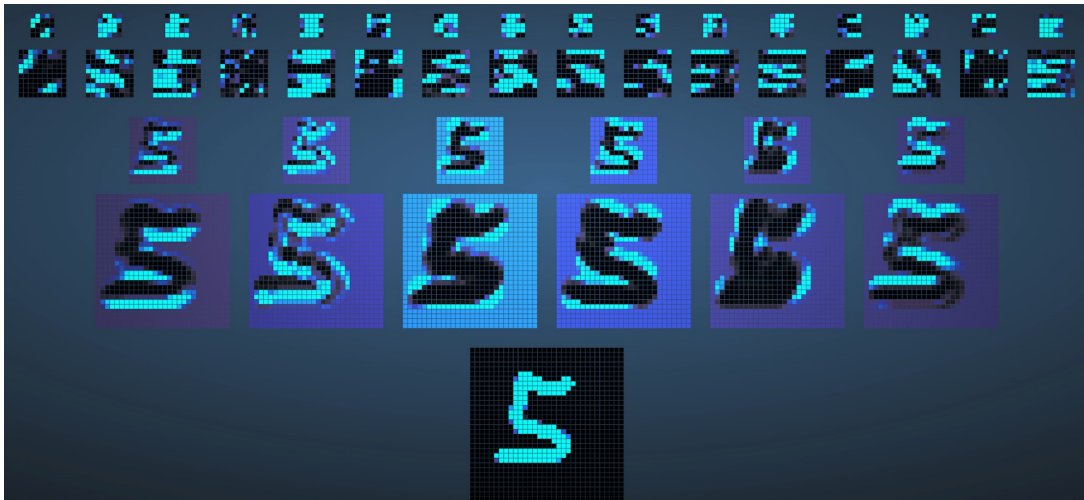


Pooling

Herr Professor Dr. Michael Wollowski

1

Processing of the Convolutional and Pooling Layers of LeNet 5



2



3

Pooling – What do we know?

- Pooling layers shrink each feature map by summarizing small neighborhoods into a single number.
- The network keeps the information "this feature showed up around here" and throws away the exact pixel position.

4

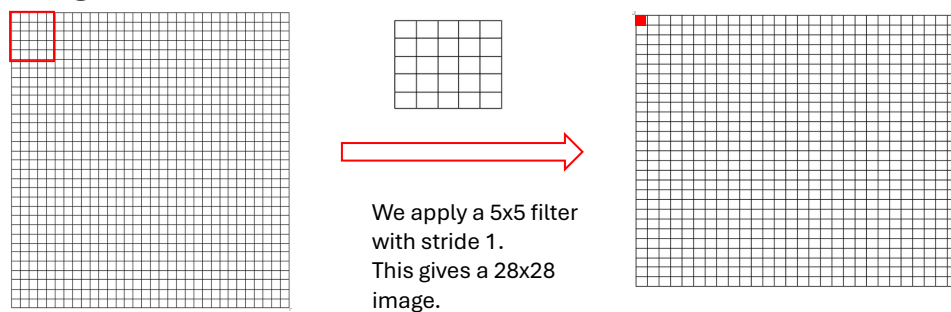
Pooling – What do we know?

- **It cuts the workload.**
 - Due to decreasing the size by a factor of four, later layers do much less computation and use less memory.
- **It lets deeper layers see bigger patterns.**
 - Applying a 2x2 reduces by a factor of four.
 - A 5x5 filter in the next layer covers a much larger area.
 - Additionally, that filter looks at the "summarized" version of the prior convolutional layer.

5

The Big Picture

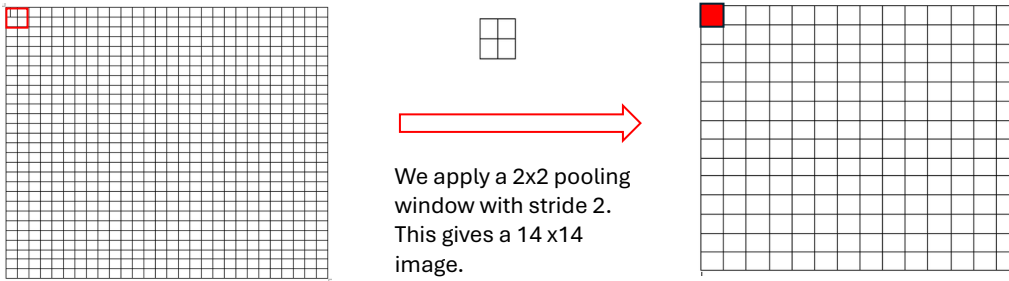
- Let's look at the processing of LeNet 5 through the layers.
- We'll start with a 32x32 input image. It is a 28x28 image with 2 cell padding.



