Learning through *recreational* research

Google Translate - Word lens

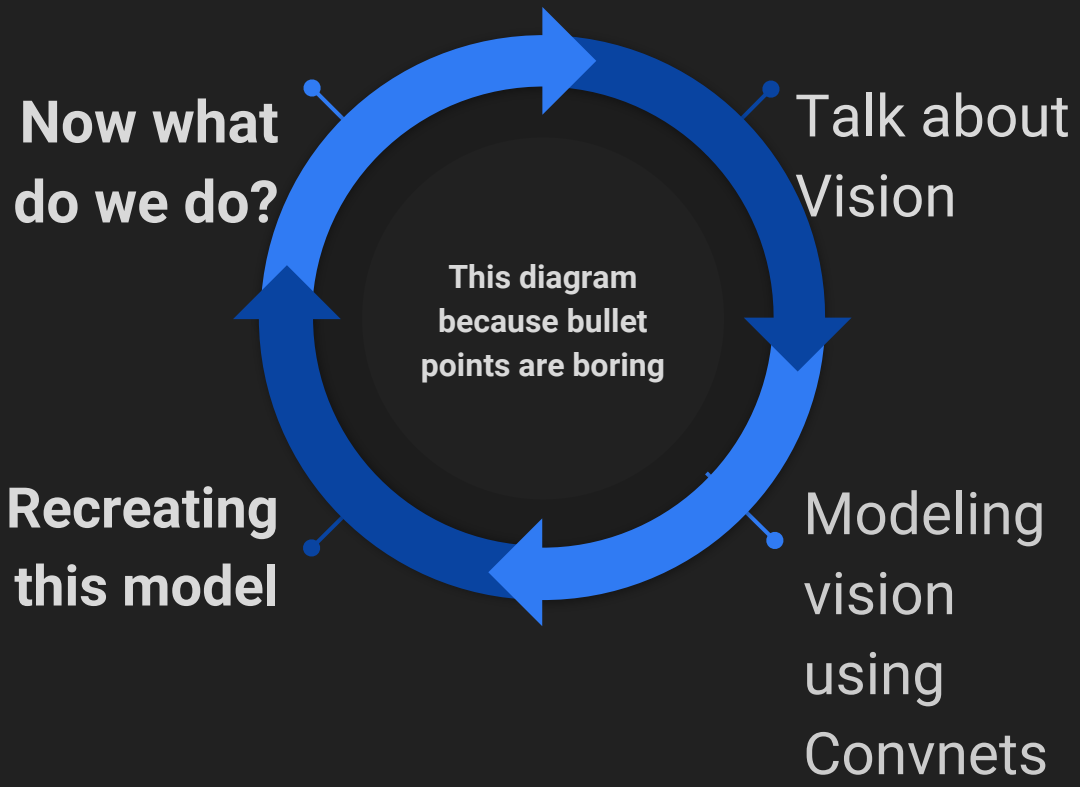

Vision helps us understand what is going on in this scene
Im adding text to this image to make it compliant to DMCA fair use policy

I think it's a close up of a man on fire.

What are we doing here?

- Talking about Computer Vision (more specifically Convnets)
- Showing how Convnets work
- “**Seeing**” how Convnets learn by recreating the landmark NIPS 2012 paper
- Now what do we do?

The process of recreational research



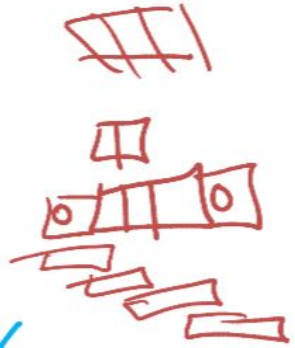
Convolution

$z = \boxed{z_1 \mid z_2}$ $M_z = 2$

$w = \boxed{w_1 \mid w_2} \leftarrow M_F = 2$

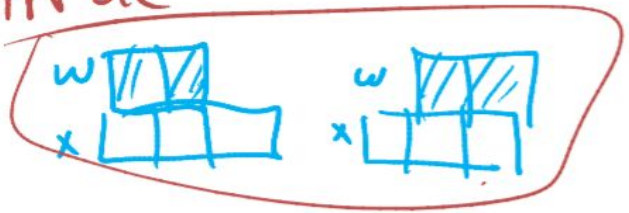
$x = \boxed{x_1 \mid x_2 \mid x_3}$ $M_1 = 3$

