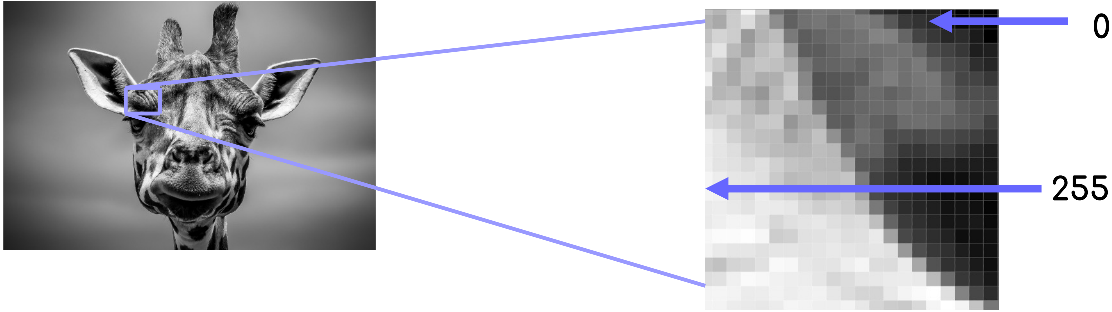
Convolutional Neural Networks

Machine Learning Techniques
(for High Energy Physics)

Adriano Di Florio
24 May 2023

Greyscale Image

- Grayscale image is a matrix of pixels [H x W]
- Pixels = picture elements
- Each pixel stores number [0,255] for brightness



RGB Image

- RGB image is a 3d array [HxWx3] or [3xHxW]
- Each pixel stores Red, Green & Blue color values [0,255]

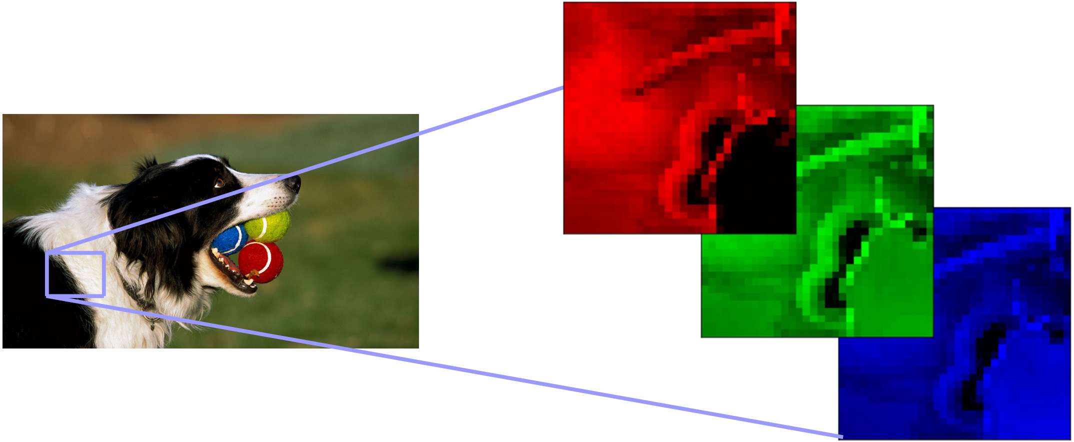
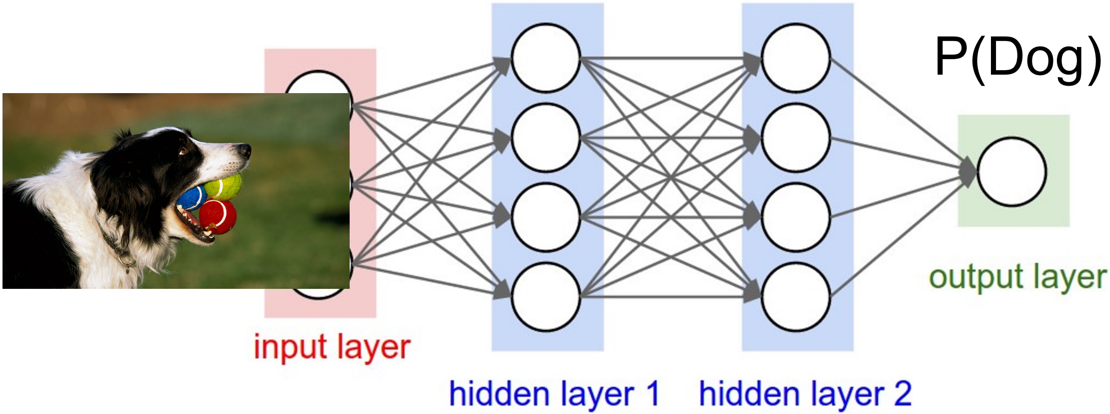


