TTIC 31230, Fundamentals of Deep Learning

David McAllester, April 2017

Interpreting Deep Networks

The Black Box Problem

The Human Black Box — Perception

Introspection is notoriously inadequate for AI.

Explain how you know there are upside down glasses in this picture.

The Human Black Box — Inference

Certain facts are obvious.

A king on empty chess board can reach every square (obvious).

A knight on an empty chess board can reach every square (true but not obvious).

The Human Black Box — Inference

Consider a graph with colored nodes.

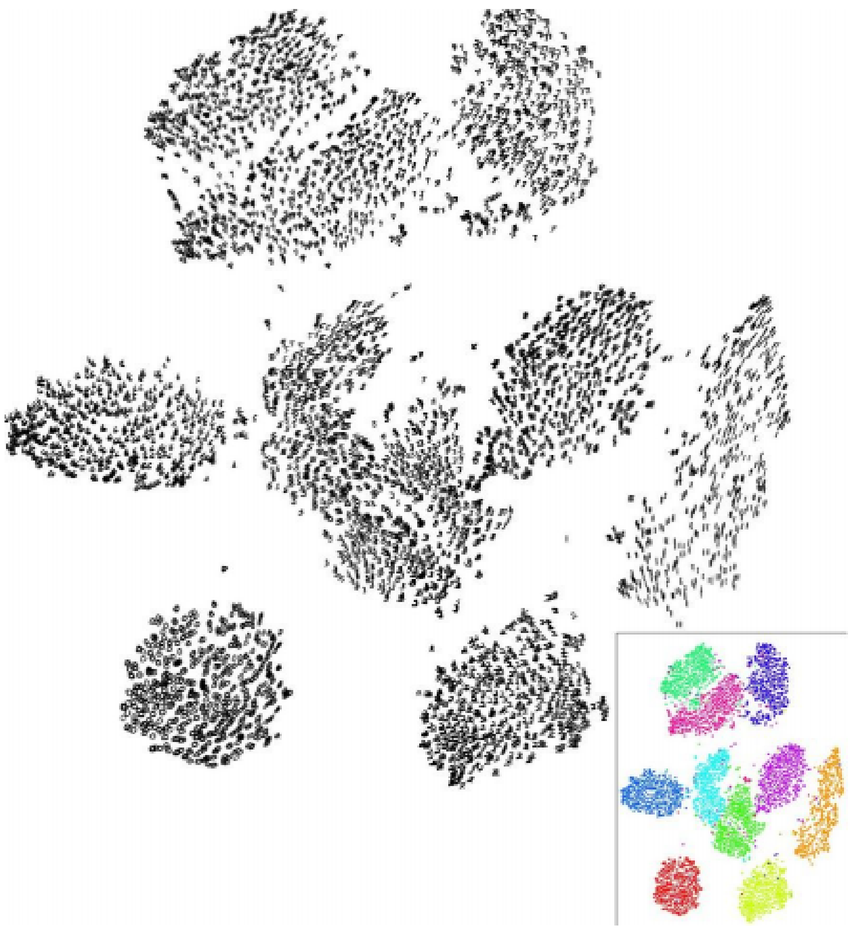
If every edge is between nodes of the same color, then any path connects nodes of the same color.

Consider a swiss chocolate bar of 3×5 little squares.

How many breaks does it take to reduce this to fifteen unconnected squares?

Dimensionality Reduction

Visualizing the representation



t-SNE visualization

[van der Maaten & Hinton]

Embed high-dimensional points so that locally, pairwise distances are conserved

i.e. similar things end up in similar places. dissimilar things end up wherever

Right: Example embedding of MNIST digits (0-9) in 2D

[Stanford CS231]

t-SNE

Consider high dimensional data $x_1, \ldots, x_N$ with $x \in \mathbb{R}^d$.

$$P(j|i) = \frac{1}{Z_i} \exp\left(\frac{-||x_i - x_j||^2}{2\sigma_i^2}\right)$$

Set σ_i such that $H(P(j|i)) = \ln k$ (soft k nearest neighbors).

$$P(i,j) = P(i)P(j|i) = \frac{1}{N}\, P(j|i)$$

t-SNE

Let $Y = \{y_i \ldots, y_N\}$ be an assignment of a vector with $y_i \in \mathbb{R}^2$ to each high dimensional point x_i.

$$Q_Y(i,j) = \frac{1}{Z}\left(\frac{1}{1 + ||y_i - y_j||^2}\right)$$

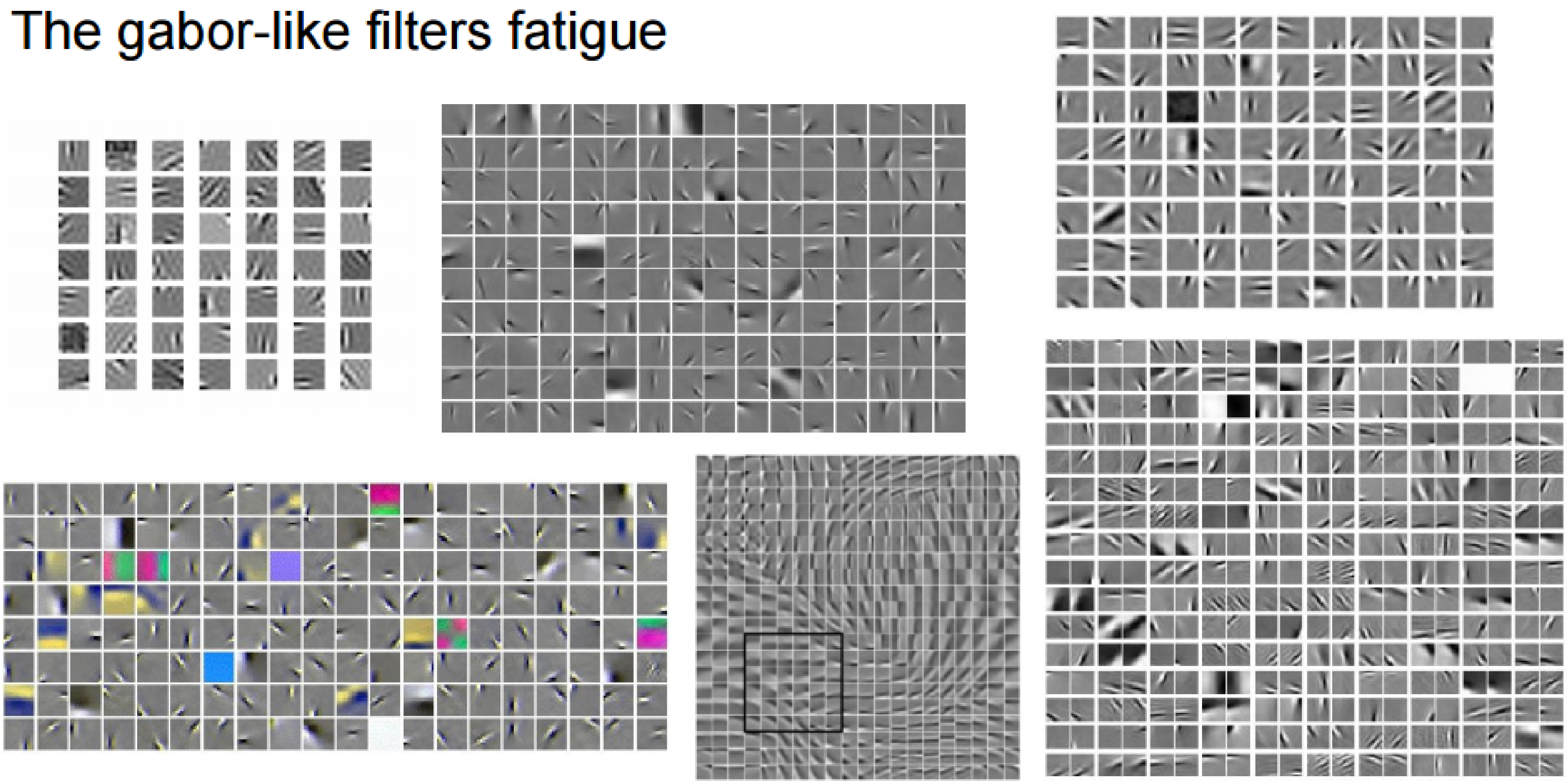
$$Y^* = \underset{Y}{\operatorname{argmin}}\ \mathrm{KL}(P, Q_Y)$$