6

The Big Picture

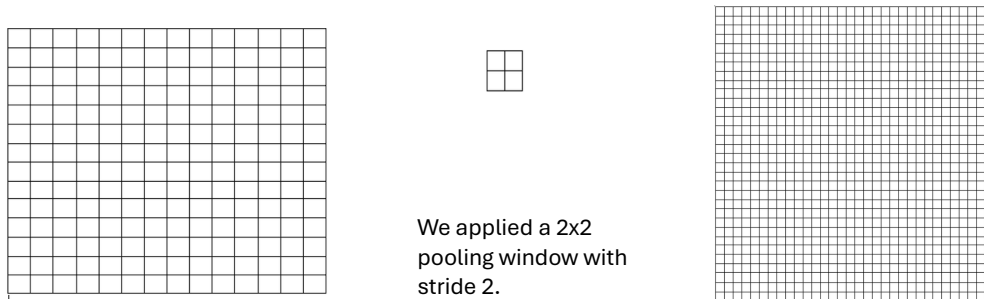
- A 5x5 filter has a receptive field of, well, 5x5.
- Pretty myopic.
- Next: pooling with a 2x2 window



7

The Big Picture

- But, what is the receptive field of each of the new cells?

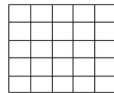
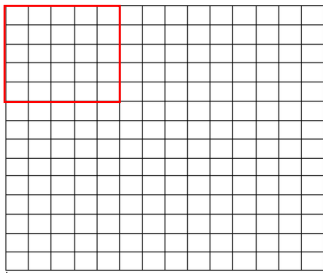


- 6 x 6

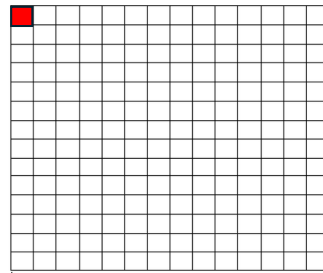
8

The Big Picture

- Now for the fun part.
- We apply another 5 x 5 filter, resulting in a 10 x 10 grid.



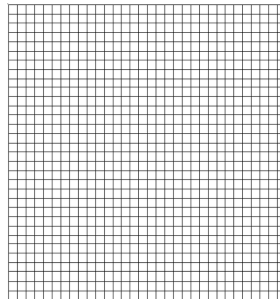
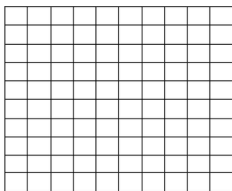
We apply a 5x5 filter
with stride 1.



9

The Big Picture

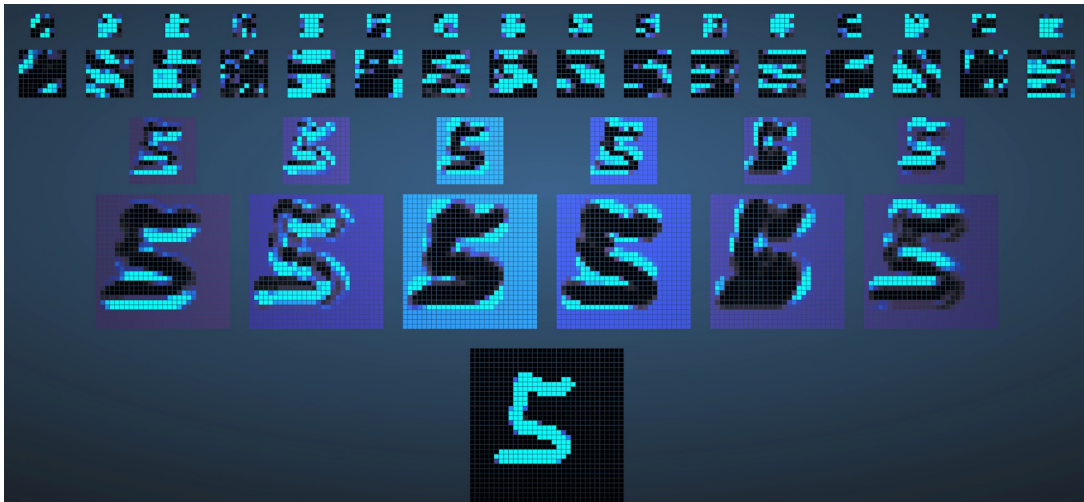
- But, what is the receptive field of each of the new cells?