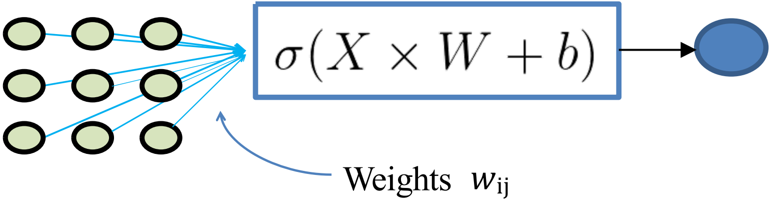
Image Recognition

→ “Dog”

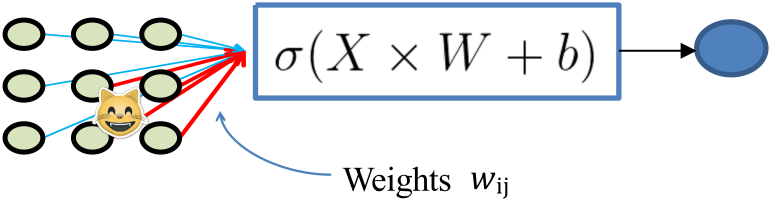
NN Approach



Problem with Images

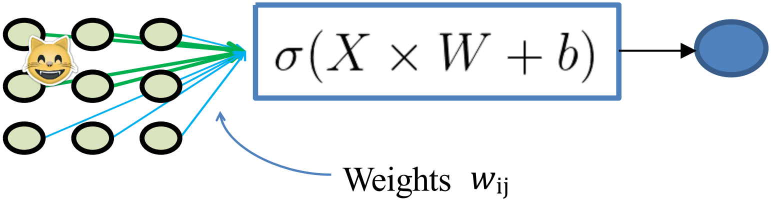


Problem with Images



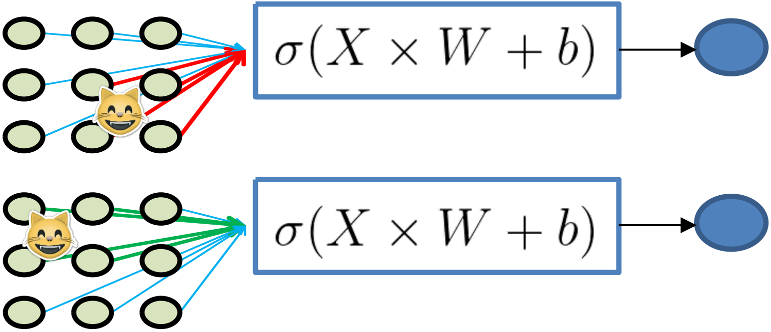
On this object, you will train
red
weights to react on cat face

Problem with Images



On this object, you will train
green
weights to react on cat face

Problem with Images



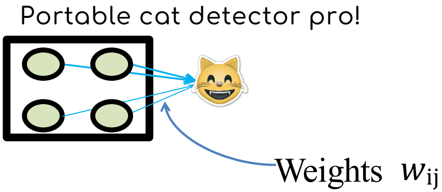
You network will have to learn those two cases separately!

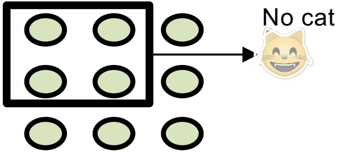
Worst case: one neuron per position.

Solution?

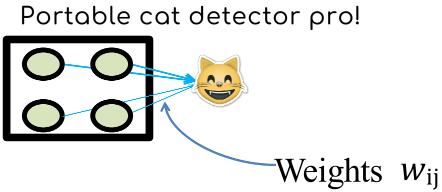
Idea: force all these “cat face” features to use **exactly the same weights**, shifting weight matrix each time.