t-SNE vs. Projection Modeling

t-SNE — $y(x)$ is defined by a table on the data points.

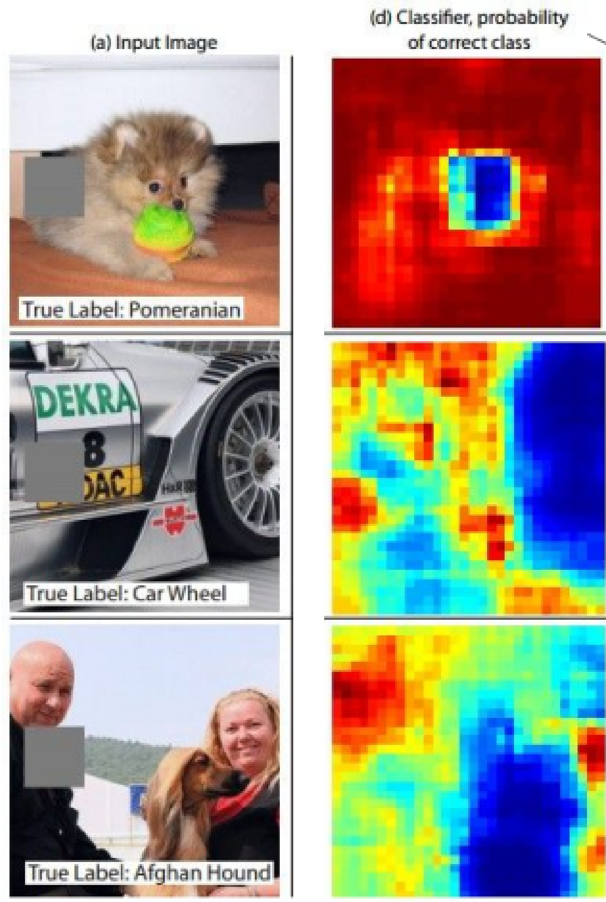
In PCA or Isomap we have $y_\Phi(x) \in \mathbb{R}^2$ for a parameterized function y_Φ.

Visualizing the Filters



[Stanford CS231]

Occlusion experiments
[Zeiler & Fergus 2013]



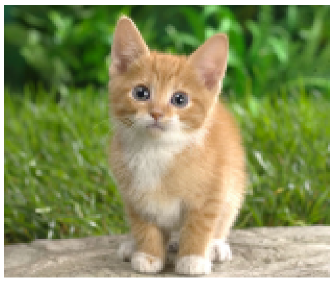
(as a function of the position of the square of zeros in the original image)

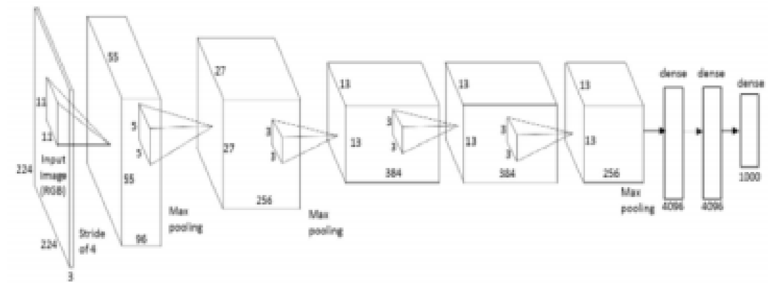
[Stanford CS231]

Backpropagation from Individual Neurons

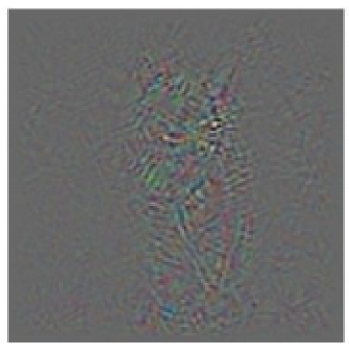
Deconv approaches

1. Feed image into net





2. Pick a layer, set the gradient there to be all zero except for one 1 for some neuron of interest
3. Backprop to image:



"Guided backpropagation:" instead

[Stanford CS231]

Guided Backpropagation

Rather than $\partial\ell/\partial x$ we are interested in $\partial\mathrm{neuron}/\partial x$.

We are interested in $\partial\mathrm{neuron}/\partial x$ where x in one color channel of one input pixel.

It turns out that $\partial\mathrm{neuron}/\partial x$ looks like image noise.

Instead we compute $x.\mathrm{ggrad}$ — a **guided** version of $\partial\mathrm{neuron}/\partial x$.

Guided Backpropagation

Guided backpropagation only considers computation paths that activate (as opposed to suppress) the neuron all along the activation path.

The backpropagation at activation functions is modified.

