U. Amsterdam

Canadian Institute for Advanced Research

Physics for Deep Learning & Deep Learning for Physics

Max Welling

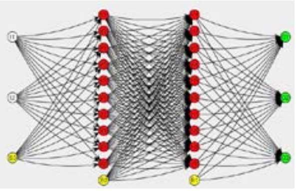
Uva-Qualcomm Lab

UvA - BOSCH DELTA LAB

Amsterdam Machine Learning Lab

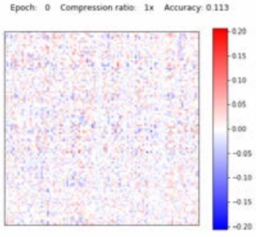
Overview

- Part I: Deep Learning 101



- Part II: Symmetries for Deep Learning

-

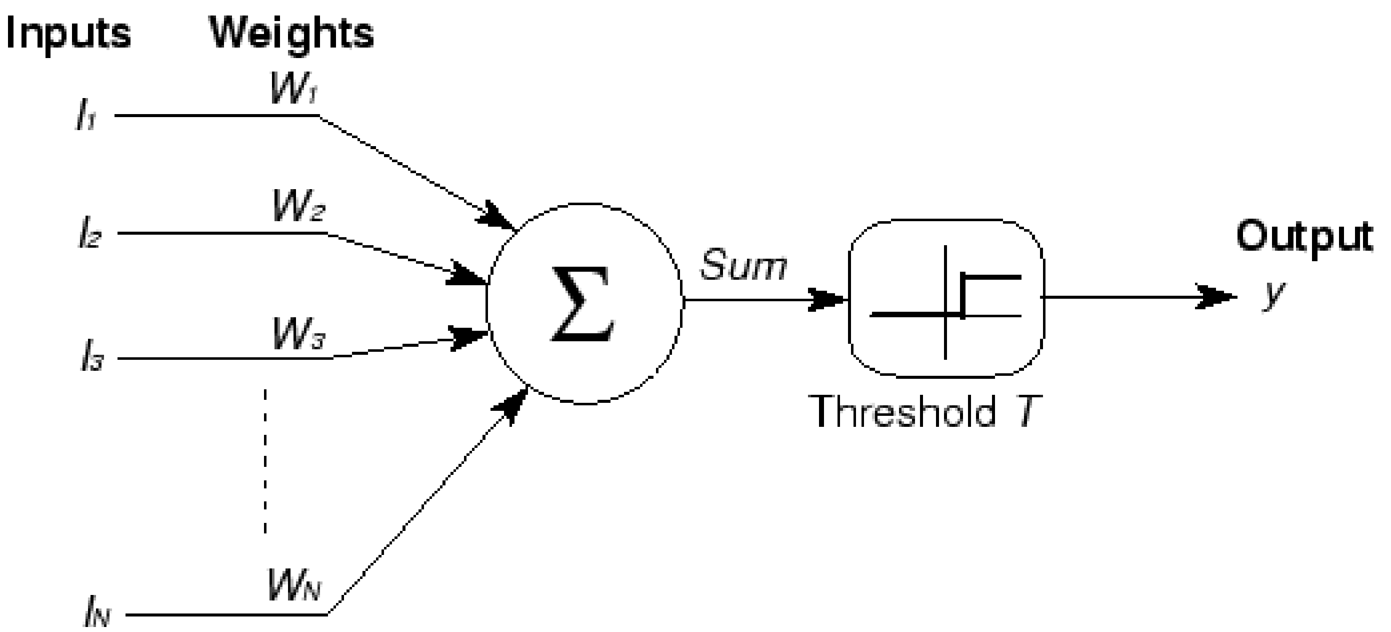


- Part III: Statistical Physics of Deep Learning