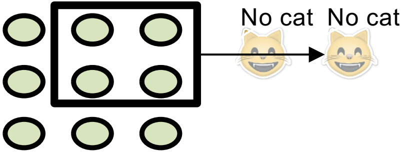
Same feature for each spot



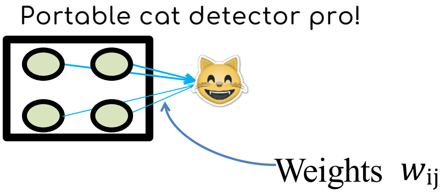


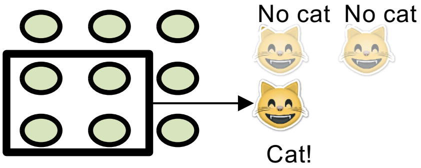
Same feature for each spot



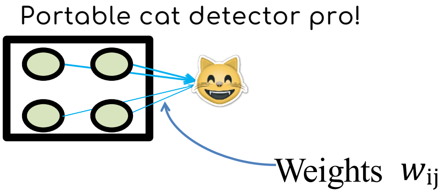


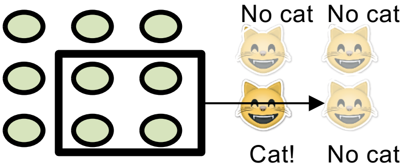
Same feature for each spot



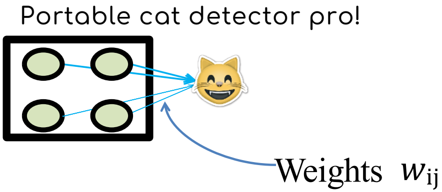


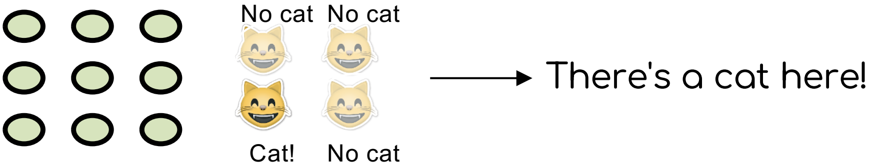
Same feature for each spot





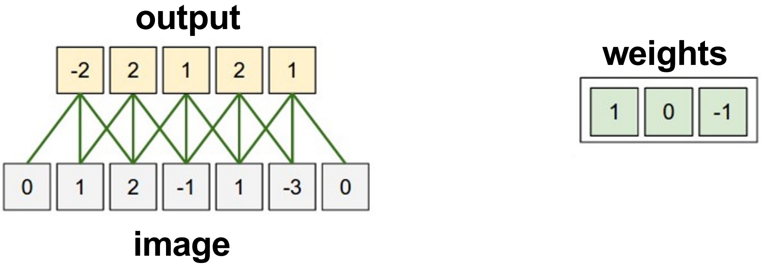
Same feature for each spot



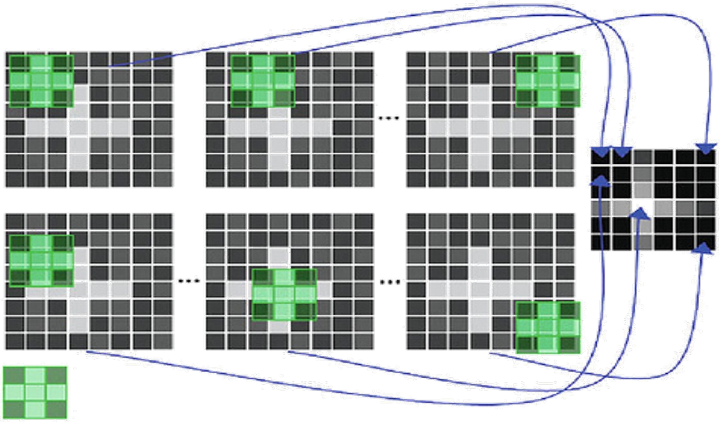


Convolution

➢ Apply same weights to all patches

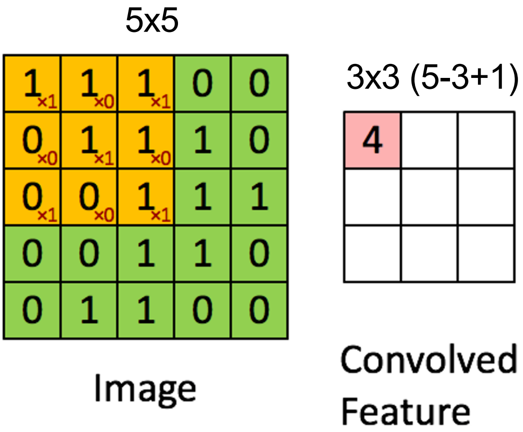


Convolution



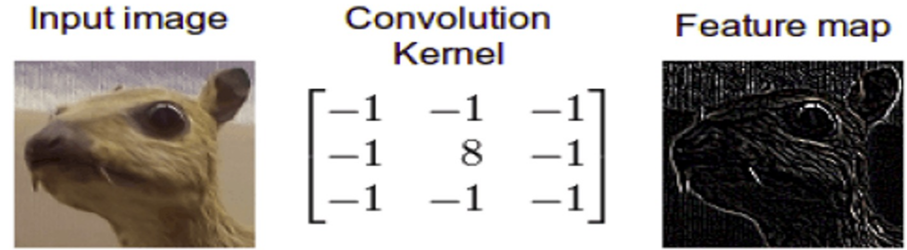
apply one “filter” to all patches

Convolution