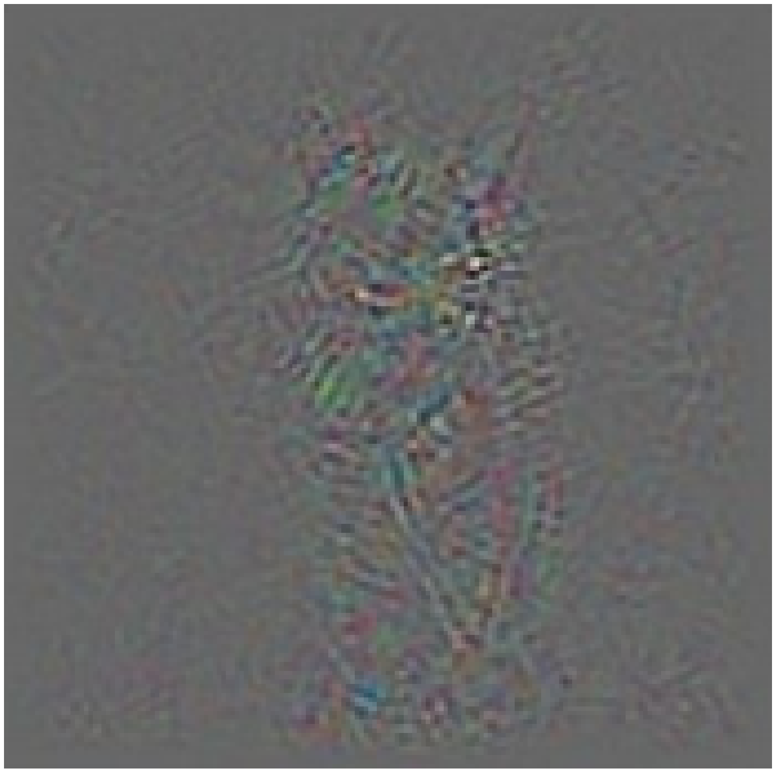
For a neuron y with $y = s(x)$ for activation function s:

$$x.\mathrm{ggrad} = \mathbf{1}[y.\mathrm{ggrad} > 0]\; y.\mathrm{ggrad}\; ds/dx$$

Guided Backpropagation

Guided Backpropagation

[Zeigler and Fergus 2013]



[Zeigler and Fergus 2013]

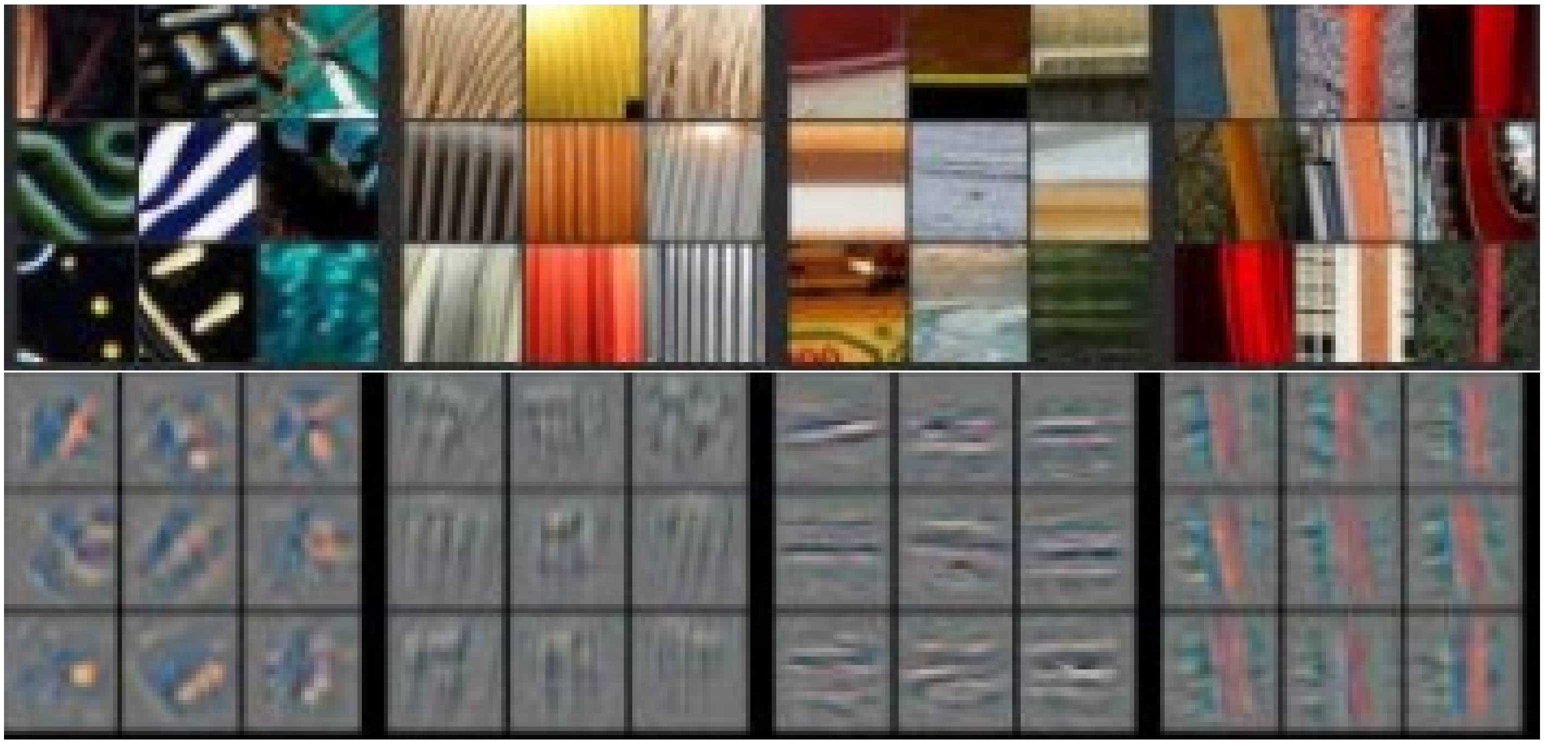
Neural Network Neuroscience

[Zeigler and Fergus 2013]

Take a neuron (linear threshold unit) and select the images causing the greatest response of that Neuron.

