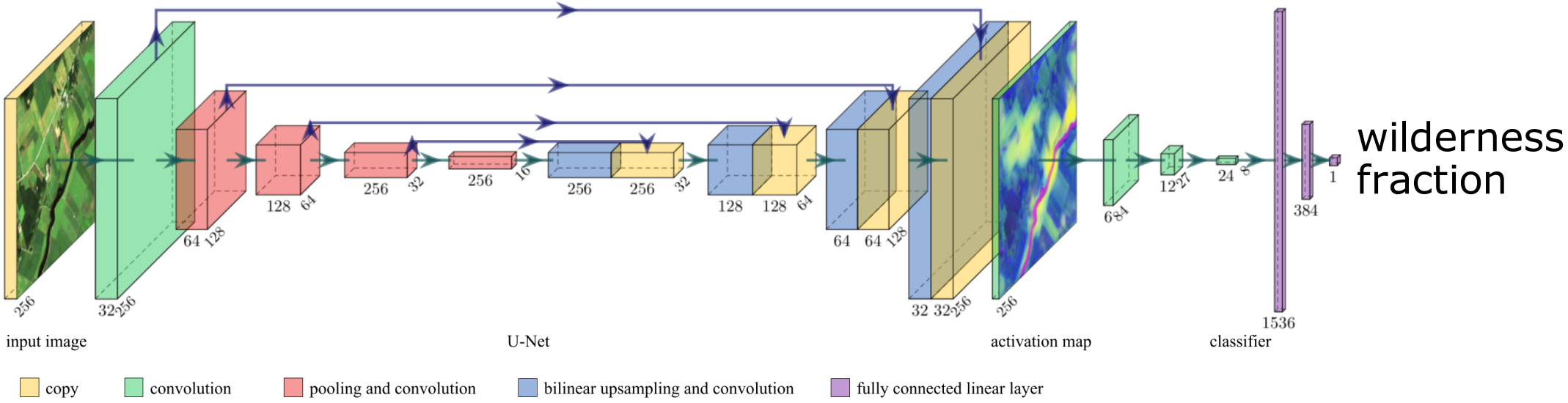
Conceptual framework (jUngle-Net)



multispectral Sentinel-2 images (mixed with CutMix)



D-dimensional activation maps

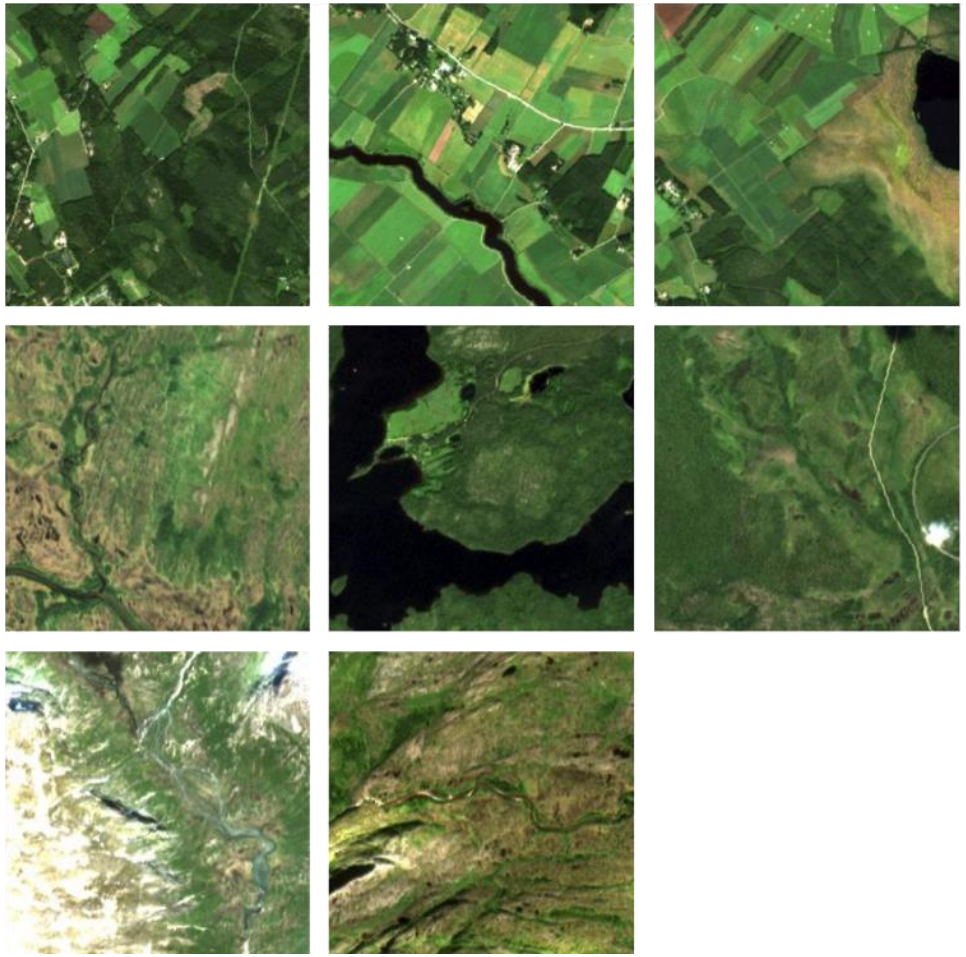


wilderness fraction

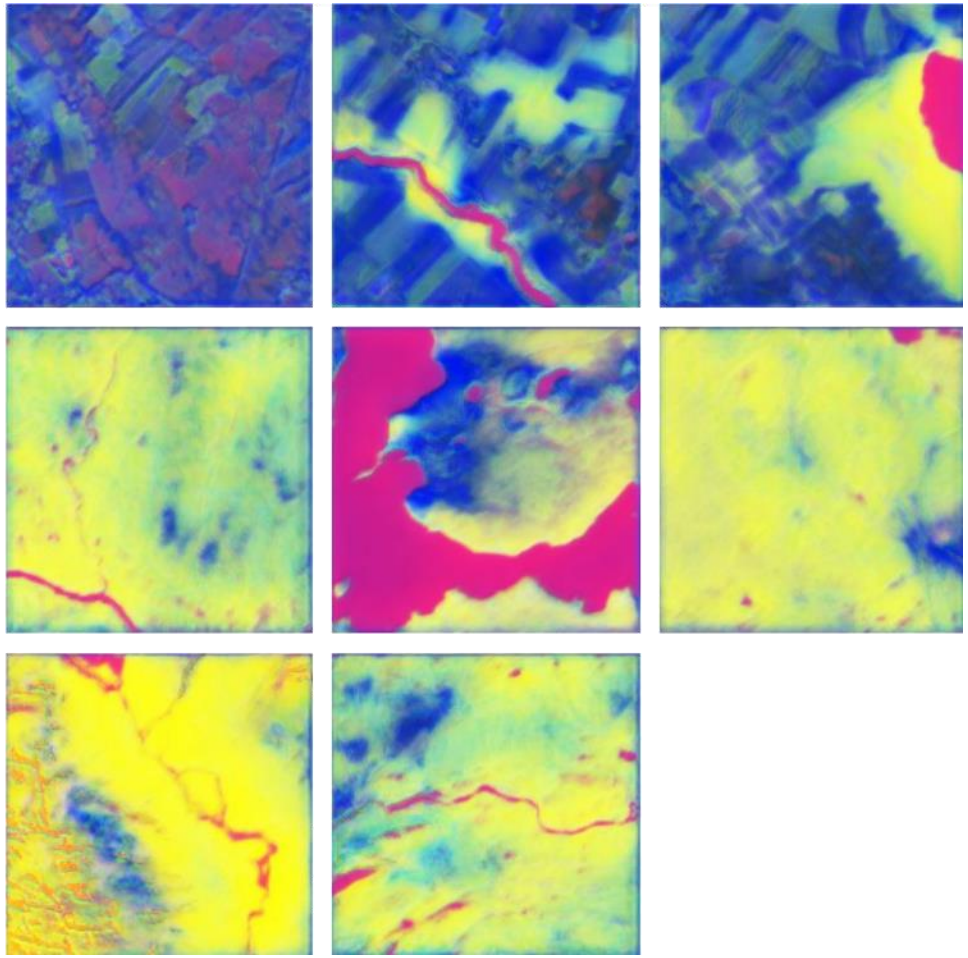
Stomberg, T., Weber, I., Schmitt, M., & Roscher, R. (2021). jUngle-Net: Using explainable machine learning to gain new insights into the appearance of wilderness in satellite imagery. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *3*, 317-324.
Stomberg, T. T., Stone, T., Leonhardt, J., & Roscher, R. (2022). Exploring Wilderness Using Explainable Machine Learning in Satellite Imagery. *arXiv preprint arXiv:2203.00379*.