- Part IV: Deep Art

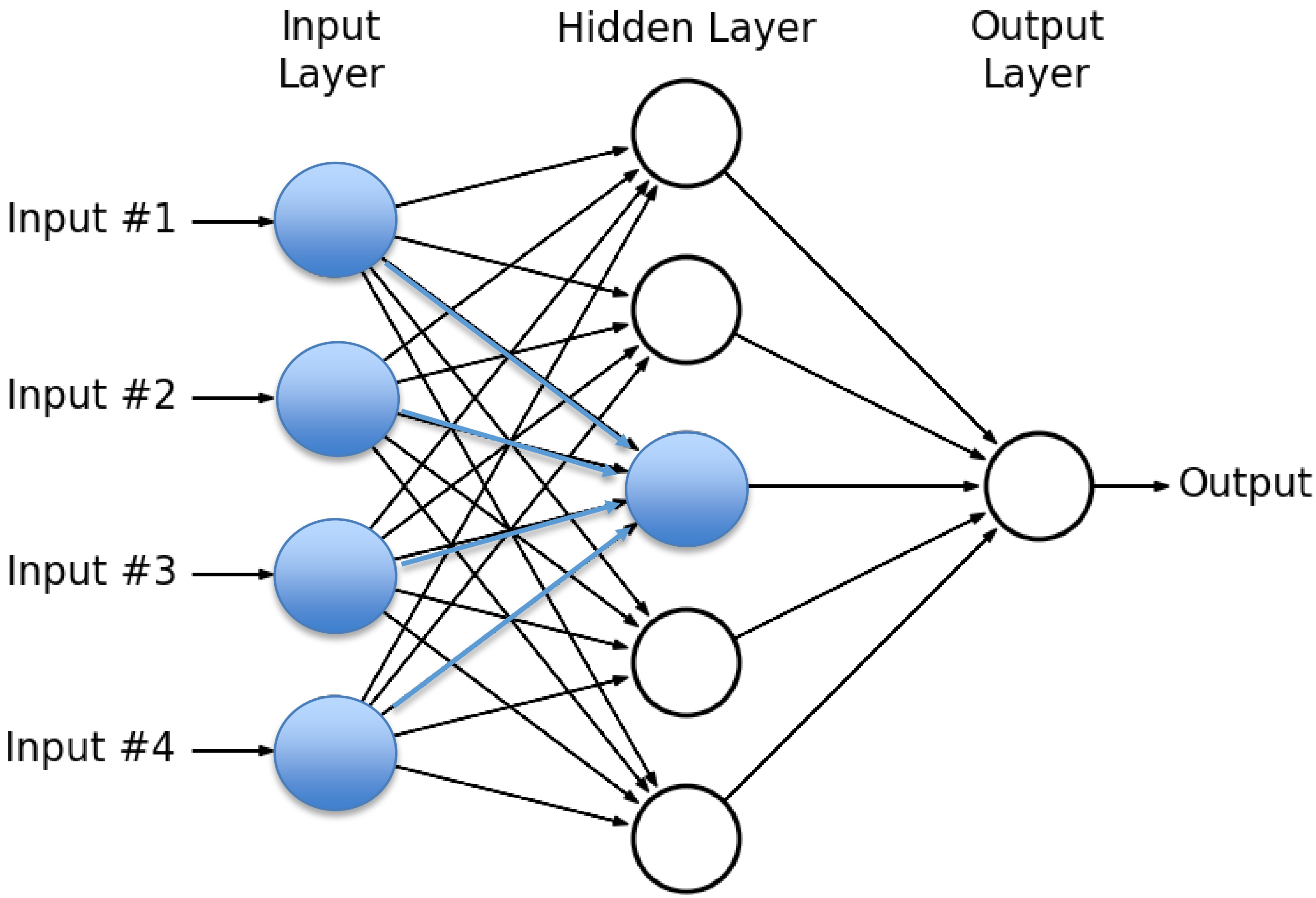
Part I: Deep Learning 101

70 Years Ago



First Neural Network: McCullogh & Pitts, 1943

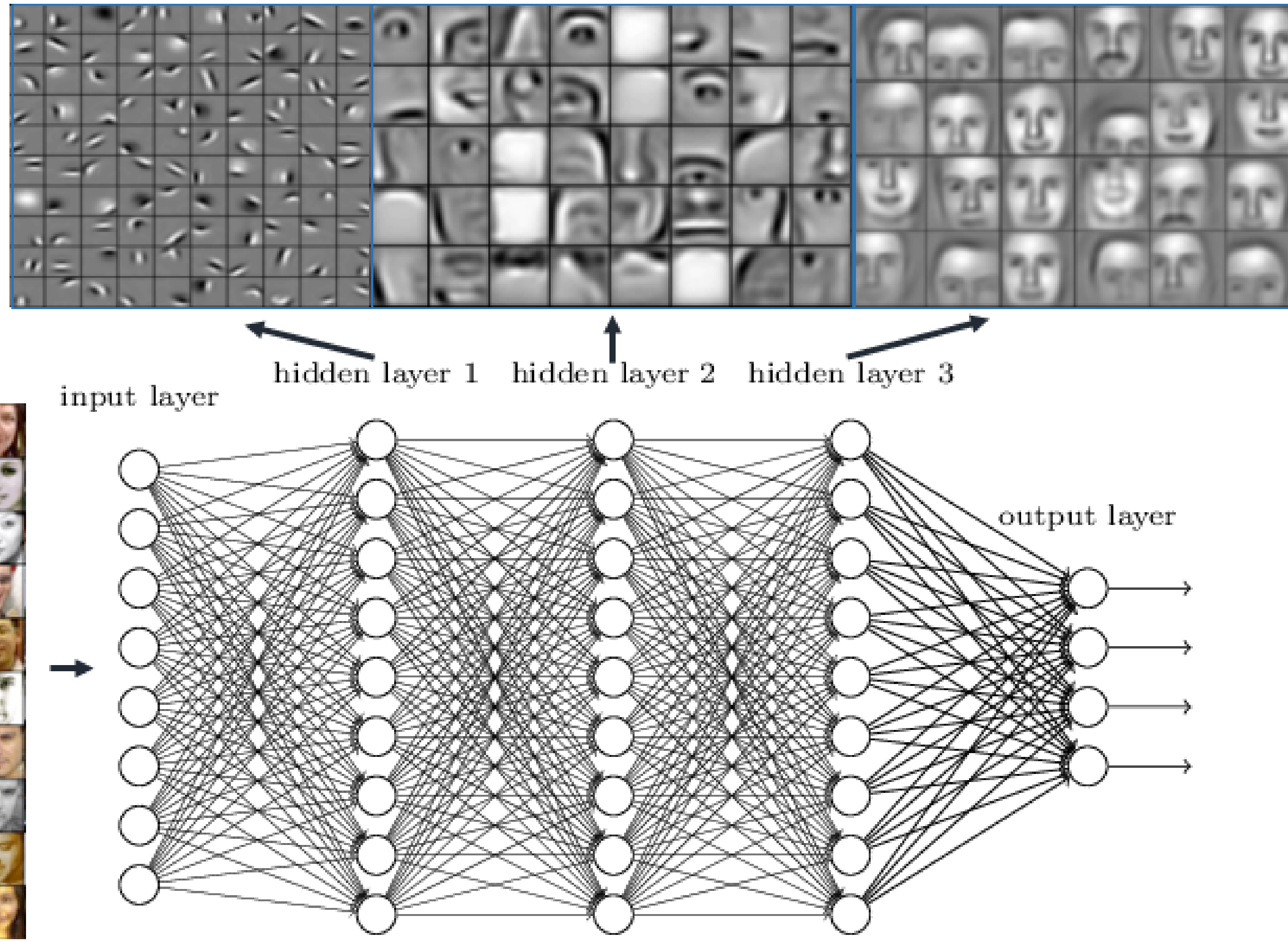
50 Years Ago