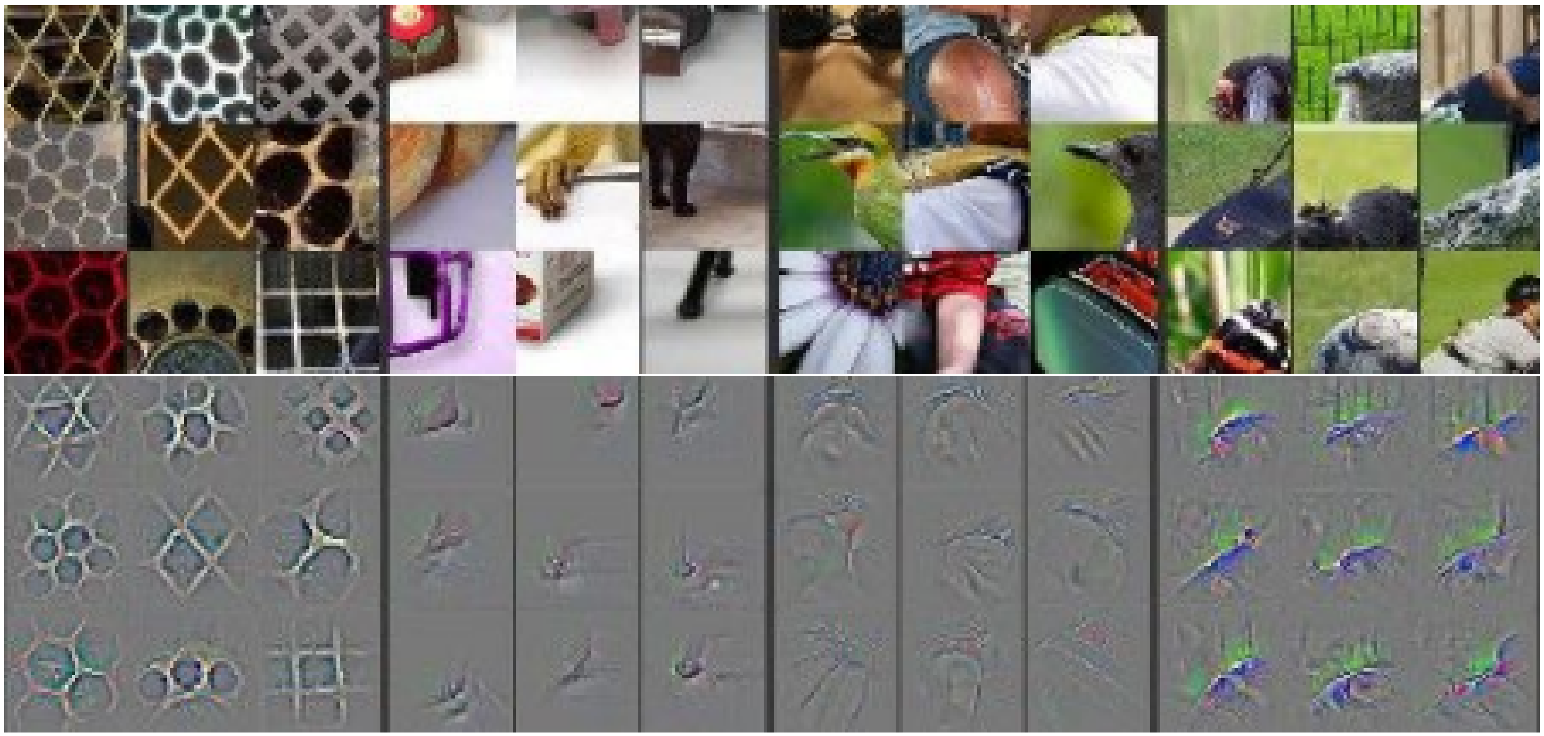
Do guided backpropagation from that neuron onto the image.

Guided Backpropagation Layer 2



[Zeigler and Fergus 2013]

Guided Backpropagation Layer 3



[Zeigler and Fergus 2013]

Guided Backpropagation Layer 4

[Zeigler and Fergus 2013]

Guided Backpropagation Layer 5

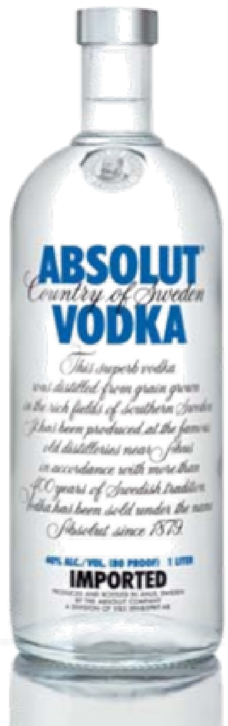
[Zeigler and Fergus 2013]

A Wheel or Face Detector

The nine strongest stimulators of the "wheel or face cell" are the following.

[Zeigler and Fergus 2013]

it's like "*vodka & potato*" *classifier!*



+

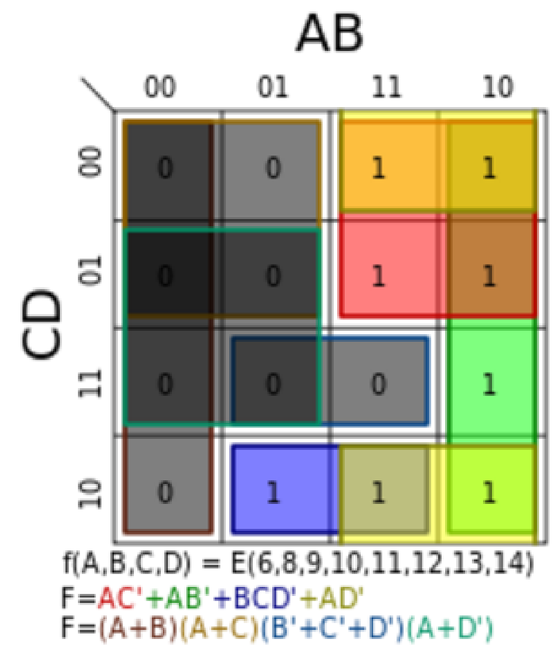
[Alyosho Efros]

The Kaurnaugh Model of DNNs

The Karnaugh map, also known as the K-map, is a method to simplify boolean algebra expressions.

Truth table of a function

	A	B	C	D	f(A, B, C, D)
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	1	0	0
3	0	0	1	1	0
4	0	1	0	0	0
5	0	1	0	1	0
6	0	1	1	0	1
7	0	1	1	1	0
8	1	0	0	0	1
9	1	0	0	1	1
10	1	0	1	0	1
11	1	0	1	1	1
12	1	1	0	0	1
13	1	1	0	1	1
14	1	1	1	0	1
15	1	1	1	1	0



$$
\begin{aligned}
F(A,B,C,D) &= AC' + AB' + BCD' + AD' \\
&= (A+B)(A+C)(B'+C'+D')(A+D')
\end{aligned}
$$

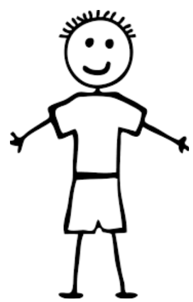
A Kaurnaugh Person Detector

Wheel or Face

Hand or Flower

Hand or Flower

Leg or Tree



Leg or Tree

The set of locally minimal models (circuits) could be vast (exponential) without damaging performance.

Is a Boolean circuit a distributed representation?

The Glass Model of SGD

Pysical glass (ordinary silica glass) is a metastable state — the ground state is quartz crystal.

As molten glass cools there is a temperature T_g (± 1 degrees C) at which it "solidifies" (the viscosity becomes huge).

This soldification process is very repeatable with a well defined final energy.

However, the local optimum achieved is presumably very different for each instance of cooling.

Identifying Channel Correspondences

Convergent Learning: Do Different Neural Networks Learn The Same Representations?, Li eta al., ICLR 2016.