Activation maps

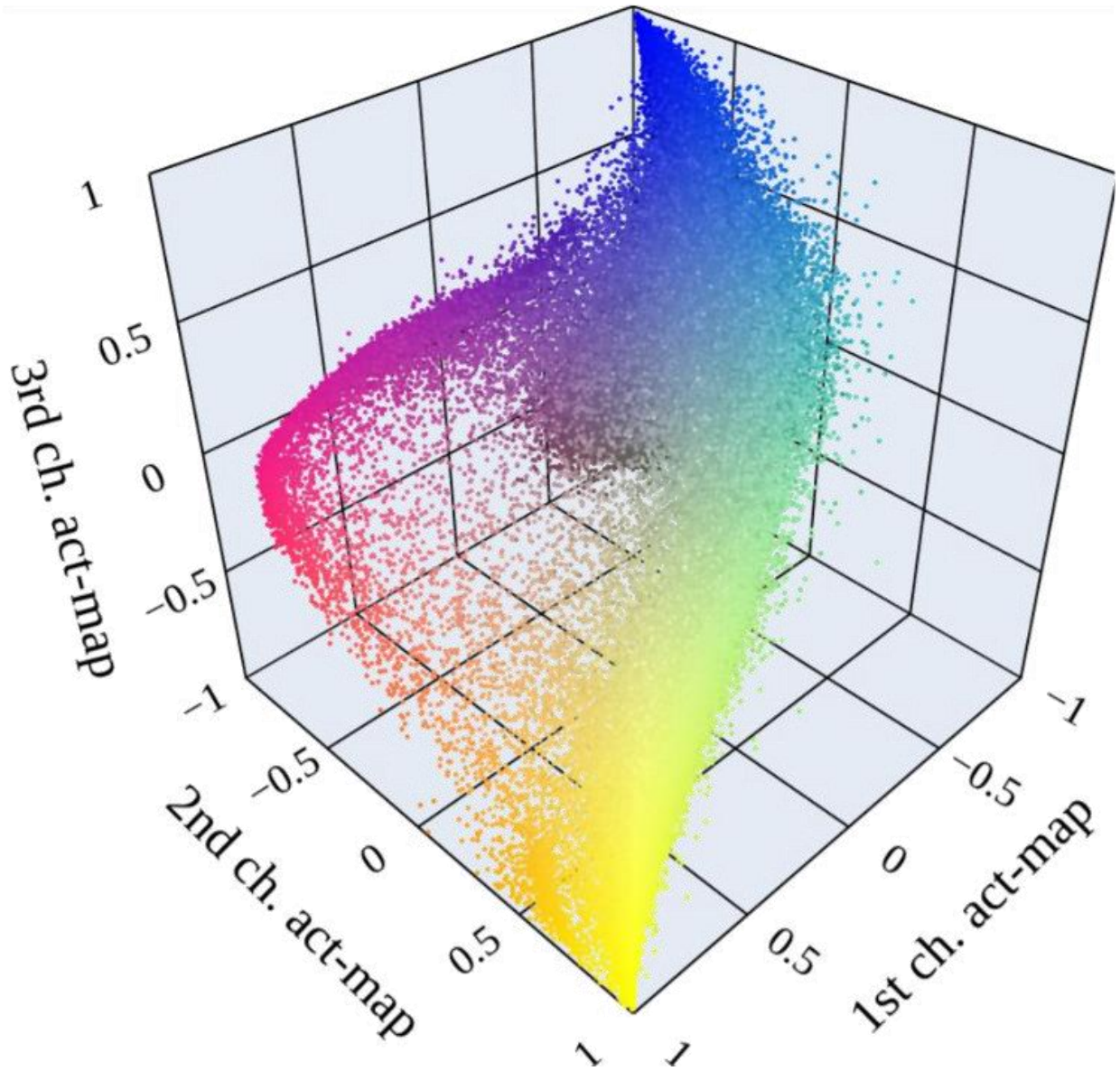
19.123 training samples



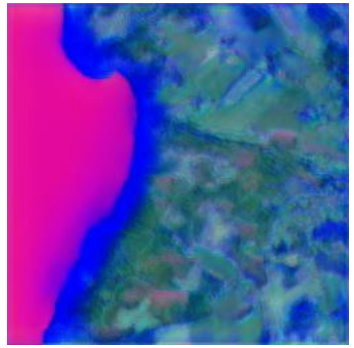
activation maps (3 channels)