Intuitively: how cat-like is this square?

Convolution

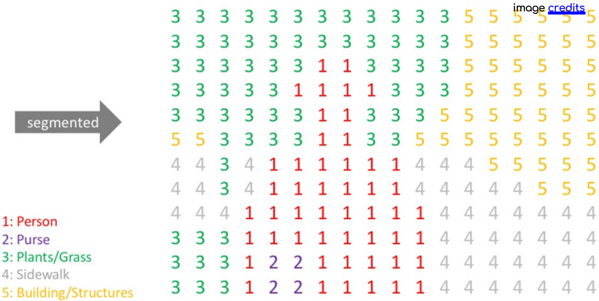


Intuitively: how edge-like is this square?

Semantic Segmentation

Semantic segmentation: the goal is to take either a multichannel image (e.g. an RGB color image with 3 channels) in and output a segmentation map (a matrix) where each pixel contains a class label represented as an integer.

Input



Semantic Labels

More specifically, the goal of semantic image segmentation is to label *each pixel* of an image with a corresponding *class* of what is being represented.

In our case : a 128x128x**13** image with [0,1] labels (changed-unchanged).

Semantic Segmentation

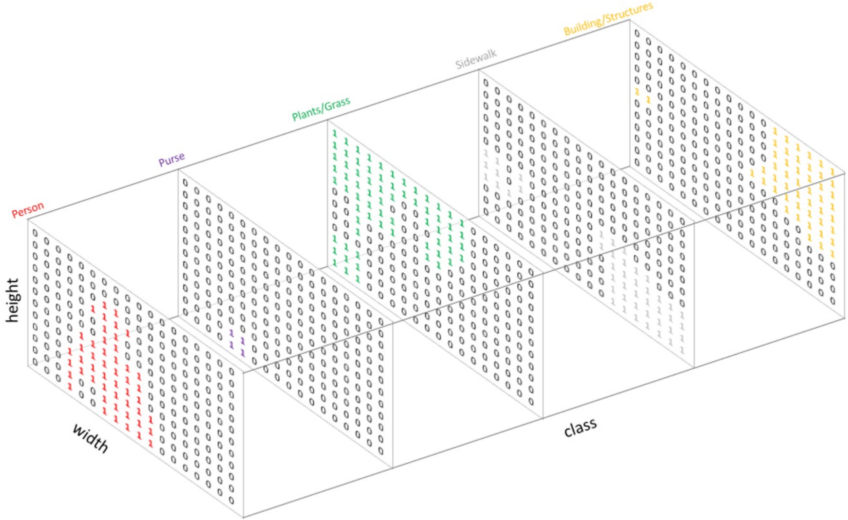
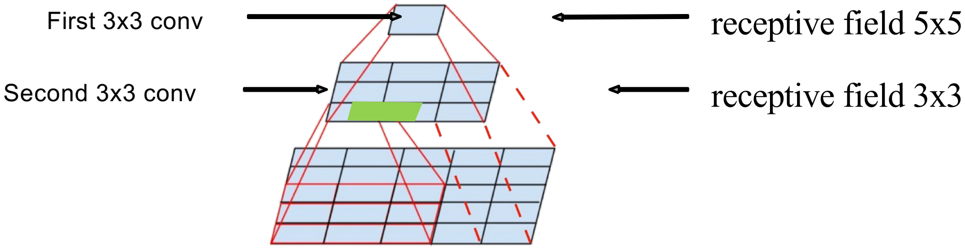