Train Alexnet twice with different initializations to get net1 and net2.

For each convolution layer, each channel i of net1, and each channel j of net2, compute their correlation.

$$\rho_{i,j} = \mathrm{E}\left[\frac{(u_i - \mu_i)(u_j - \mu_j)}{\sigma_i \sigma_j}\right]$$

Semi-matching and Bipartite matching

Semi-matching: for each i in net1 find the best j in net2:

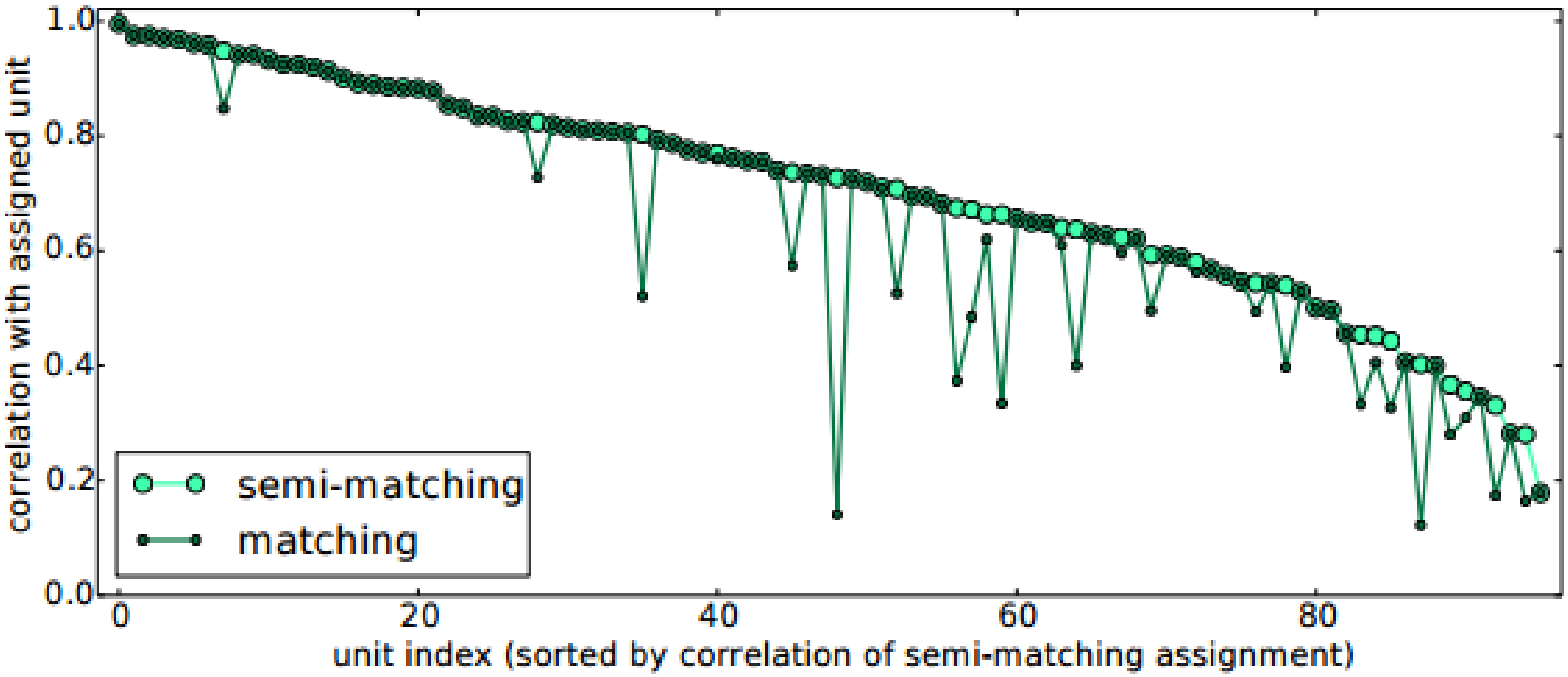
$$\hat{j}(i) = \operatorname*{argmax}_j \rho_{i,j}$$

Biparetitie Matching: Find the best one-to-one correspondence.

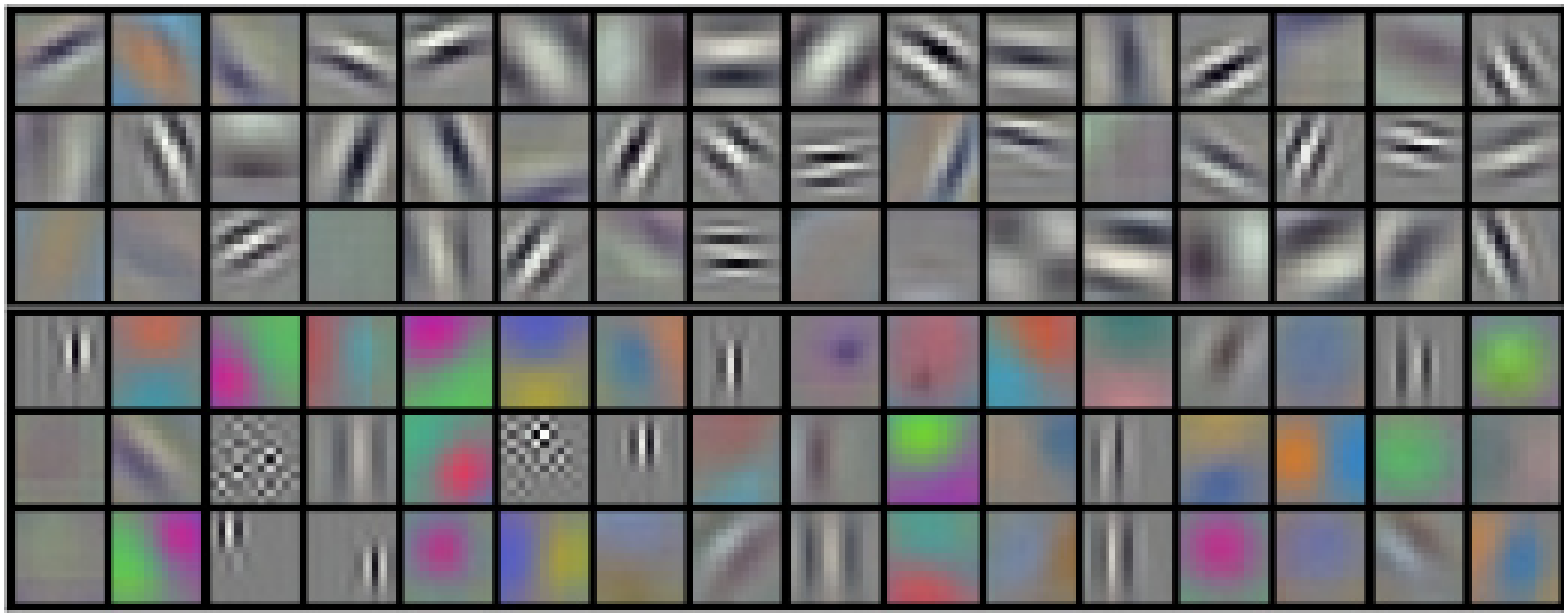
$$\hat{j} = \operatorname*{argmax}_{\hat{j} \text{ a bijection}} \sum_i \rho_{i,\hat{j}(i)}$$

Bipartite matching can be solved by a classical algorithm [Hopcroft and Karp, 1973]. John Hopcroft (age 77) is an author on this ICLR paper.