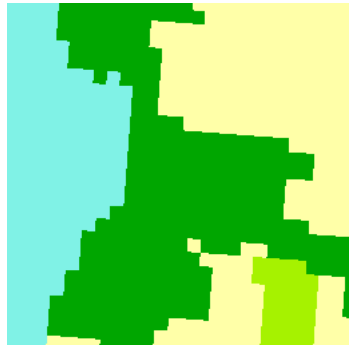


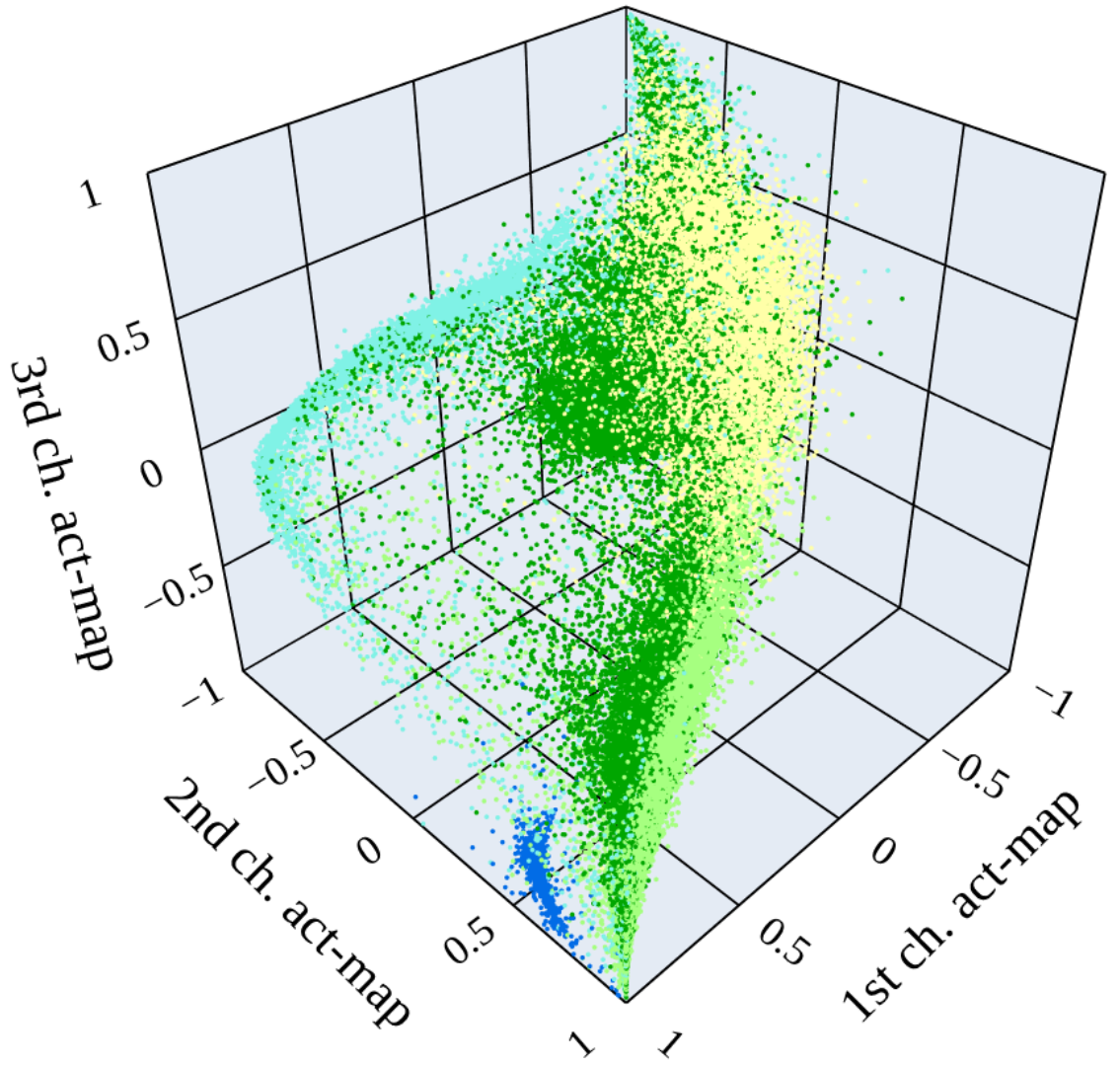
Activation space



Connecting label information

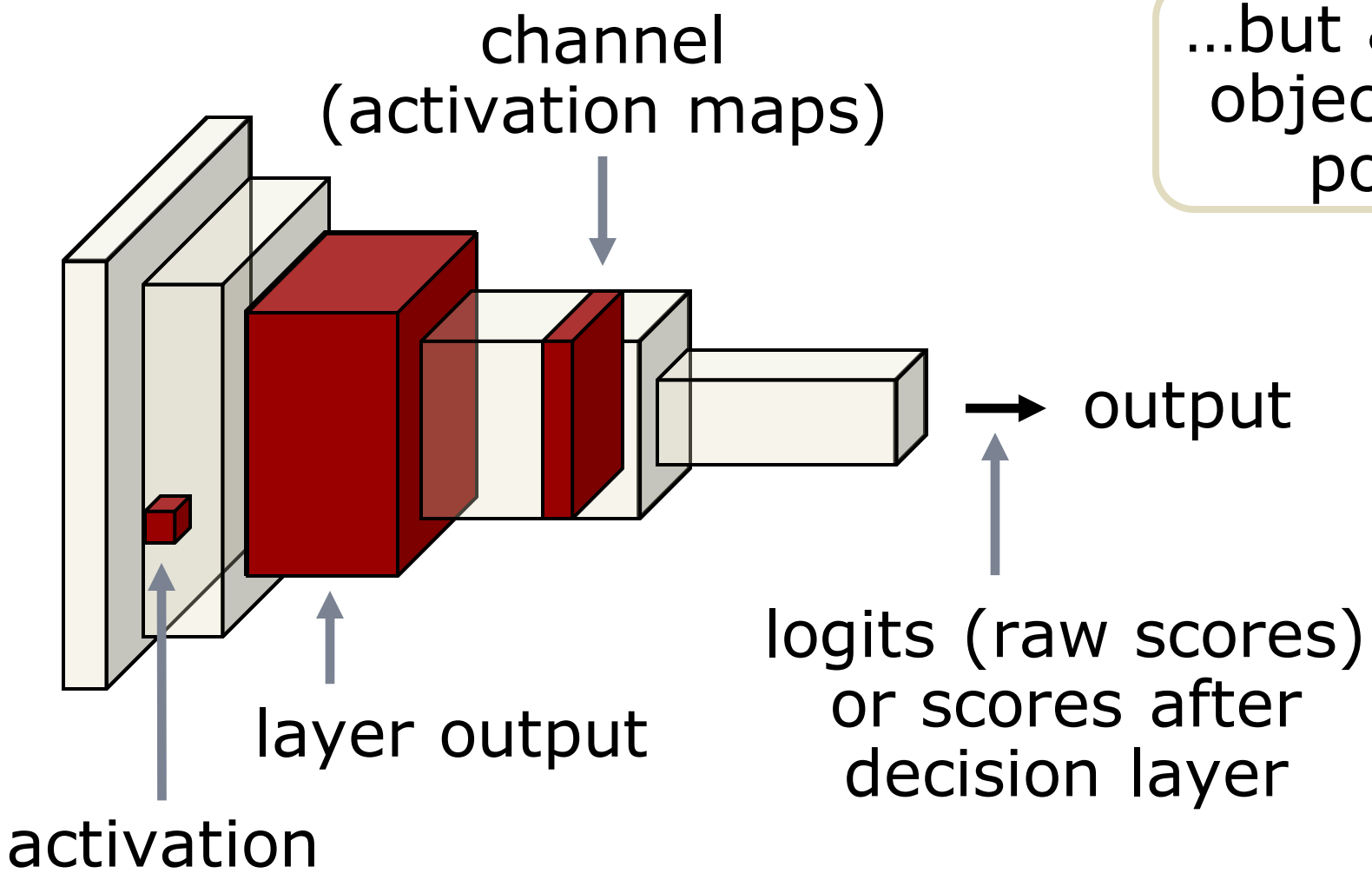






Feature visualization by selecting dataset samples and generating examples

Visualizing activations



…but also other objectives are possible

Feature visualization by selecting dataset samples

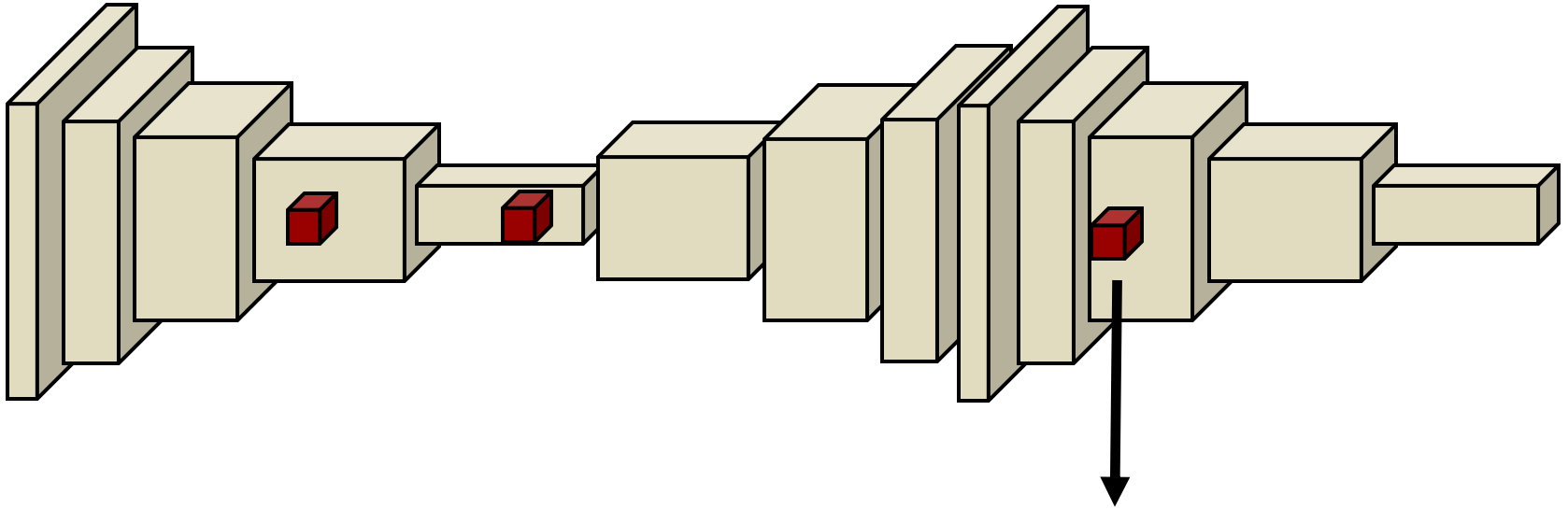
Highly-activating samples

Select samples from the dataset that **maximize a specific activation** or a set of activations

$$X^* = \arg\max_X a(X), \qquad X \in \mathcal{D}$$

- Can be extended to maps and layers by summing over the activations

Highly-activating samples



Highly-activating samples

Advantages

- Samples are realistic
- Easy to “compute”

Disadvantages

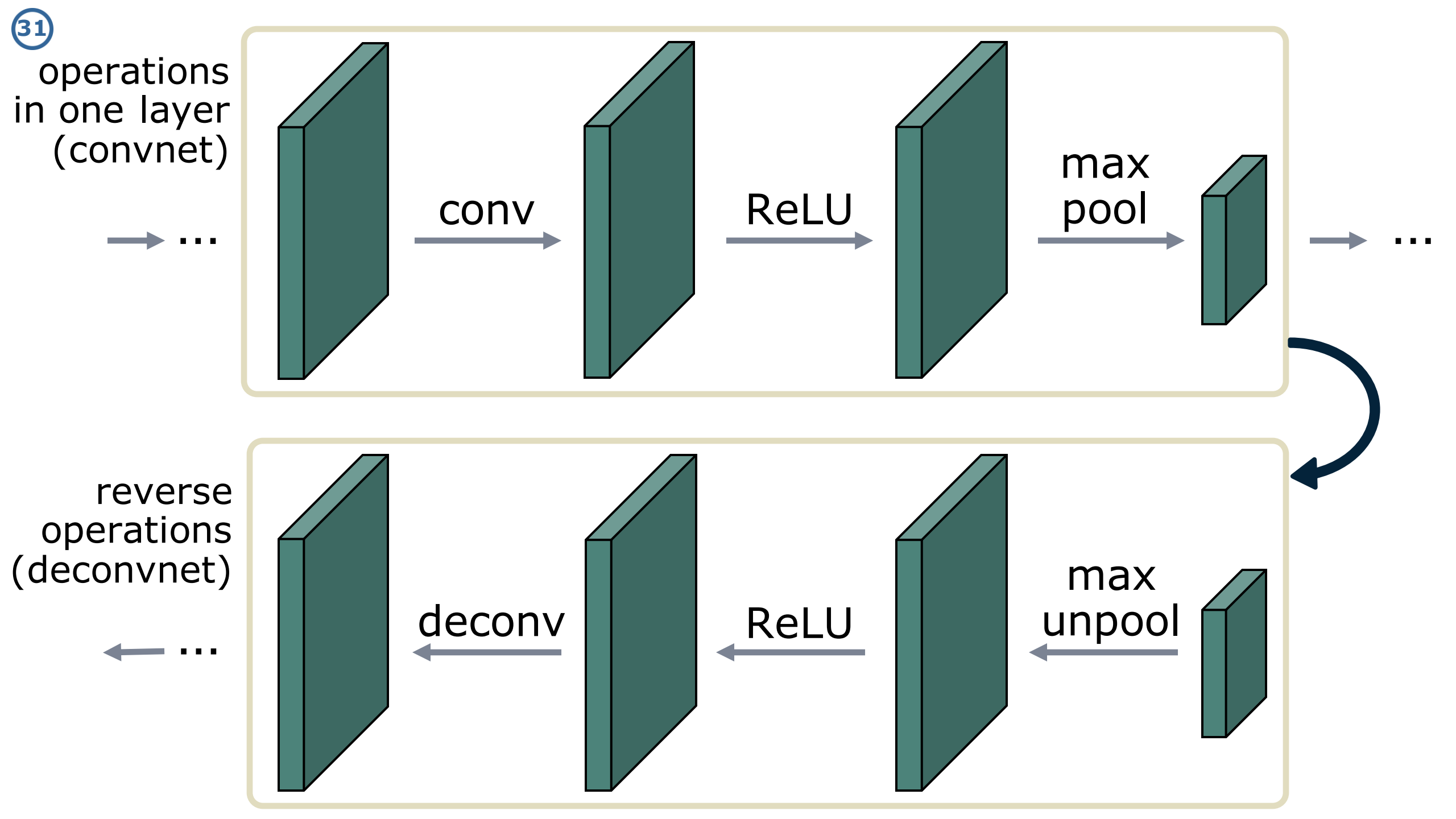
- Unclear which elements of the sample (e.g., part of the image) caused the high activation

Deconvnets – visualizing activation maps

Deconvnets

- Used to interpret standard convolutional neural networks
- Activations are **mapped back to the input pixel space** ("backpropagating" activations instead of gradients)
- Visualize input patterns that cause specific activations
- **Convolutional filters** are analyzed by finding the patterns in input space that cause the activations in the produced map to be high

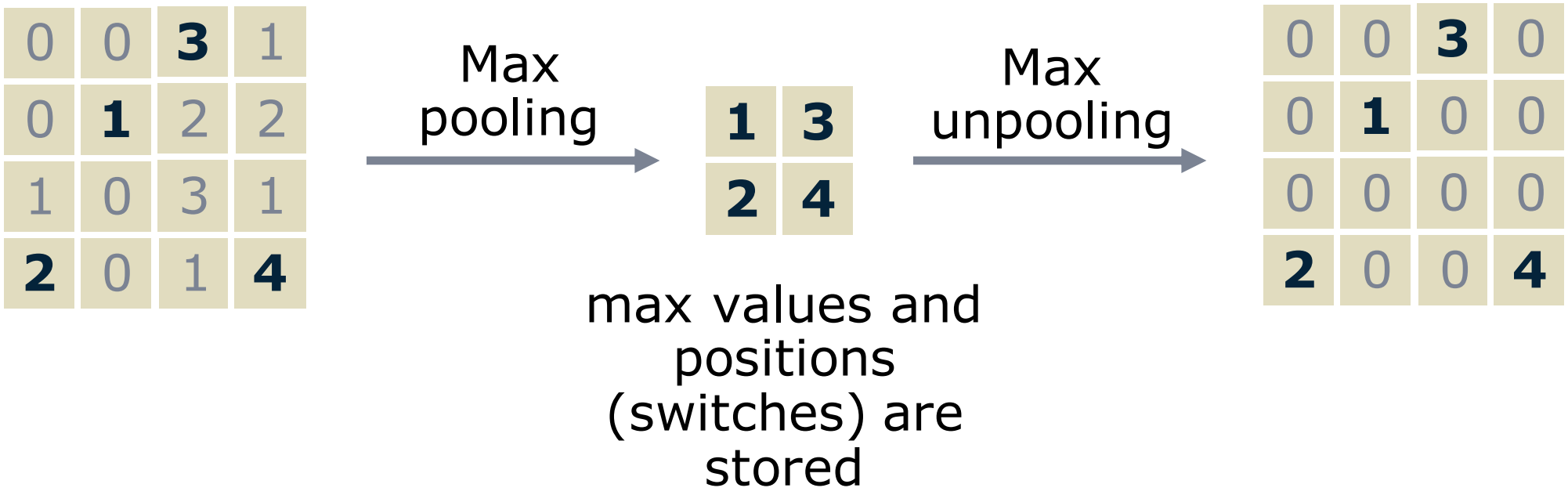
Zeiler, M. D., & Fergus, R. (2014, September). Visualizing and understanding convolutional networks. In *European Conference on Computer Vision* (pp. 818-833). Springer, Cham.


operations
in one layer
(convnet)
...
conv
ReLU
max
pool
...
reverse
operations
(deconvnet)
...
deconv
ReLU
max
unpool