- 14 x 14
- What is the overlap?

10

Great – now what?



11

Visualizing the Processing of CNNs

- The prior image shows the activations.
- However, the activations are not the right thing to study.
- Instead, we need to investigate what portions of the network are looking for.

12

Pooling - Analogy

- Suppose I described a digit as "a flat bar across the top, a short stroke down the left, and a curve bulging right at the bottom."
- You would know it is a 5.
- The sentence does not look like a 5.
- The top row of the MNIST CNN visualization is that kind of "description."
- It is written in a vocabulary the network invented for itself.
- We cannot read it, but the classifier can.

Analogy courtesy of Claude.ai; Edited for clarity.

13

Pooling - Caveat

- LeNet5 has no convolutional layer for whole digits.
- The last step happens in the fully connected classifier, which learns which combinations of features in which regions mean "5."
- The edges -> parts -> objects story shows up most clearly in deep networks trained on natural photos.
- Many individual units respond to messy mixtures of patterns.
- ImageNet-trained CNNs turn out to rely more on texture than on shape.
- However, later layers respond to larger, more complex combinations of earlier features.

14

Human Vision and CNNs Visualization

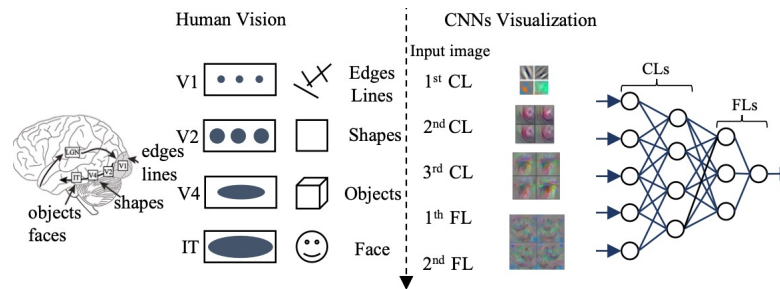
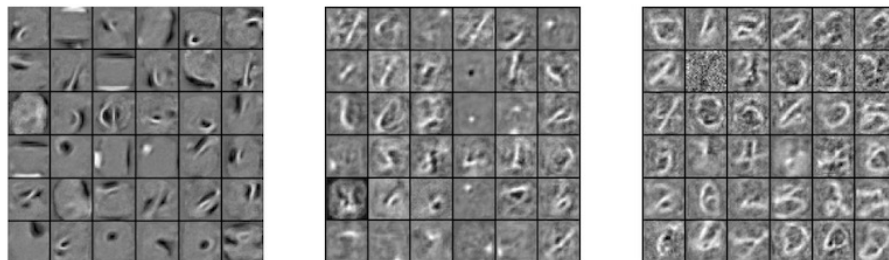


Image source: Qin et al. How convolutional neural networks see the world. doi:10.3934/mfc.2018008

15

Activation maximization

- Look for input patterns which maximize the activation of a given hidden unit.
- Below, it is applied on MNIST.
- Visualization of 36 units from the first, second, and third hidden layers.



Source: Visualizing higher-layer features of a deep network. Erhan et al, University of Montreal, Vol 1341, pp. 3. 2009.

16

Feature Visualization

- It investigates what portions of a network are looking for by generating examples.