Correlations at Layer 1 (Wavelet Layer)

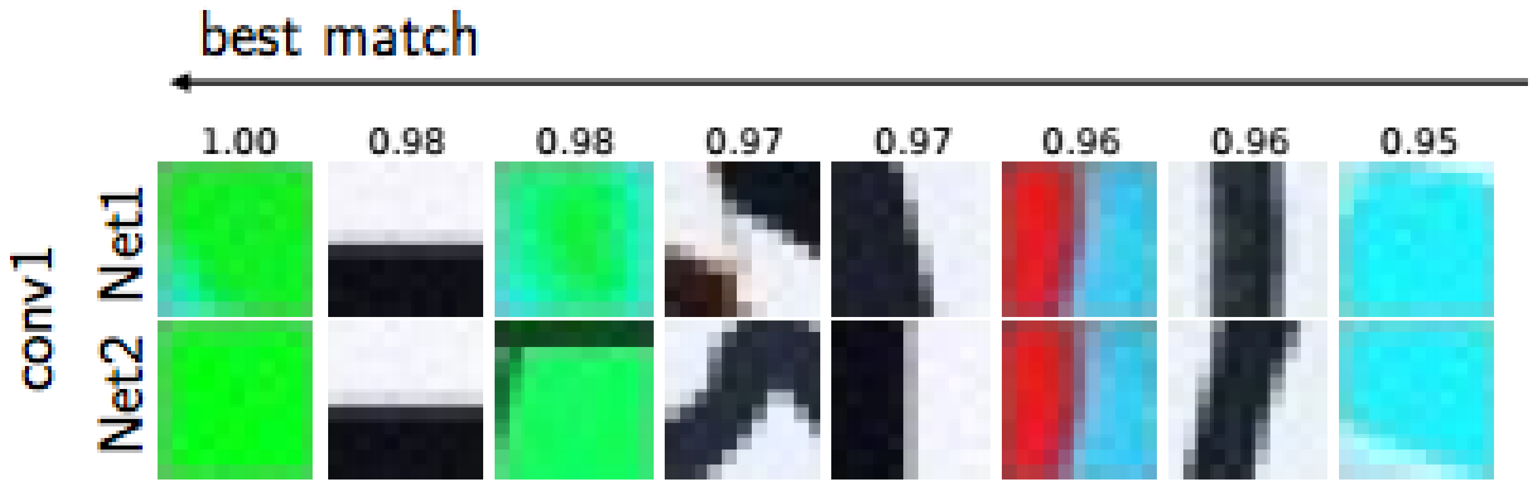


Alexnet Layer 1



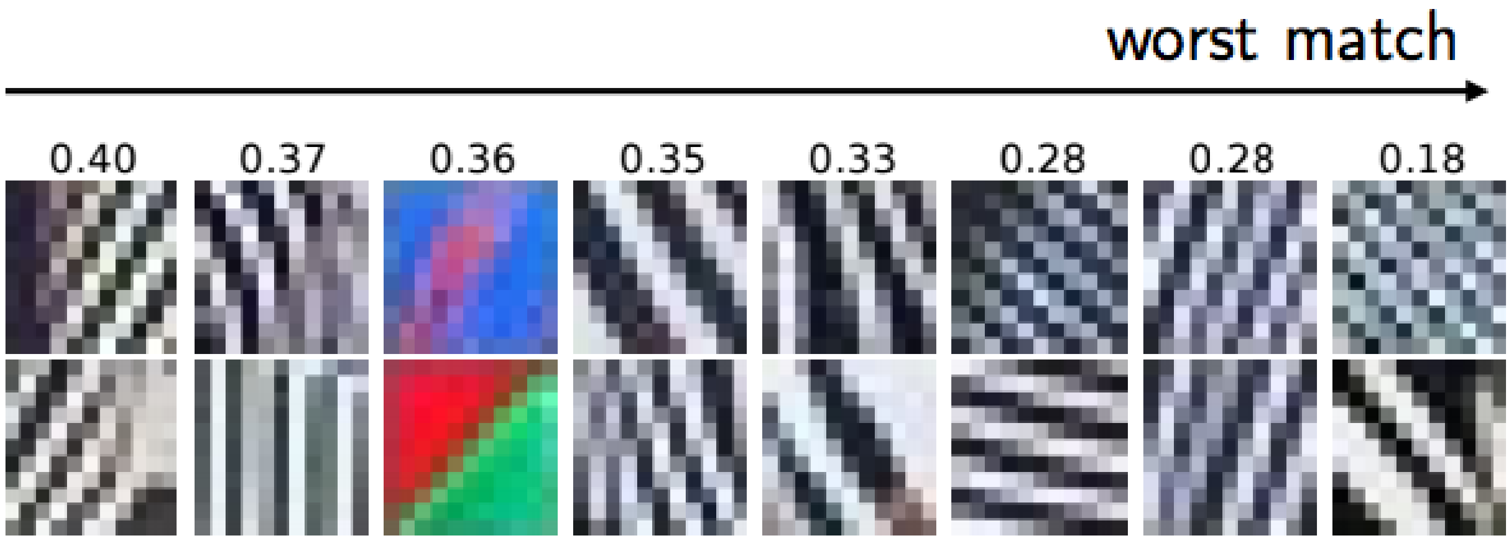
[Krizhevsky et al.]