Procedure

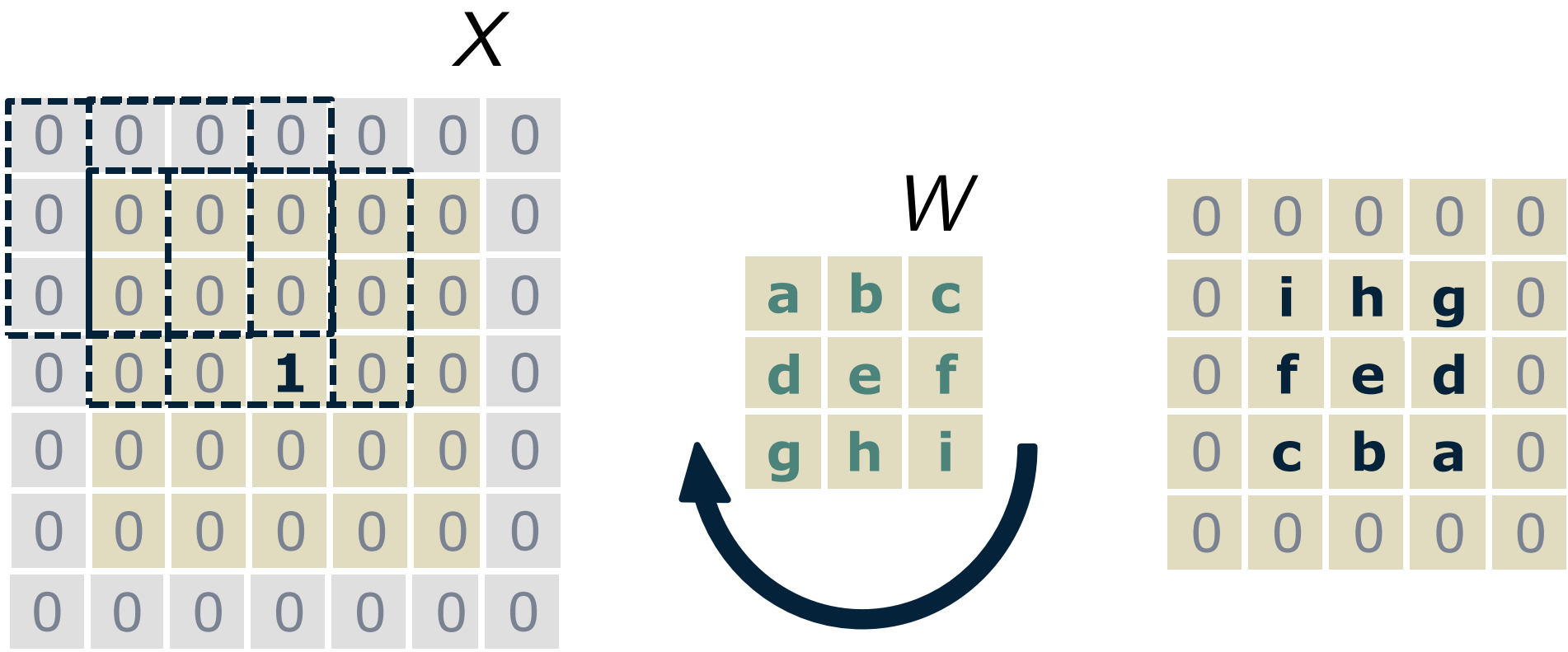
1) Forward propagate an image
2) Choose a filter of interest
3) Set all activation maps (channels) to zero except the activation map of interest
4) Attach a deconvnet to each layer and follow the path back to input space
5) Visualize the input to show the pattern the activation of interest is sensitive to

Unpooling


0 0 3 1
0 1 2 2
1 0 3 1
2 0 1 4
Max pooling
1 3
2 4
Max unpooling
0 0 3 0
0 1 0 0
0 0 0 0
2 0 0 4
max values and positions (switches) are stored

Deconvolution

Uses the learned convnet-filters, but horizontally and vertically flipped (rotated by 180°)



Deconvolution

Uses the learned convnet-filters, but horizontally and vertically flipped (rotated by 180°)

X

0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	**1**	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

W

i	**h**	**g**
f	**e**	**d**
c	**b**	**a**

0	0	0	0	0
0	**a**	**b**	**c**	0
0	**d**	**e**	**f**	0
0	**g**	**h**	**i**	0
0	0	0	0	0

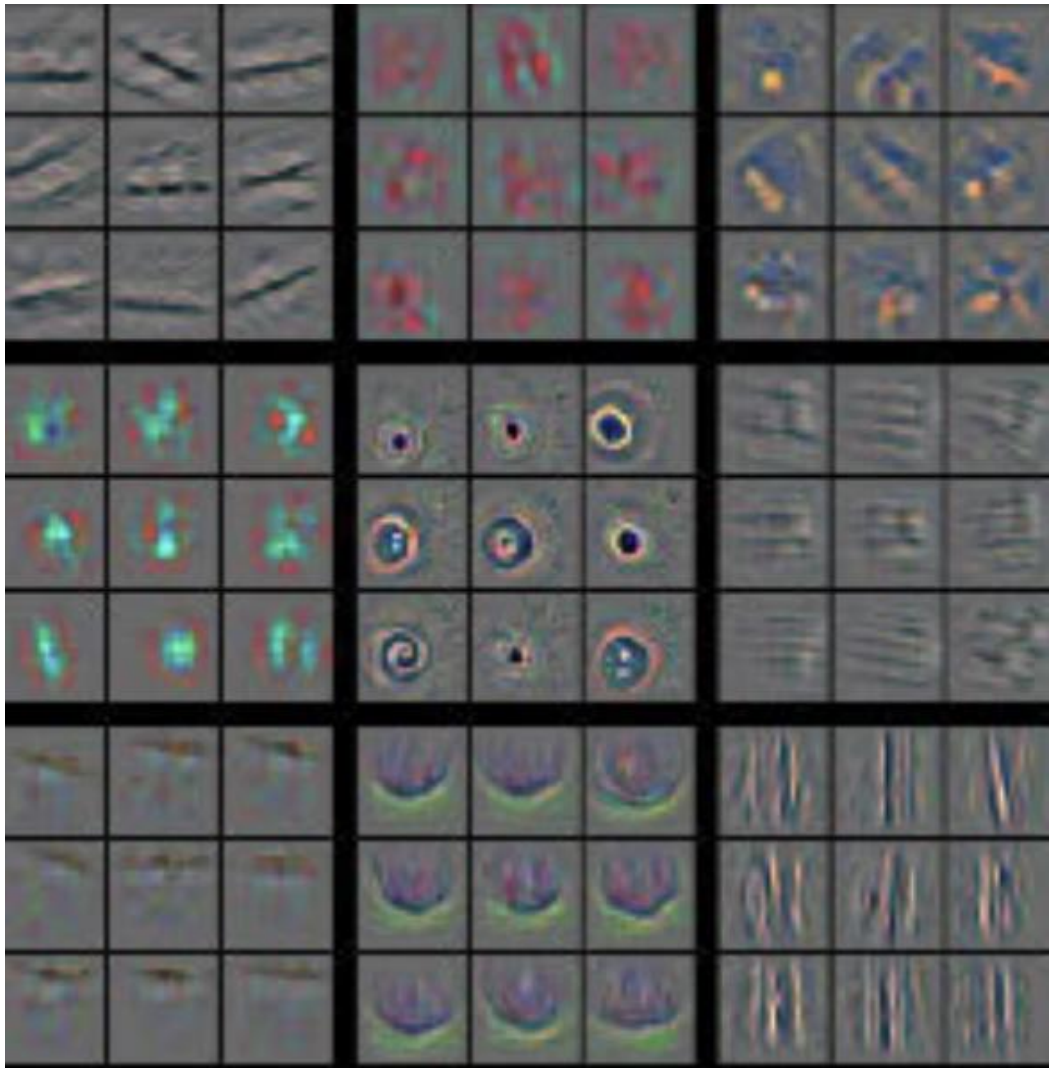
Visualization (layer 1)

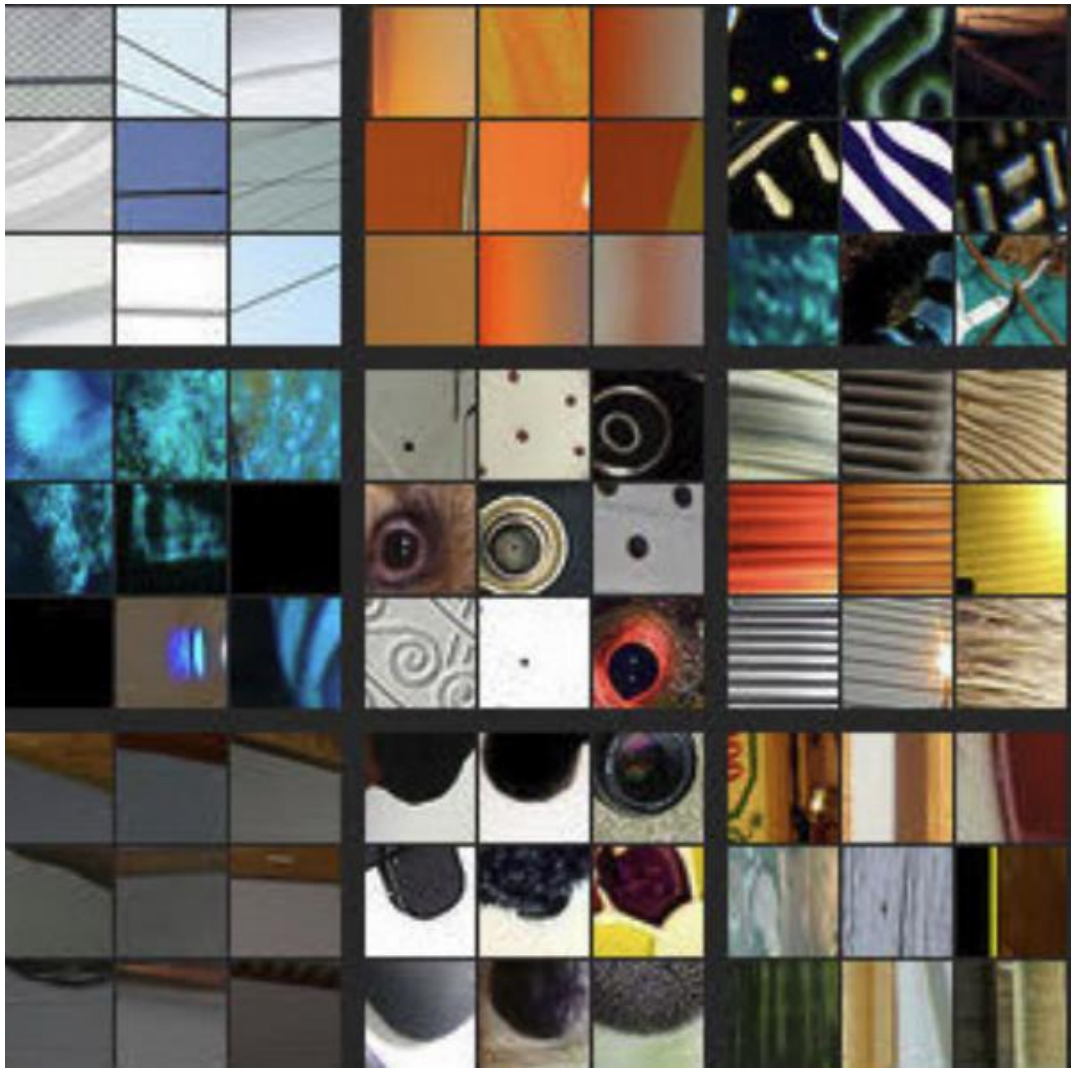
image patches that highly activate one neuron

- Image patches causing maximum activations are visualized
- Size of the image patches depend on the receptive field
- Top activations are visualized, i.e. activations which were maximally activated given an image

Zeiler, M. D., & Fergus, R. (2014, September). Visualizing and understanding convolutional networks. In *European Conference on Computer Vision* (pp. 818-833). Springer, Cham.