$$\begin{cases} z_1 = w_1 x_1 + w_2 x_2 \\ z_2 = w_1 x_2 + w_2 x_3 \end{cases}$$



zero padding

Stride



$$z_{i'} = \sum_{i=1}^{M_F=2} w_i x_{i'+i-1}$$

Correlation (similarity)

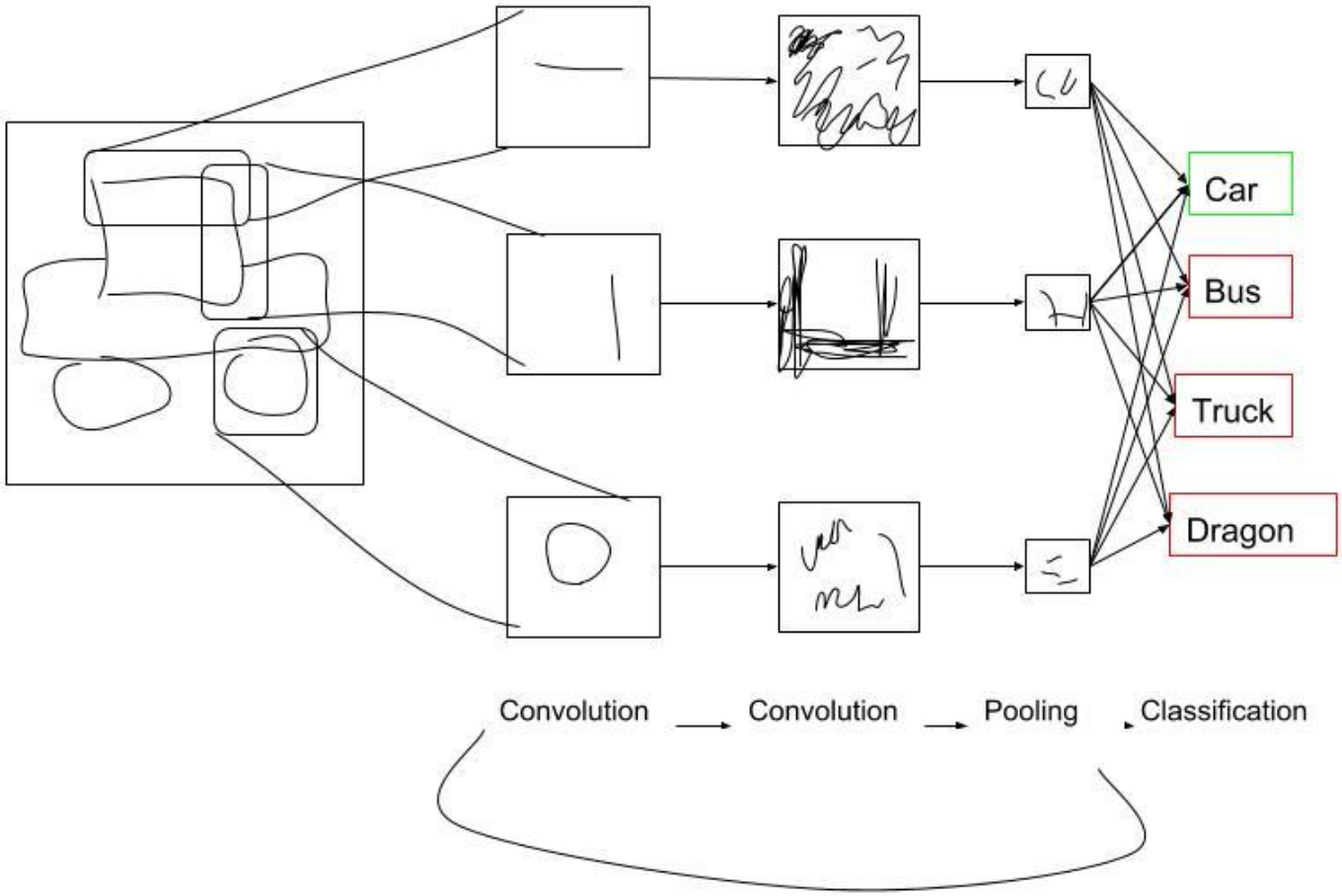
$$\|$$

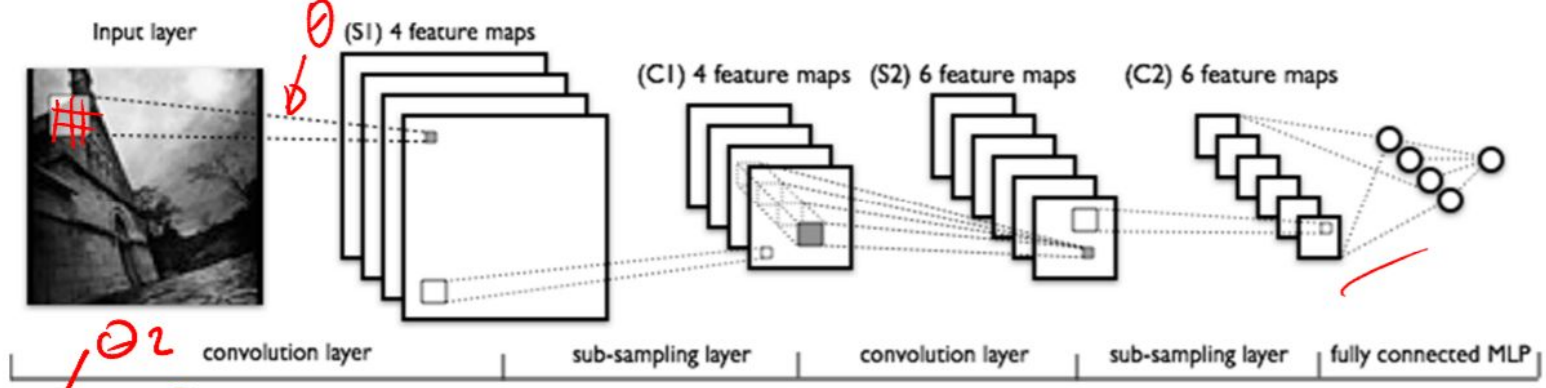
$$z_{i'} = \sum_{i=1}^{M_F=2} x_{i'+i-1} \bar{w}_{M_F-i+1}$$