Best Matches in Layer 1 semi-matching



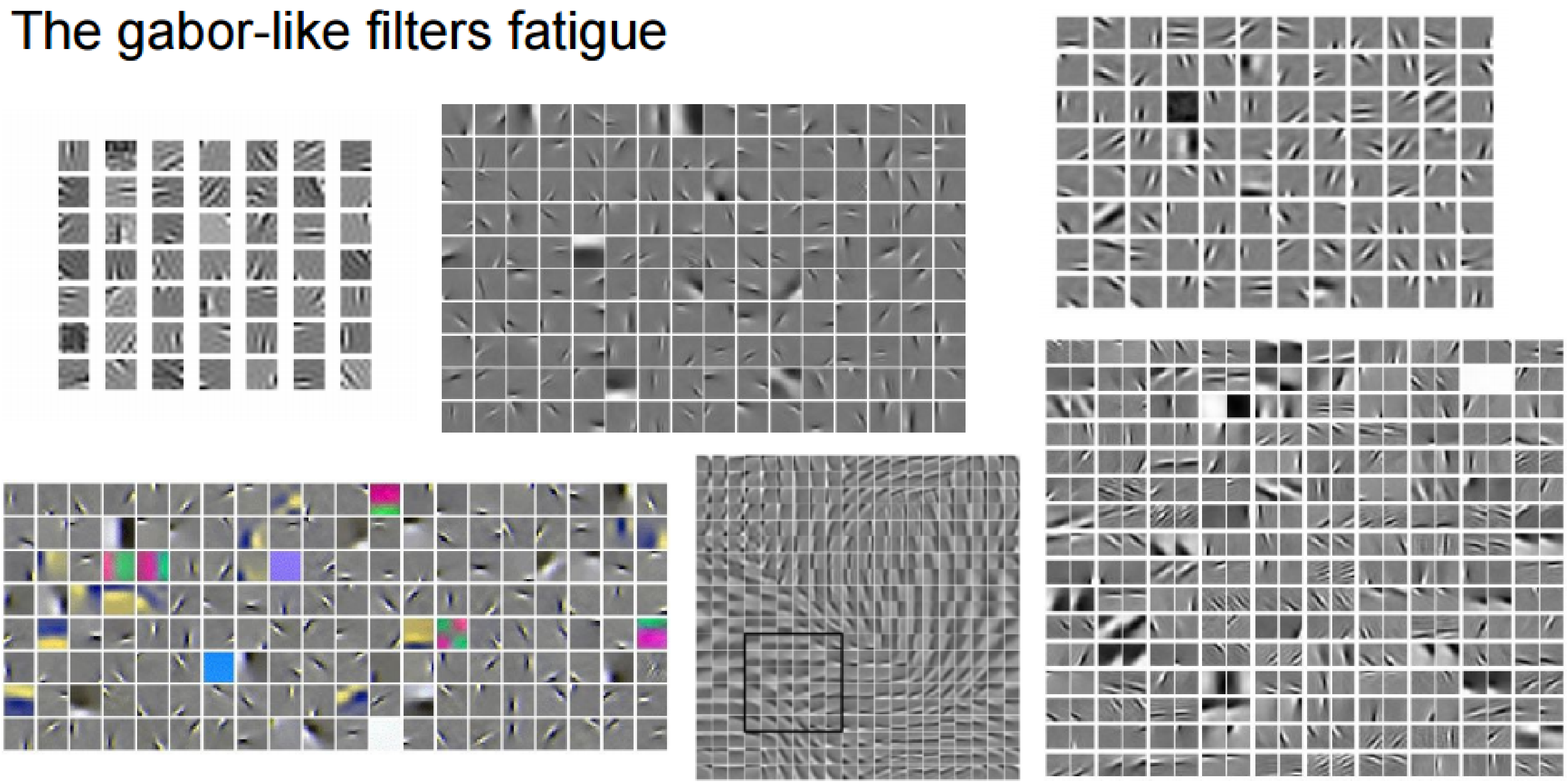
[Li et al.]

Worst Matches in Layer 1 semi-matching



[Li et al.]

Layer 1 in Other Networks



[Stanford CS231]

Regression Between Networks at Layer 1

Model each channel of net1 as a linear combination of channels of net2 using least squares regression.

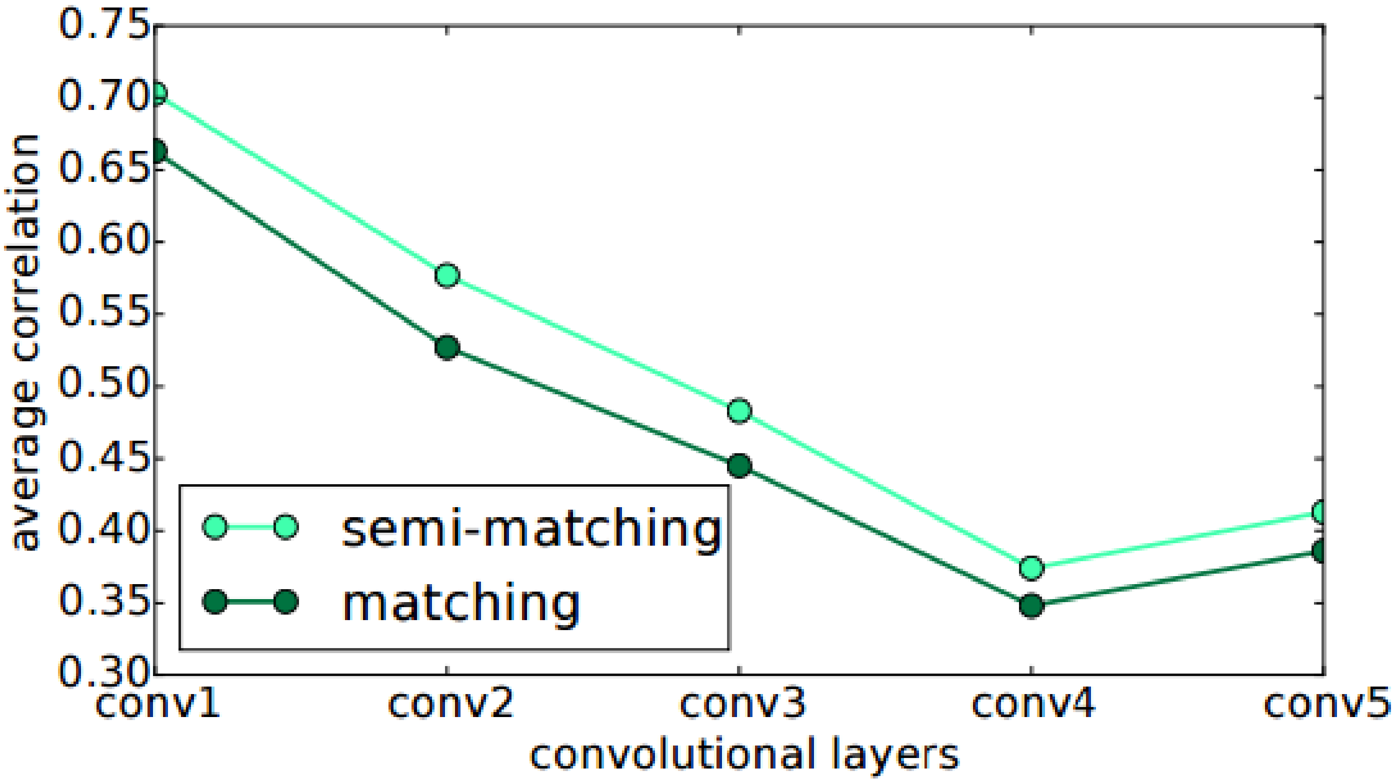
Before the regression each channel is normalized to have zero mean and channel variance.