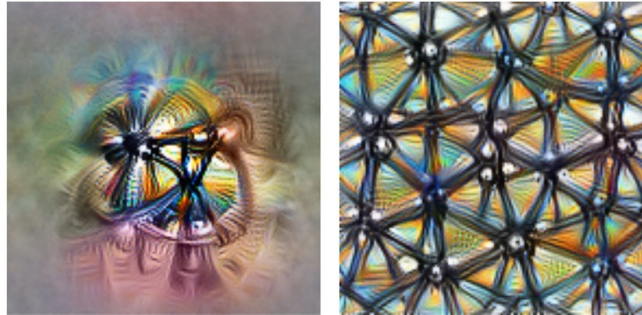


Image source: <https://distill.pub/2017/feature-visualization/>

17

Feature Visualization by Optimization

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

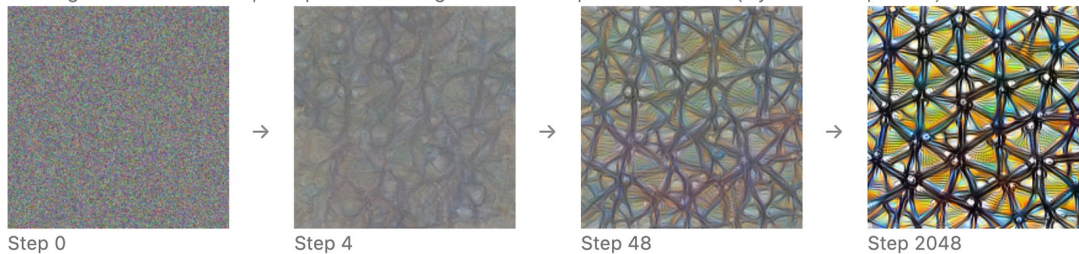


Image source: <https://distill.pub/2017/feature-visualization/>

18

Feature Visualization by Optimization

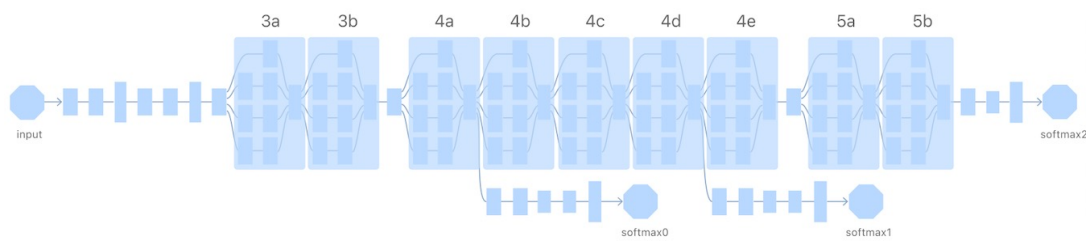
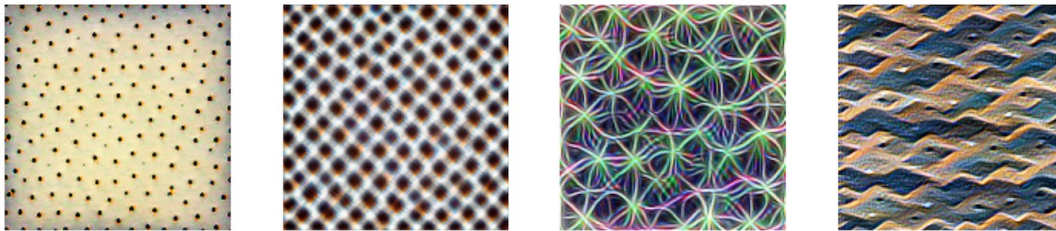


Image source: <https://distill.pub/2017/feature-visualization/appendix/>

19

Feature Visualization by Optimization

- Layer 3A:



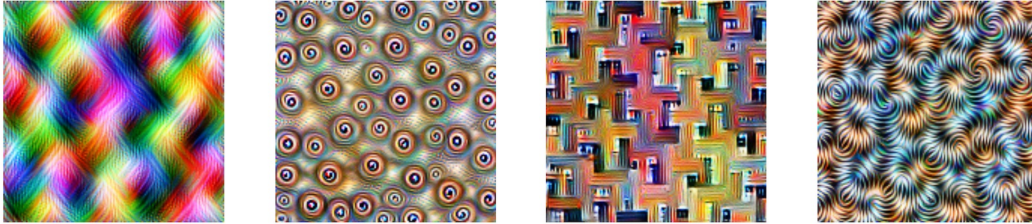
This first inception layer already shows some quite interesting textures. Each neuron only looks at a small receptive field, so these channel visualizations show you a tiling of them.