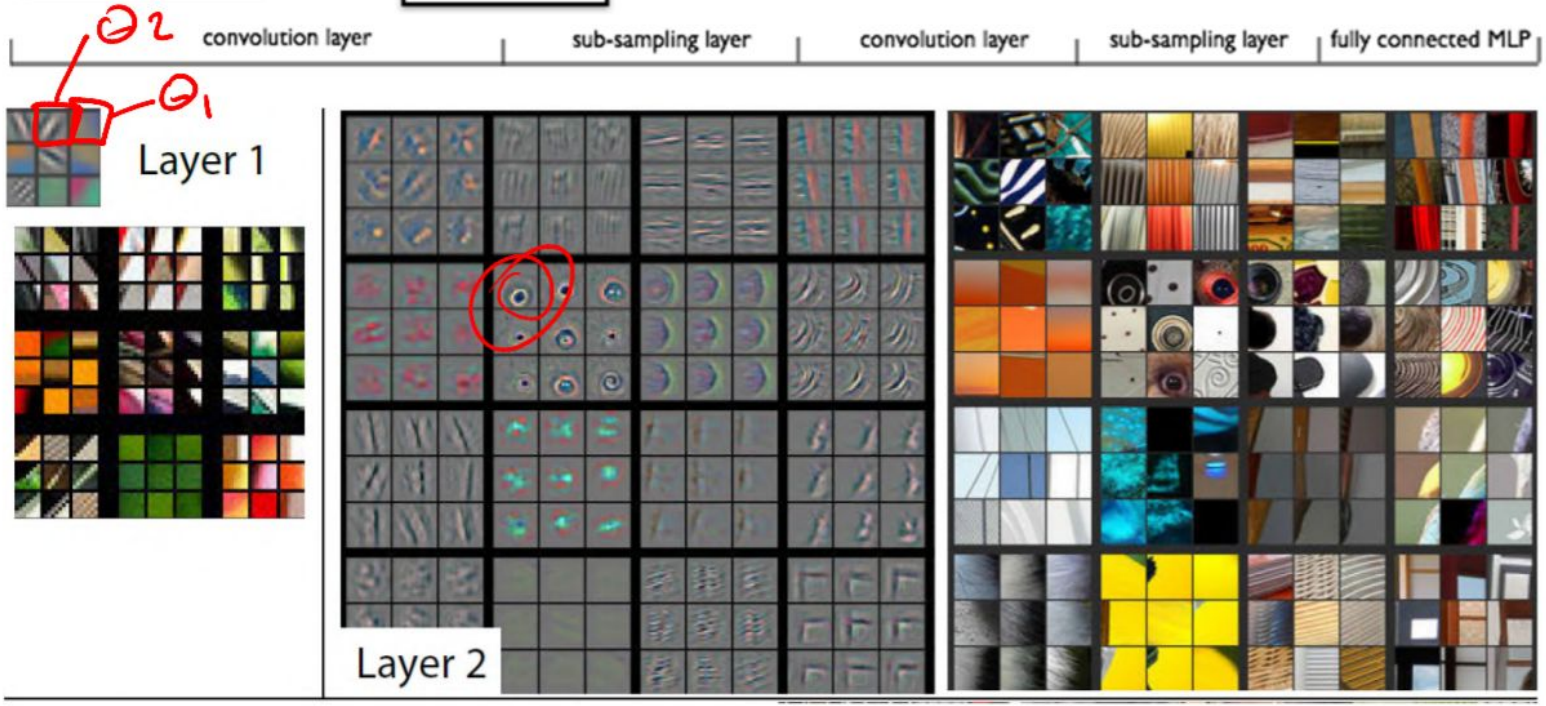
(convolution)