No correlation would yield a square loss of 1.000.

No regularization gives a square loss of 0.170 and uses 96 channels in each prediction.

L1 regularization gives a square loss of 0.235 and uses 4.7 channels in each prediction.

Deeper Layers



In the regression experiment squared error was not significantly reduced at layers 3 through 5 even without regularization.

SVCCA Analysis

Raghu et al., November 2017

Consider a matrix $W[i, j]$ used in some model as

$$y[i] = \sigma\left(\sum_j W[i, j]x[j] + b[i]]\right)$$

We want to understand the meaning of the matrix $W[i, j]$ and the vector $y[I]$.

SVCCA

$$y[i] = \sigma\left(\sum_j W[i,j]x[j] + b[i]\right)$$

We can collect a set $y[t, I]$ of vectors y computed from $x[t, J]$

$$y[t,i] = \sigma\left(\sum_j W[i,j]x[t,j] + b[i]\right)$$

Here t can range over different inputs to the entire network, or different image locations where the filter W is used, or different times with a single RNN execution.

PCA on the matrix outputs

We can then perform PCA over the vectors $y[t, I]$ to find a a reduced set of covariance eigenvectors $B[k, I]$. and represent y by its projection on the eigenvectors.

$$\tilde{y}[t, k] = B[k, I]^\top (y[t, I] - \mu[I])$$

PCA can be defined by

$$B^* = \underset{B}{\operatorname{argmin}} \sum_t \left\| y[t, I] - \mu[I] + \sum_j \tilde{y}[t, k] B[k, I] \right\|^2$$

PCA reduction

Now consider a layer that uses $y[I]$ and the tensor **before the nonlinearity**.

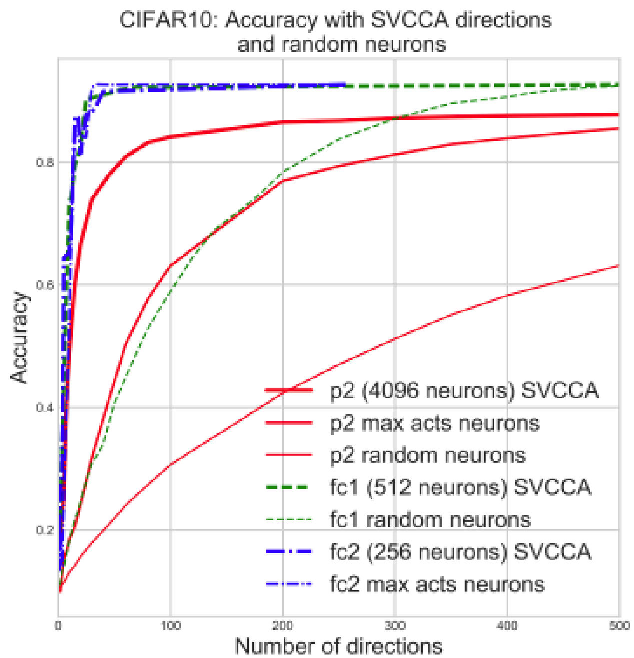
$$z[t,v] = \sum_i W'[v,i]y[i] + B'[v]$$

This layer can now be redefined to use the $\tilde{y}$

$$z[t,v] = \sum_i W''[v,k]\tilde{y}[t,k] + B''[v]$$

$$W''[v,k] = \sum_i W'[v,i]B[k,i]$$

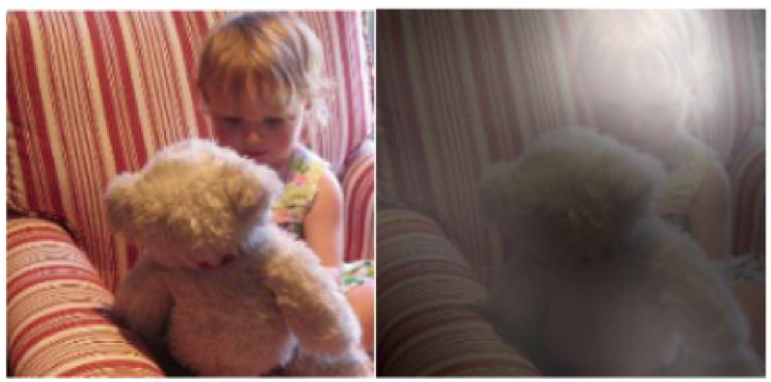
Reduction



CIFAR10: Accuracy with SVCCA directions
and random neurons
Accuracy
Number of directions
0
100
200
300
400
500
0.2
0.4
0.6
0.8
p2 (4096 neurons) SVCCA
p2 max acts neurons
p2 random neurons
fc1 (512 neurons) SVCCA
fc1 random neurons
fc2 (256 neurons) SVCCA
fc2 max acts neurons

Attention as Explanation

A woman is throwing a frisbee in a park.



A little girl sitting on a bed with a teddy bear.

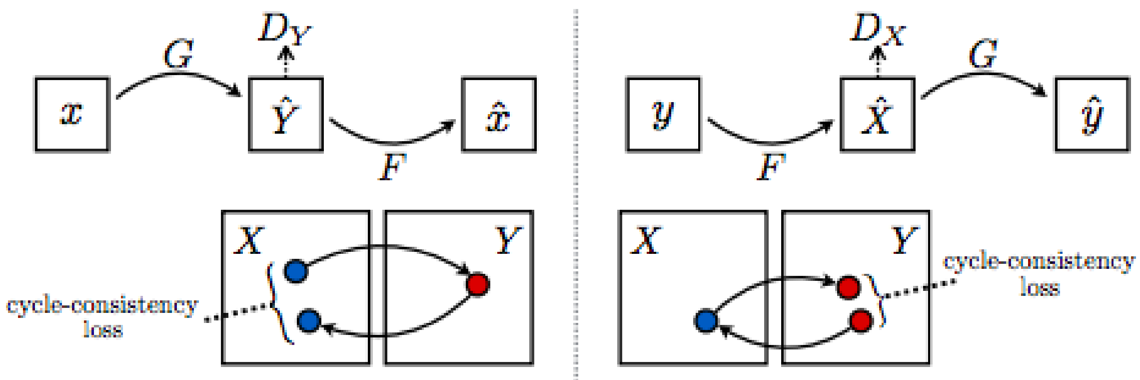
Xu et al. ICML 2015

Interpretation from Domain Correspondence

Kushner verliert den Zugang zu streng geheimen Informationen.

$$\Rightarrow$$

Kushner loses access to top-secret intelligence.



Causal Models are Explicitly Interpretable

Flu causes symptoms x, y z.

Strep causes symptoms x, y, u.

For the given information on the patient, the prior probability for flu is ...

Can Alpha Zero Explain Chess Moves?

I did x because if I did y they would do z and, in that case, $\ldots$.

END