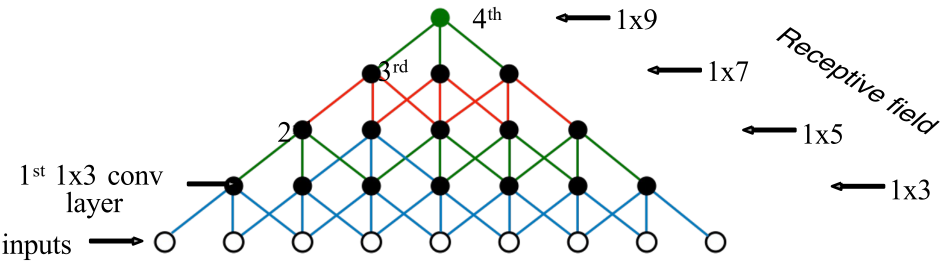
image credits

Receptive Field



We can recognize larger objects by stacking several small convolutions!

Receptive Field



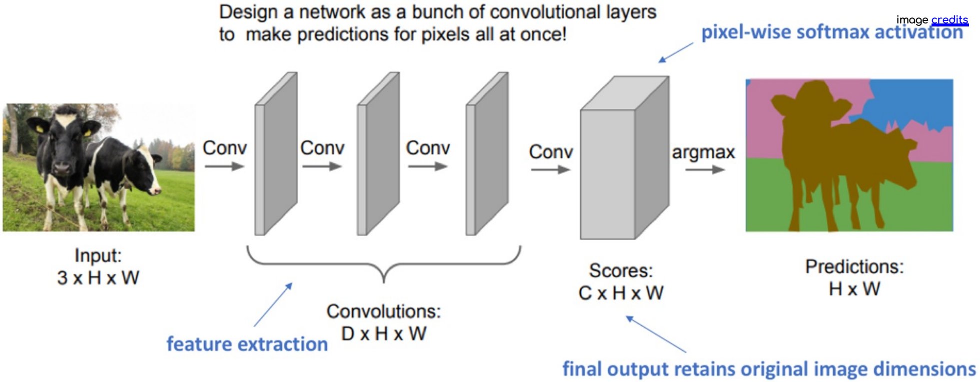
How many 3x3 convolutions we should use to recognize a 100x100px cat

Naive CNNs

Naive approach with CNN:

- stack a number of convolutional layers and output a final segmentation map. No resizing.
- This directly learns a mapping from the input image to its corresponding segmentation through the successive transformation of feature mappings;

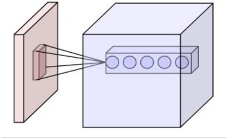
image credits



Quite **computationally expensive to preserve the full resolution throughout the network**.

Pure Convolution

Image : 3 (RGB) x 100 px x 100 px

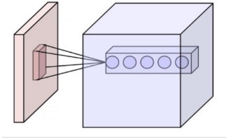


Filters: 100x(3x5x5)

?!

Pure Convolution

Image : 3 (RGB) x 100 px x 100 px



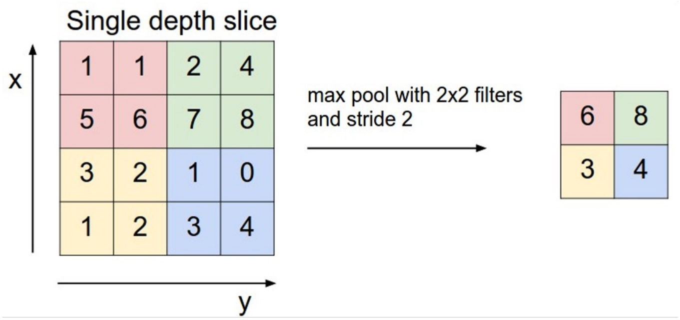
Filters: 100x(3x5x5)

100x96x96

~10^6

Somewhat too many!

Pooling



Intuitively: What is the highest catness over this area?