flip $\vec{w} = \bar{w} = \boxed{w_2 \mid w_1}$

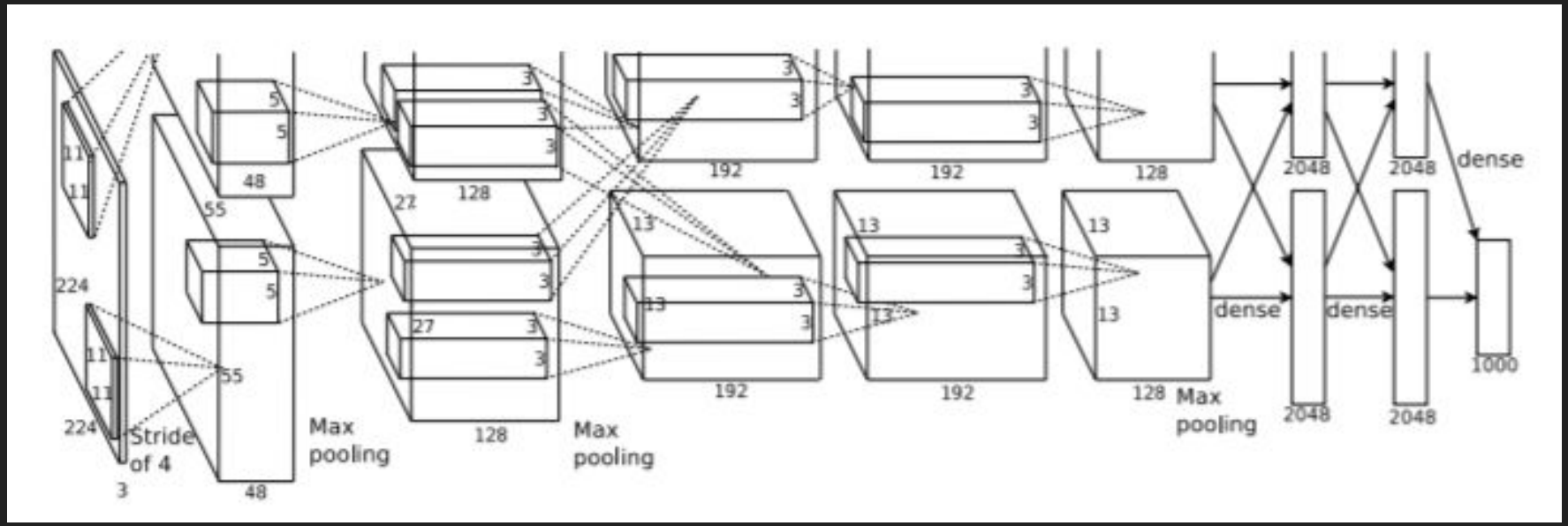
https://www.cs.ox.ac.uk/people/nando.defreitas/machinelearning/lecture9.pdf


Car
Bus
Truck
Dragon
Convolution
Convolution
Pooling
Classification





[Matthew Zeiler & Rob Fergus]



Jump to Jupyter notebook for recreating this model
and jump to Section 3.5

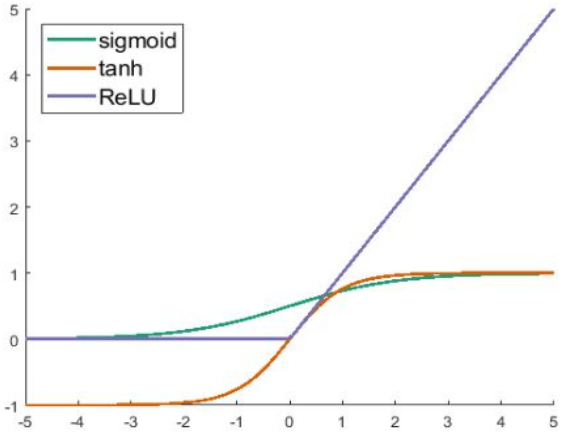
Activation function

- Traditionally, saturating nonlinearities:
  - hyperbolic tangent function: $f(x) = \tanh(x) = 2 * \frac{1}{1+e^{-2x}} - 1$
  - sigmoid function: $f(x) = \frac{1}{1+e^{-x}}$

  → slow to train

- Non-saturating nonlinearity:
  - Rectified Linear Unit (ReLU): $f(x) = \max(0, x)$

  → quick to train



http://cvml.ist.ac.at/courses/DLWT_W17/material/AlexNet.pdf

Local Response Normalization

- ReLUs do not require input normalization to prevent them from saturating
- However, Local Response Normalization aids generalization

Activity of a neuron by applying kernel i at position (x,y)

$$b^i_{x,y} = a^i_{x,y} \Big/ \left(k + \alpha \sum_{j=\max\left(0, i-\frac{n}{2}\right)}^{\min\left(N-1, i+\frac{n}{2}\right)} \left(a^j_{x,y}\right)^2 \right)^{\beta}$$