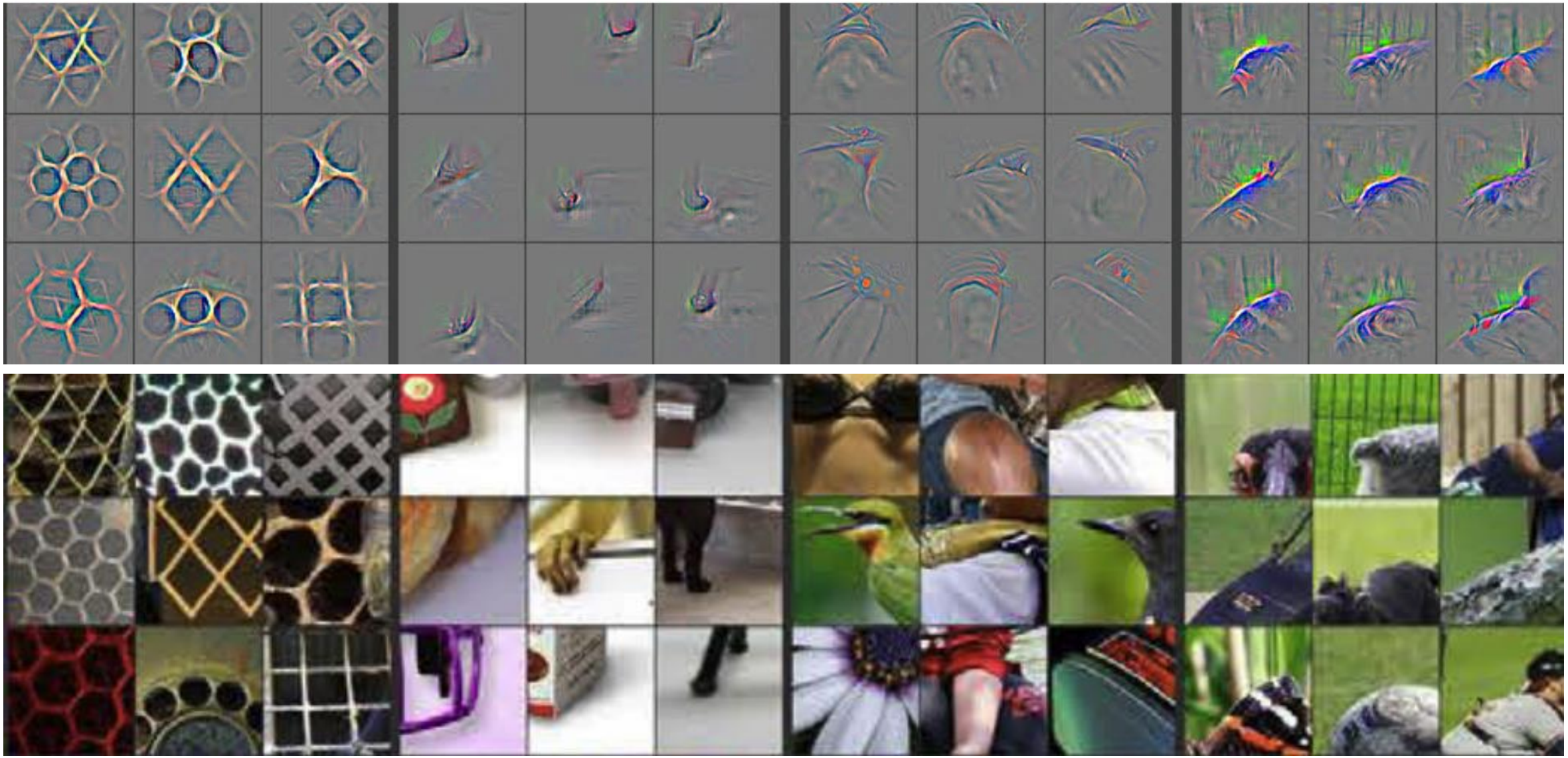
Visualization (layer 2)





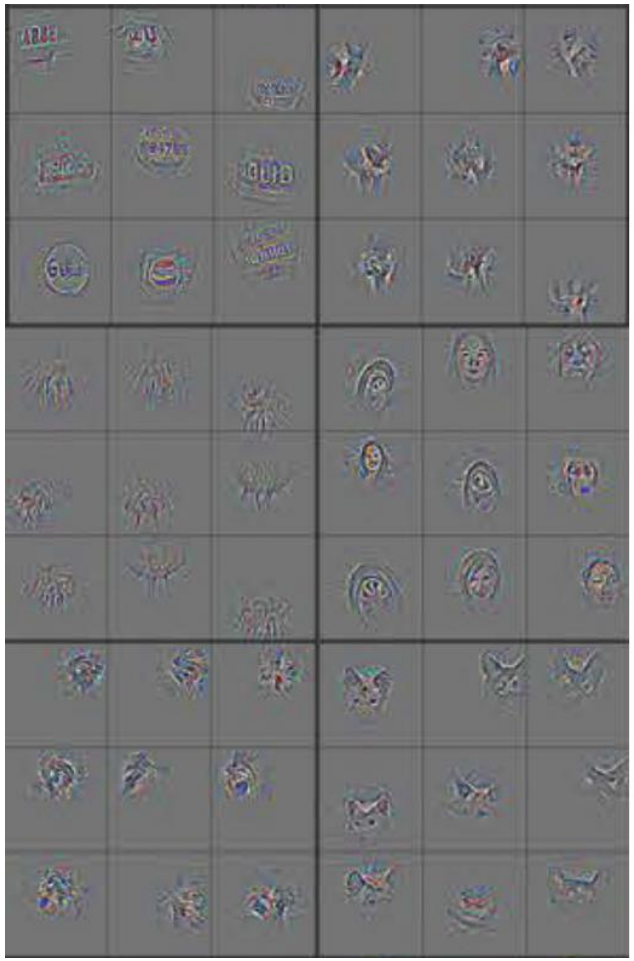
Zeiler, M. D., & Fergus, R. (2014, September). Visualizing and understanding convolutional networks. In *European Conference on Computer Vision* (pp. 818-833). Springer, Cham.

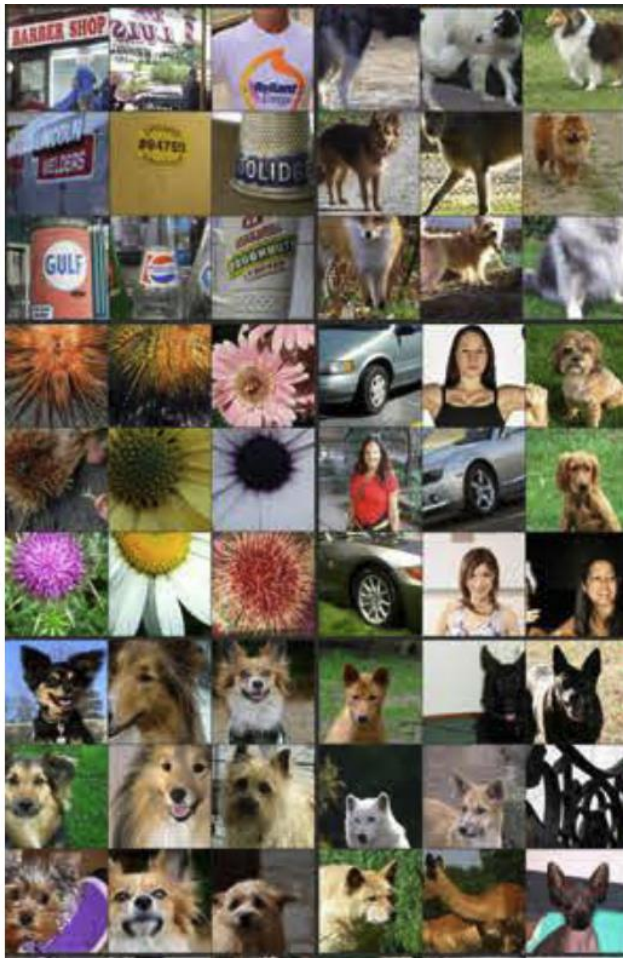
Visualization (layer 3)

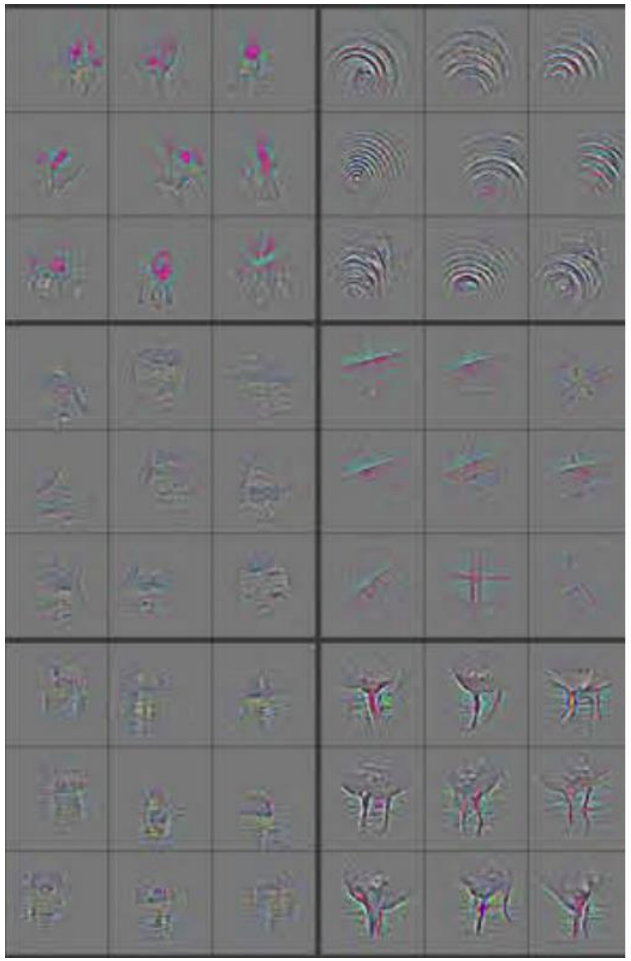


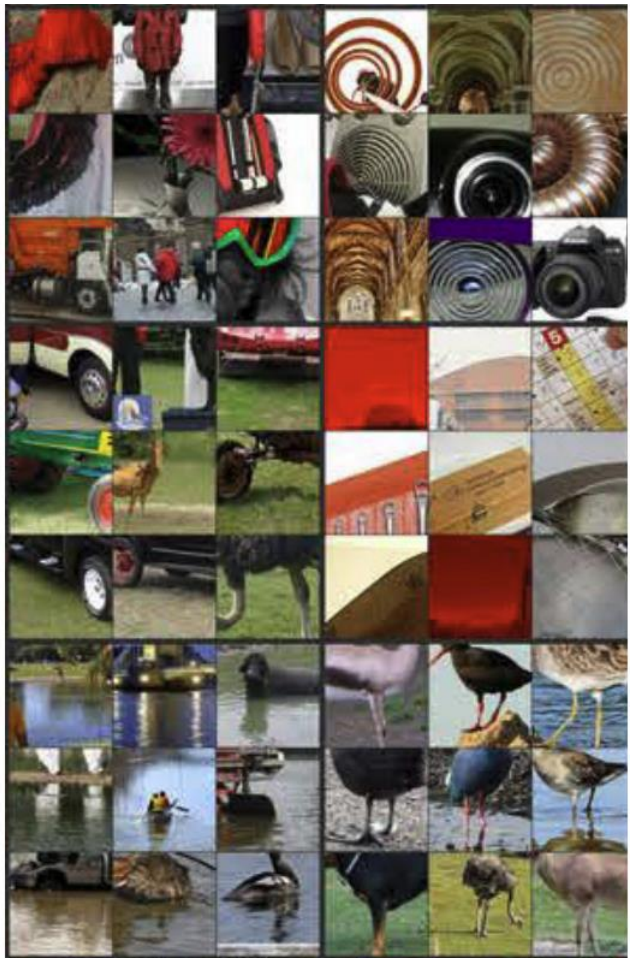
Zeiler, M. D., & Fergus, R. (2014, September). Visualizing and understanding convolutional networks. In *European Conference on Computer Vision* (pp. 818-833). Springer, Cham.