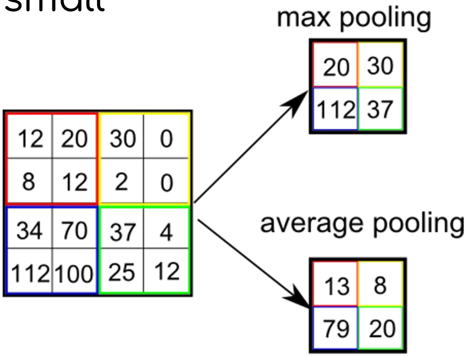
Pooling

Motivation:

- Reduce layer size by a factor
- Make NN less sensitive to small image shifts

Popular types:

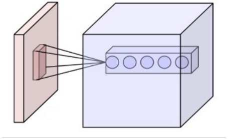
- Max
- Mean(average)



Pool + Convolution

Image : 3 (RGB) x 100 px x 100 px

→



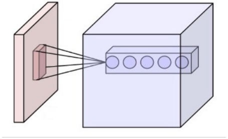
Filters: 100x(3x5x5)

→ 100x96x96

~10^6

100x96x96 → pool 3x4 → ???

Pool + Convolution



100x96x96

~10^6

Filters: 100x(3x5x5)

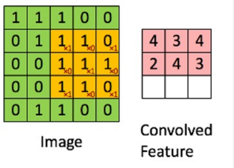
Image : 3 (RGB) x 100 px x 100 px

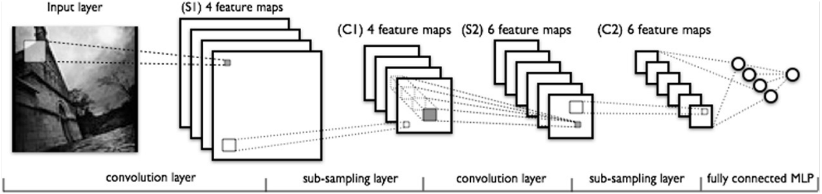
100x96x96 → pool 3x4 → 100x32x32

~10^5

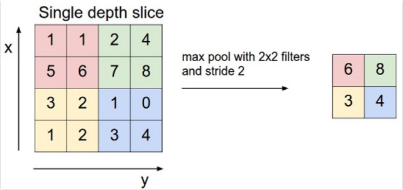
CNN Summary

Convolutional Neural Networks are a specialized kind of neural networks for processing data that has a grid-like structure, such as 2D images. The building block of a CNNs is a layer thatuses discrete convolution in place of general matrix multiplication.

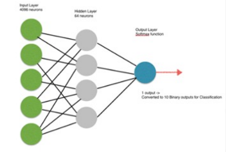




Pooling: its function is to progressively reduce the spatial size of the representation.



Fully connected: Neurons in a fully connected layer have full connections to all activations in the previous layer, as seenin regular Neural Networks. Reduce input to a unique score: *softmax.*



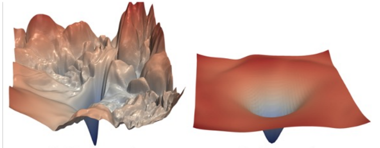
$$\sigma(\mathbf{z})_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}}$$

U-Nets

A drawback of pooling is the loss of spatial information in the downsampling-upsampling process.

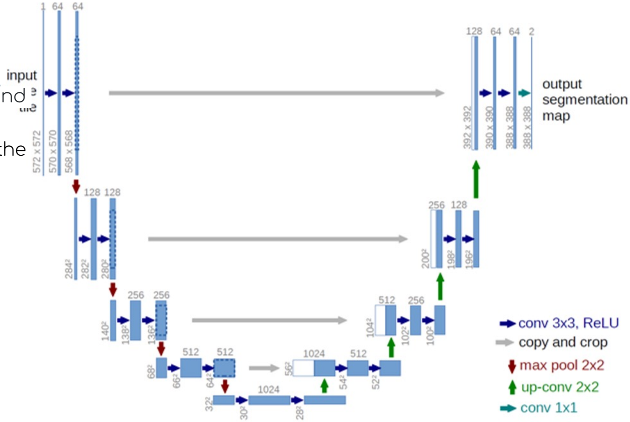
U-shaped architecture (UNet):

- fully convolutional network;
- skip connections allow layers to skip layers and connect to layers further up the network;
- fine-grained details can be recovered in the prediction;
- regularize the loss function surfaces.



(a) without skip connections (b) with skip connections

The loss surfaces of ResNet-56 with/without skip connections from https://arxiv.org/pdf/1712.09913.pdf



Data Augmentation

- Idea: we can get N times more data by tweaking images.
- If you rotate cat image by 15°, it's still a cat
  - Rotate, crop, zoom, flip horizontally, add noise
  - Sound data: apply background noises