$k = 2$
$n = 5$
$\alpha = 10^{-4}$
$\beta = 0.75$

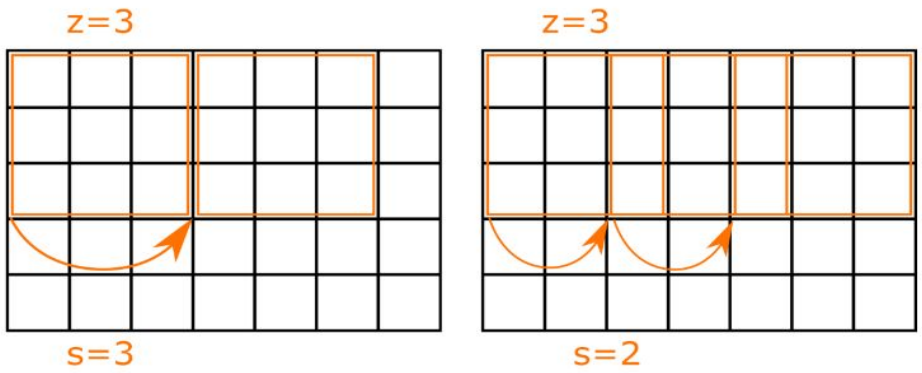
- Improvement:
  - top-1 error rate by 1.4%
  - top-5 error rate by 1.2%

sum runs over n "adjacent" kernel maps at the same spatial position

http://cvml.ist.ac.at/courses/DLWT_W17/material/AlexNet.pdf

Overlapping Pooling

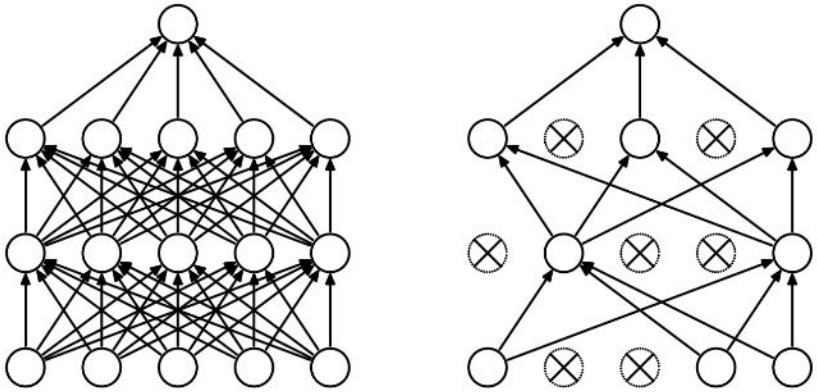
- Pooling layers summarize the outputs of neighboring neurons in the same kernel map.
- Overlapping pooling → s < z



- Improvement using MaxPooling:
  - top-1 error rate by 0.4%
  - top-5 error rates by 0.3%

http://cvml.ist.ac.at/courses/DLWT_W17/material/AlexNet.pdf

Reducing Overfitting - Dropout



(a) Standard Neural Net (b) After applying dropout.

- Output of each hidden neuron is set to zero with probability 0.5
- Learning more robust features
- Doubles the number of iterations required to converge
- Applied in the first two fully connected layers

[N. Srivastava et al., *Dropout: A Simple Way to Prevent Neural Networks from Overfitting*, 2014]

Stochastic Gradient Descent

- SGD with a batch size of 128
- Learning rate initialized at 0.01; divided by 10 if validation error rate stopped improving
- Update rule for weight w:

weight decay

learning rate

$$v_{i+1} := 0.9 * v_i - 0.0005 * \epsilon * w_i - \epsilon * \left\langle \left.\frac{\partial L}{\partial w}\right|_{w_i} \right\rangle_{D_i}$$

$$w_{i+1} := w_i + v_{i+1}$$

momentum

Gradient of Loss

- ~ 90 cycles → five to six days on two NVIDIA GTX 580 3GB GPUs

http://cvml.ist.ac.at/courses/DLWT_W17/material/AlexNet.pdf

“The idea he (Alex Krizhevsky) had at the time was just to re-implement the original Lenet architecture”

:w !sudo tee %
@GianArb

Follow

You are a populist software engineer when you try to fix a problem without trying to understand it

8:56 PM - 18 Aug 2018

The process of recreational research