Image source: <https://distill.pub/2017/feature-visualization/appendix/>

20

Feature Visualization by Optimization

- Layer 3B:



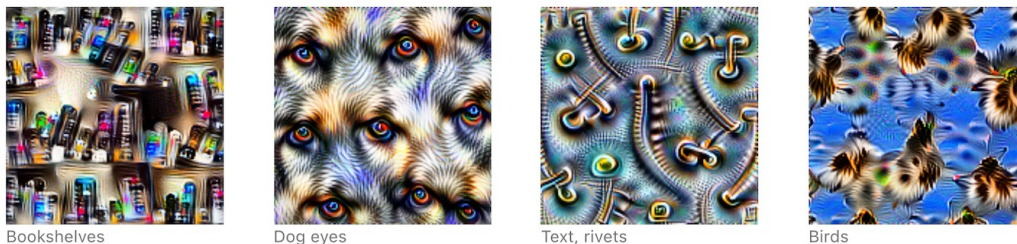
Textures become more complex, but are still very local.

Image source: <https://distill.pub/2017/feature-visualization/appendix/>

21

Feature Visualization by Optimization

- Layer 4A:



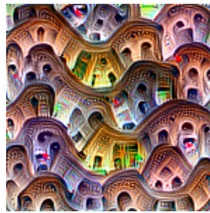
In this layer, which follows a pooling step, we see a significant increase in complexity. We begin to see more complex patterns, and even parts of objects.

Image source: <https://distill.pub/2017/feature-visualization/appendix/>

22

Feature Visualization by Optimization

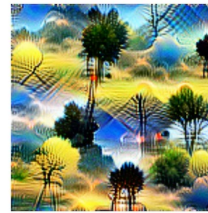
- Layer 4B:



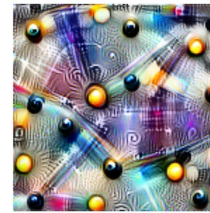
Architecture



Fluffy rope



Trees



Billiard balls

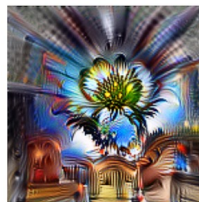
Already you can begin to make out parts of objects — such as the billiard ball detector on the right here. Visualizations start having more context, like the neuron on the second from the right which responds to trees in front of sky and ground.

Image source: <https://distill.pub/2017/feature-visualization/appendix/>

23

Feature Visualization by Optimization

- Layer 4C:



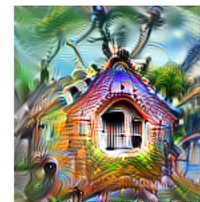
Palm trees



Wheels



Dogs on leash



Houses

In this layer things get complex enough that it can often help to look at the neuron objective rather than the channel objective. You can find neurons responding to dogs on leashes only, many wheel detectors, and a lot of other fun neurons.

Image source: <https://distill.pub/2017/feature-visualization/appendix/>

24

Feature Visualization by Optimization

- Layer 4D:



Dog snouts



Primates



Snake heads



Restaurant dishes

By this layer we find more sophisticated concepts, like a particular kind of animal snout. On the other hand, we also start to see neurons that react to multiple unrelated concepts. It helps to look at the diversity and dataset examples to double check what a neuron reacts to.

Image source: <https://distill.pub/2017/feature-visualization/appendix/>

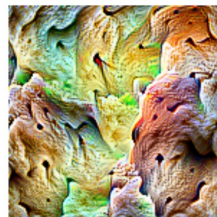
25

Feature Visualization by Optimization

- Layer 4E:



Turtle shells



Icecream & bread



Cat fur



Sombreros

At this level, many neurons will differentiate between specific animal species or react to multiple concepts. These are, however, usually still visually similar, which can lead to funny situations like reacting to both satellite dishes and sombreros. One can also still find texture detectors, though they usually react to more complex textures such as icecream, bread and cauliflower. The first neuron example here predictably reacts to turtle shells, but interestingly also to fretted instruments.

Image source: <https://distill.pub/2017/feature-visualization/appendix/>

26

Feature Visualization by Optimization

- Layer 5A:



Candles



Balls



Brass instruments



Traffic lights

Visualizations become harder to interpret here, but the semantic concepts they target are often still quite specific.

Image source: <https://distill.pub/2017/feature-visualization/appendix/>

27

Feature Visualization by Optimization

- Layer 5B:



In this layer visualizations become mostly nonsensical collages. You may still identify specific subjects, but will usually need a combination of diversity and dataset examples to do so. Neurons do not seem to correspond to particularly meaningful semantic ideas anymore.

Image source: <https://distill.pub/2017/feature-visualization/appendix/>

28