Visualization (layer 4)









Zeiler, M. D., & Fergus, R. (2014, September). Visualizing and understanding convolutional networks. In *European Conference on Computer Vision* (pp. 818-833). Springer, Cham.

Feature visualization by optimization – visualizing activation maps and layers

Feature visualization by optimization

- NN are differentiable with respect to their input
- Desired input can be determined by **optimization**

$$X^* = \arg\max_X {}^l a_i(\boldsymbol{w}, X)$$

activation function of a single neuron

- Also known as **activation maximization**

Erhan, D., Bengio, Y., Courville, A., & Vincent, P. (2009). Visualizing higher-layer features of a deep network. *University of Montreal*, *1341*(3), 1.

Olah, C., Mordvintsev, A., & Schubert, L. (2017). Feature visualization. *Distill*, *2*(11).

Nguyen, A., Yosinski, J., & Clune, J. (2019). Understanding neural networks via feature visualization: A survey. In *Explainable AI: interpreting, explaining and visualizing deep learning* (pp. 55-76). Springer, Cham.

Activation maximization

Can be extended to groups of neurons such as channels and layers

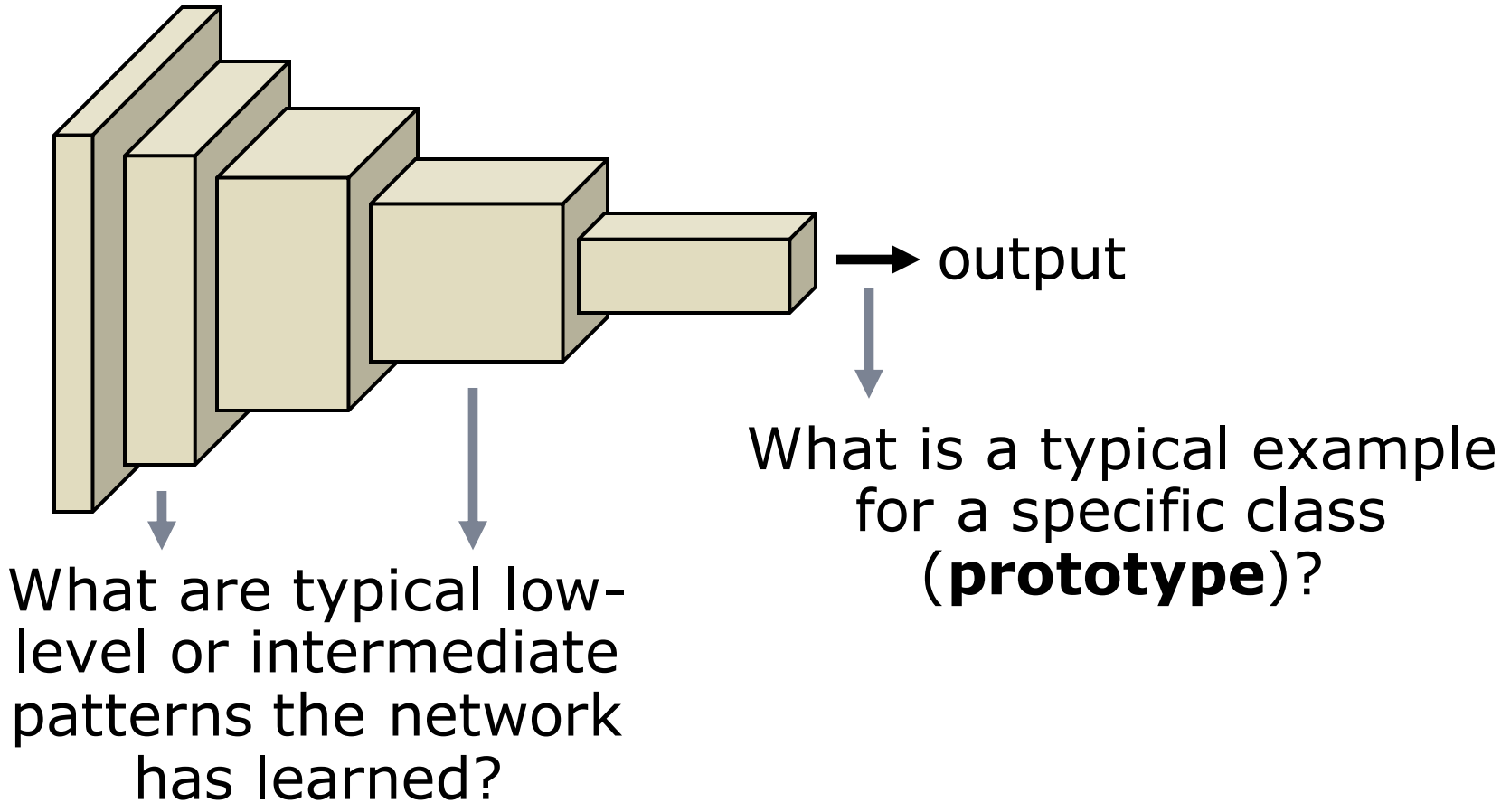
$$X^*_{\text{channel}} = \arg\max_X \sum_{r,c} {}^l a_{r,c,b}(\boldsymbol{w}, X)$$

$$X^*_{\text{layer}} = \arg\max_X \sum_{r,c,b} {}^l a_{r,c,b}(\boldsymbol{w}, X) \quad \textbf{(DeepDream)}$$

Mordvintsev, A., Olah, C., & Tyka, M. (2015). Inceptionism: Going deeper into neural networks.

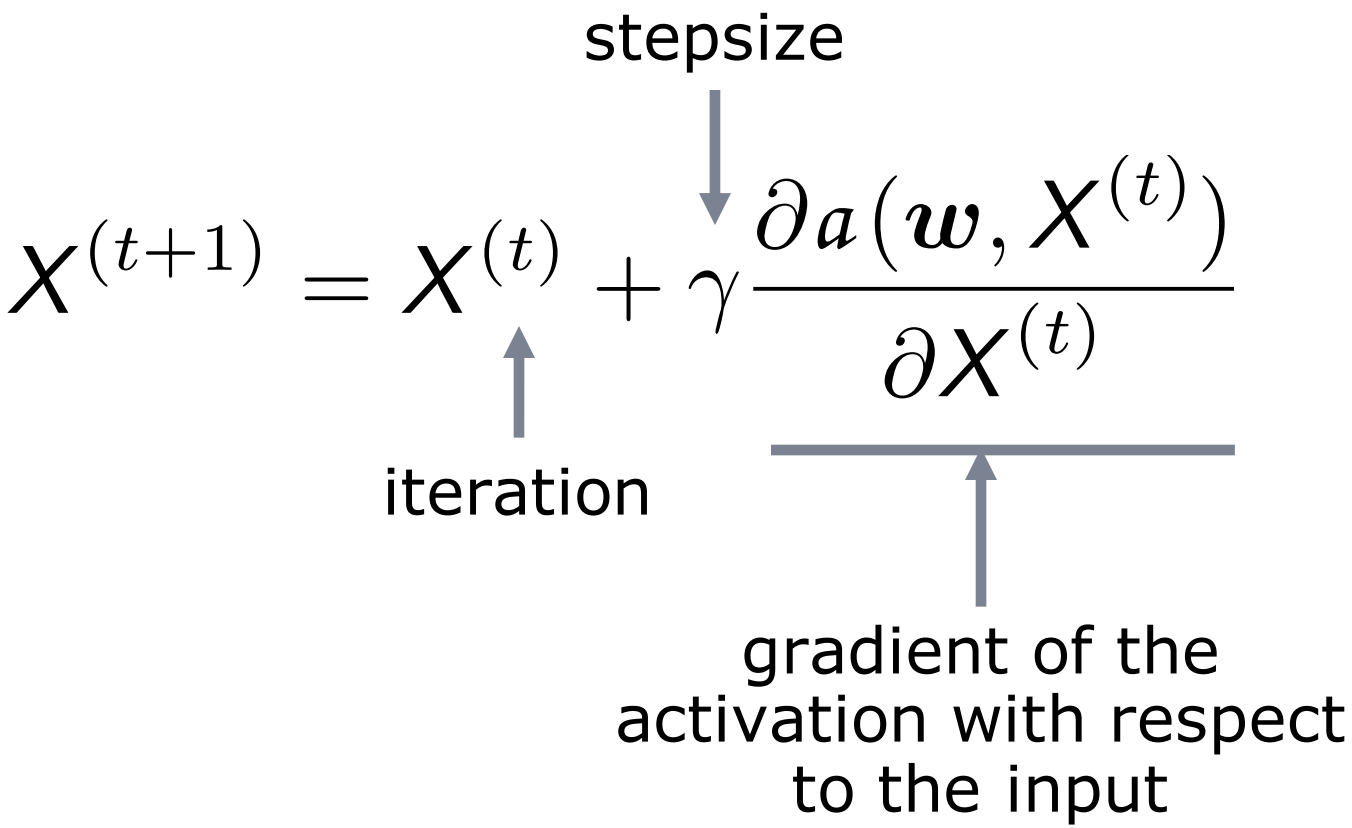
Which questions can be asked?

Analyzed activation(s) can be in each position of the network



Activation maximization

Non-convex optimization problem that can be solved, for example, by gradient ascent



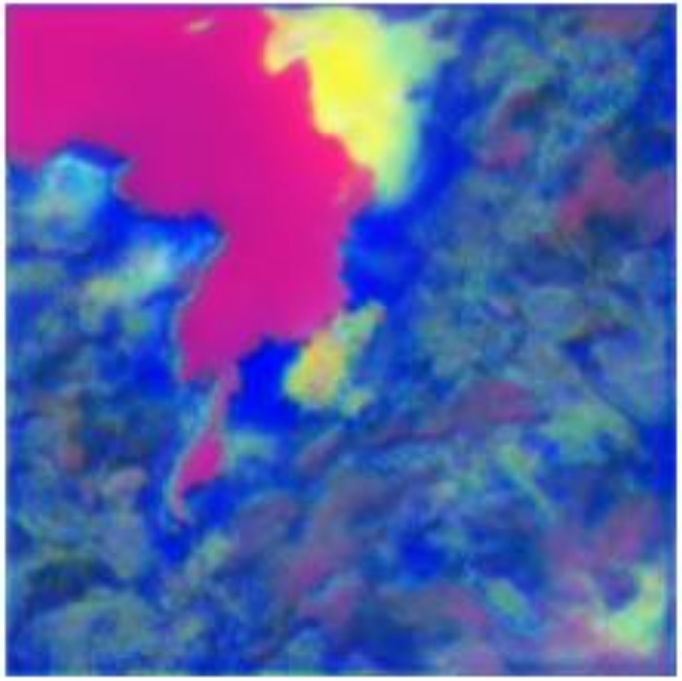
Activation maximization

$$X^{(t+1)} = X^{(t)} + \gamma \frac{\partial a(\boldsymbol{w}, X^{(t)})}{\partial X^{(t)}}$$

1. Select a neuron/an activation of interest
2. Random initialization of $X^{(0)}$
3. Compute the gradients, while the network is fixed
4. Update the input in small steps
5. Iterate 3.+4. until the gradient reaches a threshold or a fixed number of iterations

Challenge

- Leads often to unrealistic, i.e., **uninterpretable** results
- Without conditions, there are **too many solutions** which cause high activations, especially in a high-dimensional image space
- Fooling examples (patterns with high frequency noise)
- Activation maximization initialized with a real image can turn into an **adversarial example** (realistic example which cause an unexpected, and mostly undesirable, behavior of a neural network)



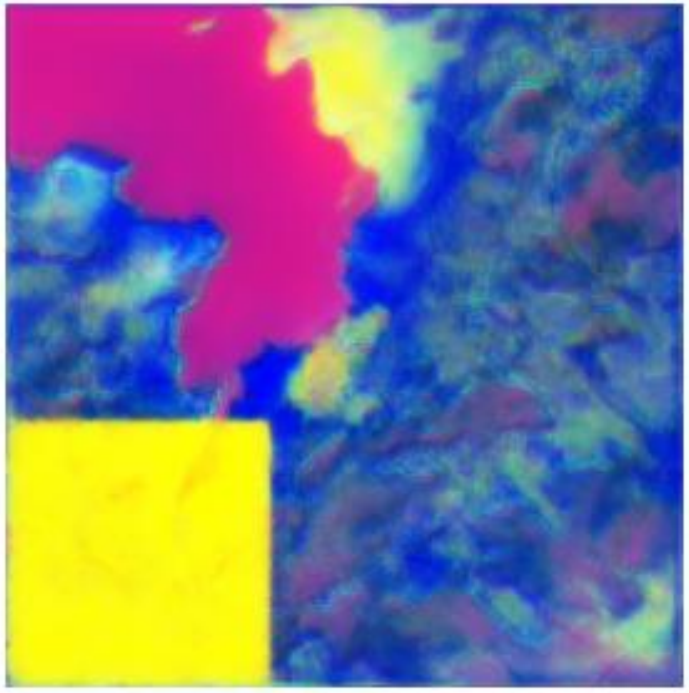
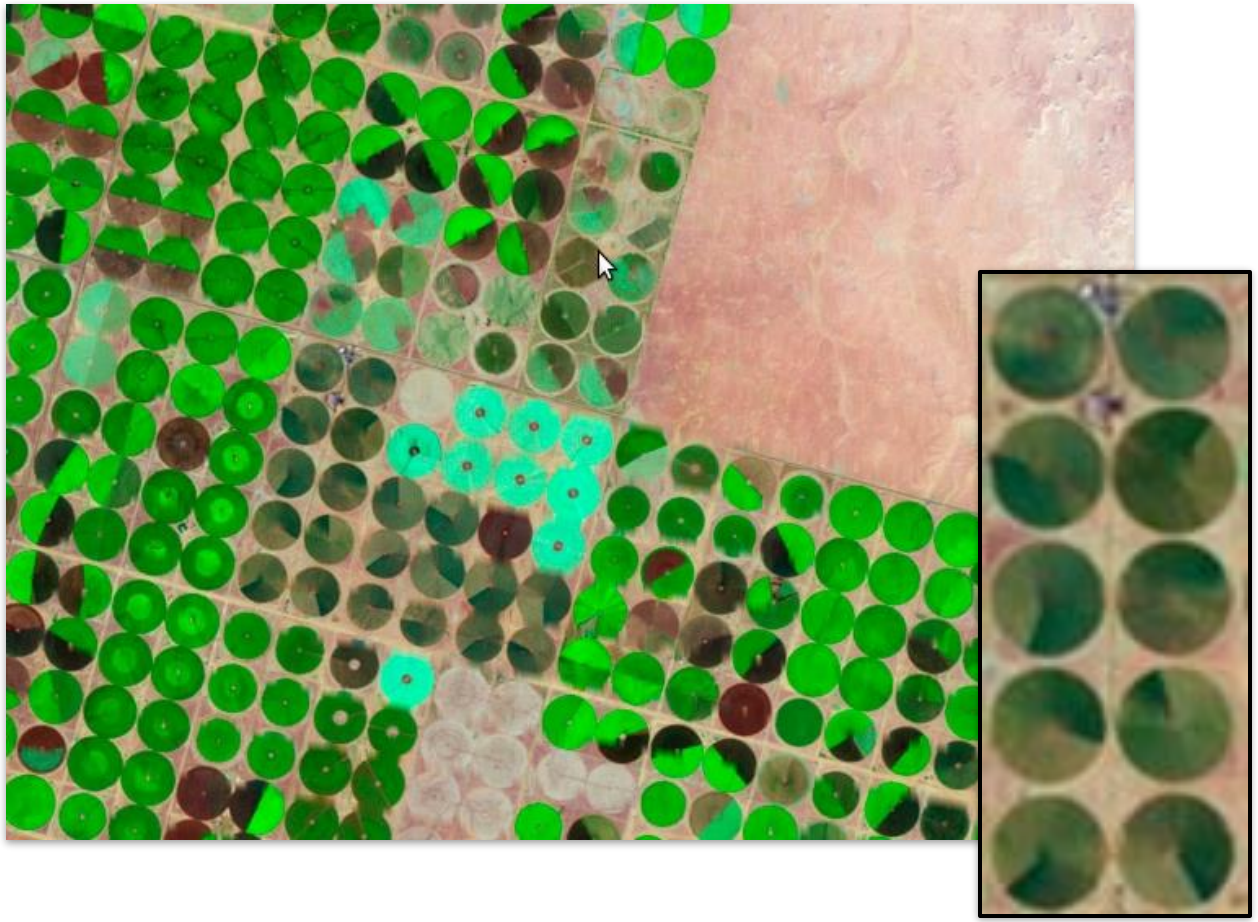
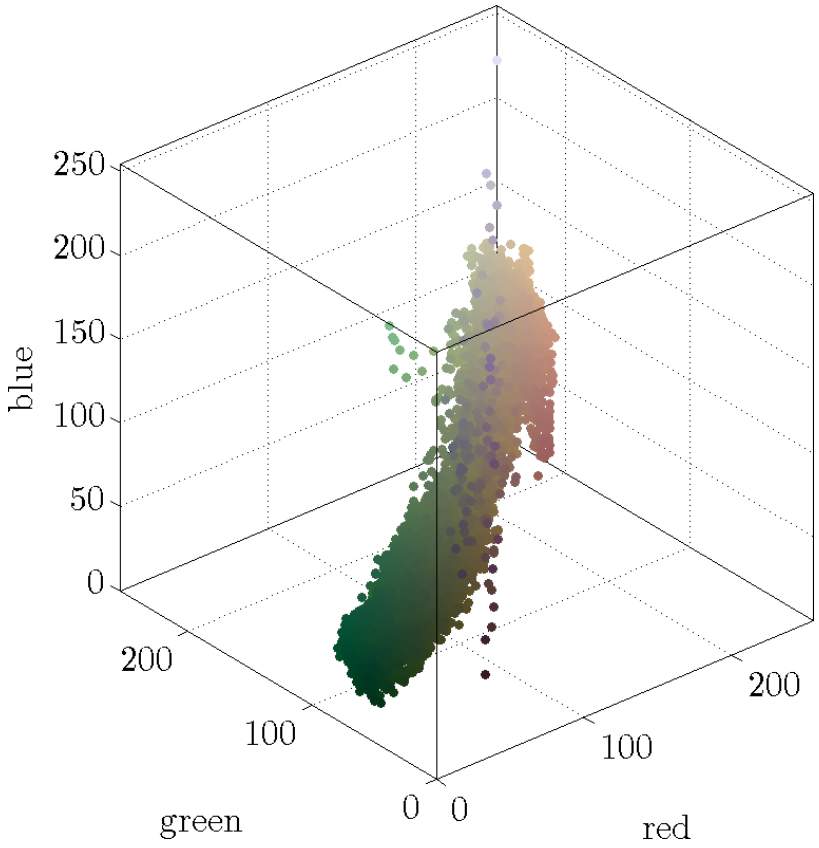


Image priors

Goal: Constrain the optimization to create an example which belongs to a distribution of **natural images**





Regularization

Activation is changed, i.e., regularized:

$$X^* = \arg\max_X \left(a(X) - r(X)\right)$$

$$X^{(t+1)} = X^{(t)} + \underbrace{\gamma \frac{\partial a(X^{(t)})}{\partial X^{(t)}}}_{\text{causes activation to be maximized}} - \underbrace{\lambda \frac{\partial r(X^{(t)})}{\partial X^{(t)}}}_{\text{causes regularization loss to be minimized}}$$

Nguyen, A., Yosinski, J., & Clune, J. (2019). Understanding neural networks via feature visualization: A survey. In *Explainable AI: interpreting, explaining and visualizing deep learning* (pp. 55-76). Springer, Cham.

Smoothness prior

- Total variation loss: Sum of the absolute differences for neighboring pixel-values
- In practice, blurred version of $X^{(t)}$ is used instead

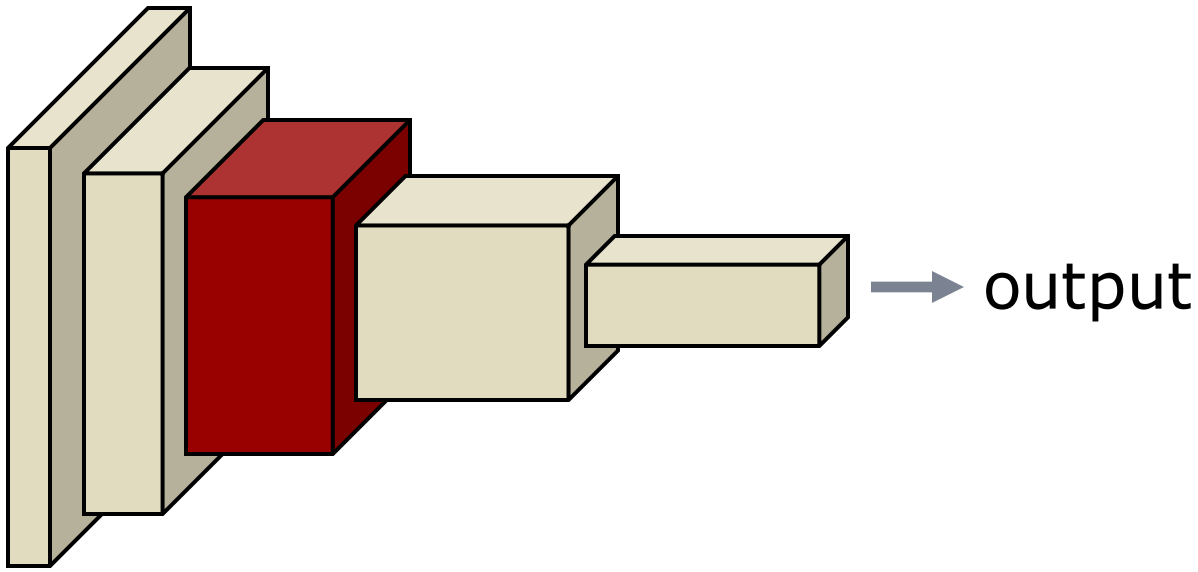
$$X^{(t+1)} = X_r^{(t)} + \gamma \frac{\partial a(X^{(t)})}{\partial X^{(t)}}$$

blurred version of $X^{(t)}$

- Also other non-differentiable regularizers can be used

DeepDream

- All activations in an entire layer are maximized instead of a single filter
- Initialization is done with an existing image instead of random noise



Mordvintsev, A., Olah, C., & Tyka, M. (2015). Inceptionism: Going deeper into neural networks.

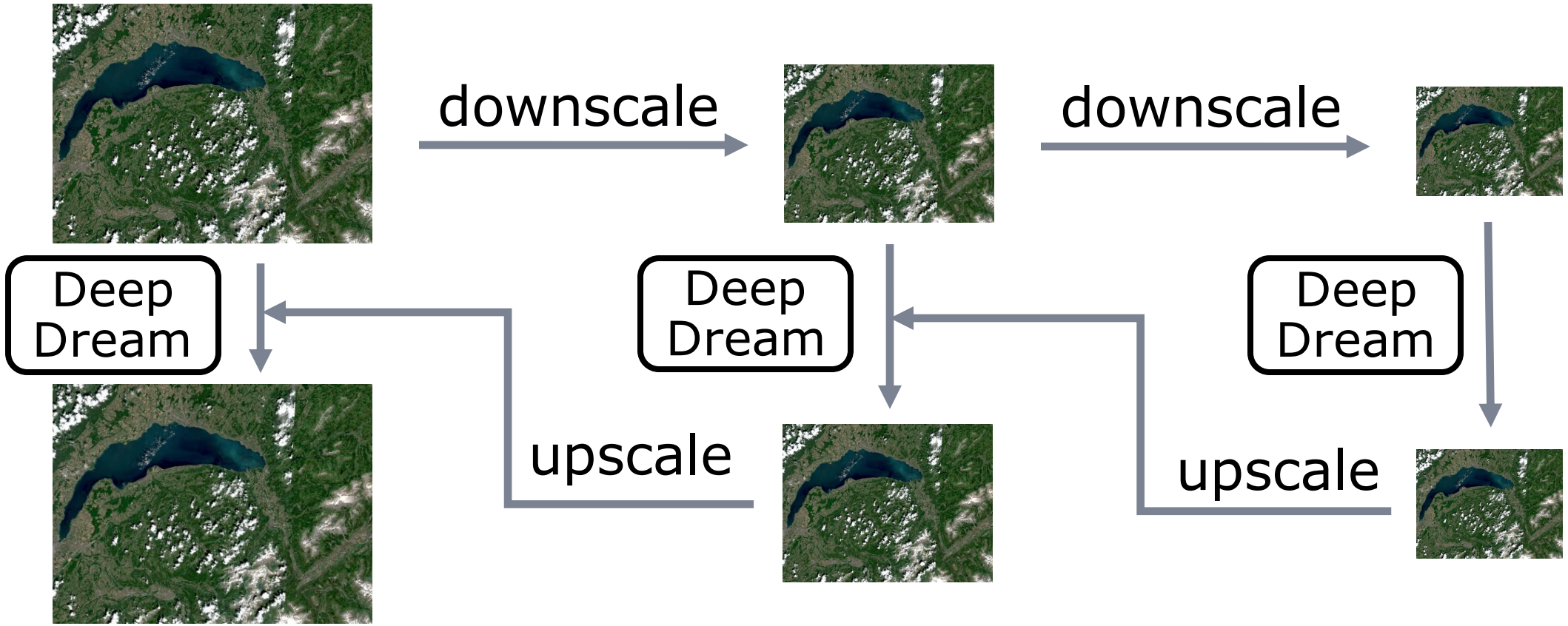
DeepDream octaves

- Applying gradient ascent at different scales
- Used to reduce noise and increase resolution and granularity of patterns

1. Choose an image and define a number of processing scales
2. Resize the image to smallest scale

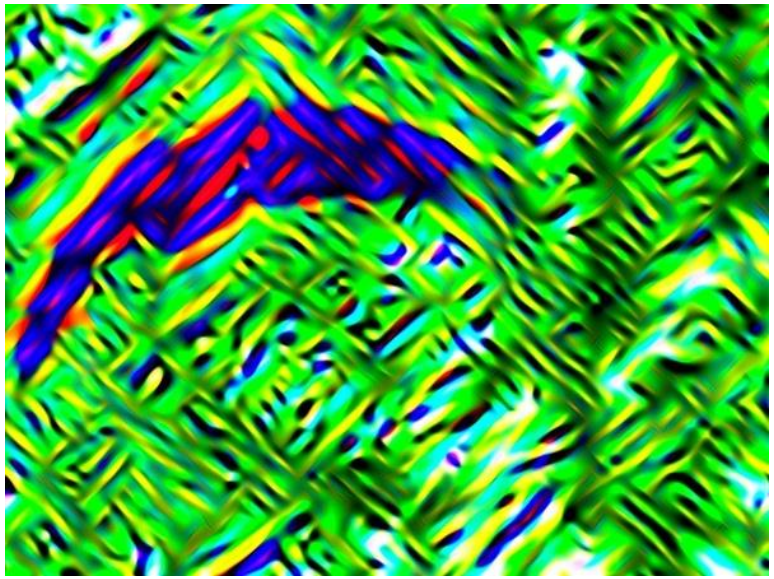
DeepDream octaves

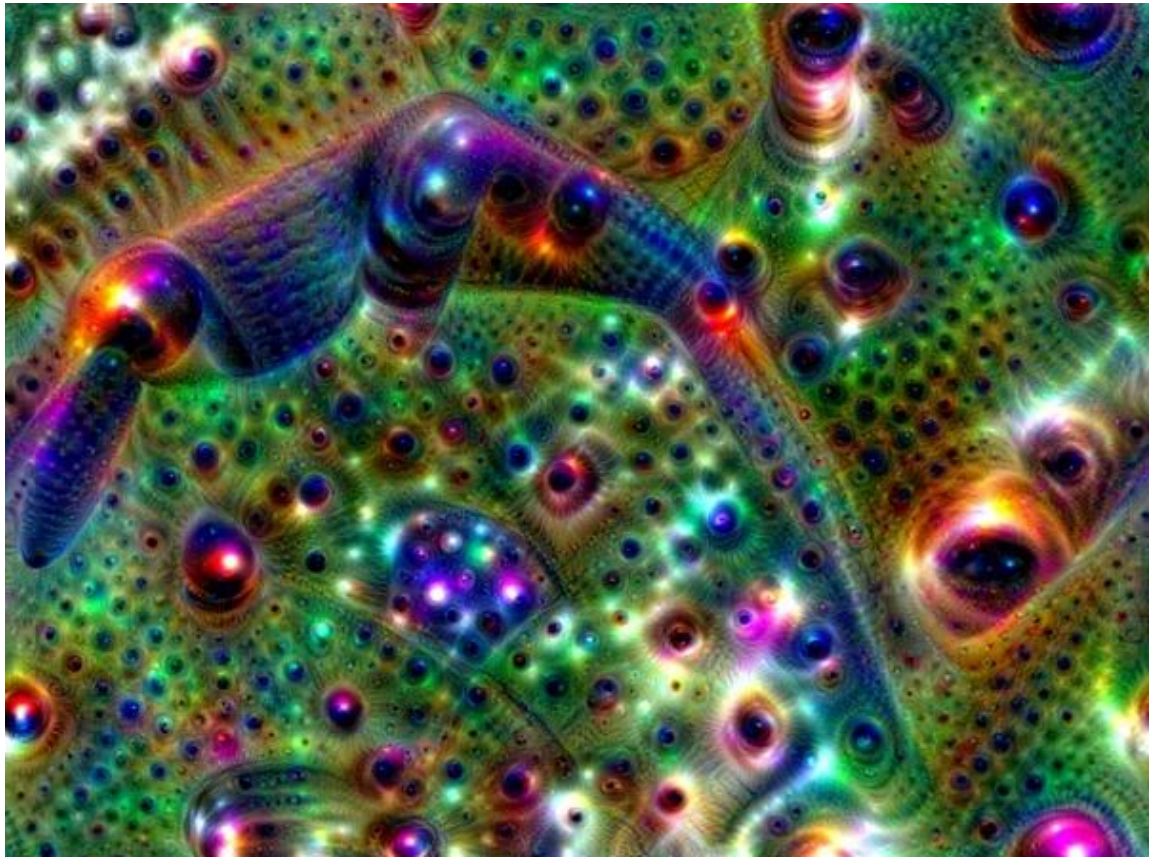
3. Perform gradient ascent and upscale image to the next scale
4. Reinject details which are lost during upscaling
5. Repeat 3. + 4. and stop after the largest scale (original size)



What is actually happening?

- Result depends on the input and what the network has learned, i.e., the output depends on the features in the training data
- Features in the input are **amplyfied** with what was learned in the network





Summary

- Different components in a neural network can be visualized, where activations are more intuitiv than weights
- Activation maximization is the basis for many visualization approaches to understand which patterns a network has learned
- (Pre-)trained networks are necessary

Further literature

- See references in the bottom of the slides
- https://distill.pub/: Many papers and source code related to feature visualization