Input Layer
Hidden Layer
Output Layer
Input #1
Input #2
Input #3
Input #4
Output

5 Years Ago

Deep neural networks learn hierarchical feature representations



1 Year Ago: Resnets

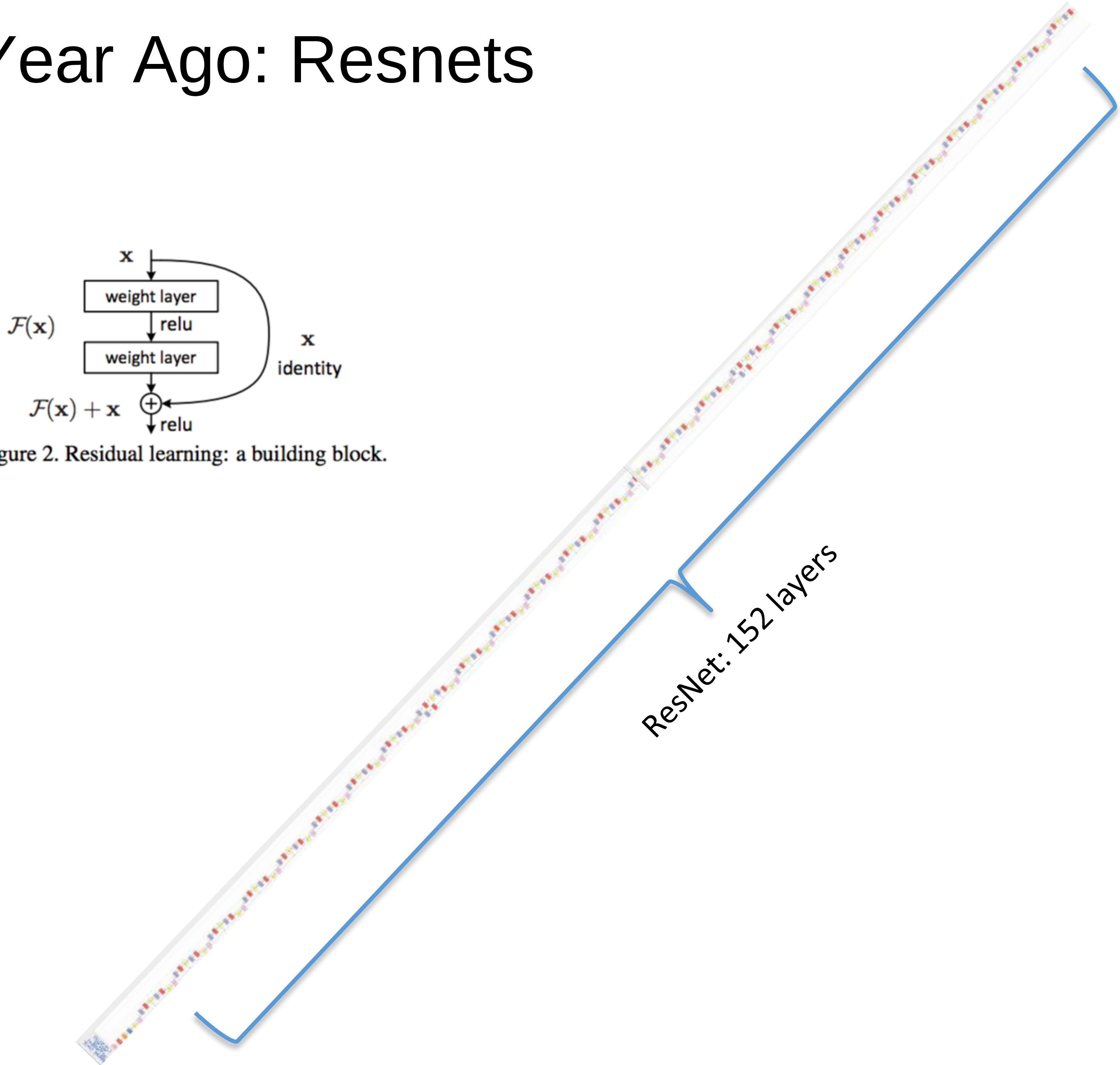
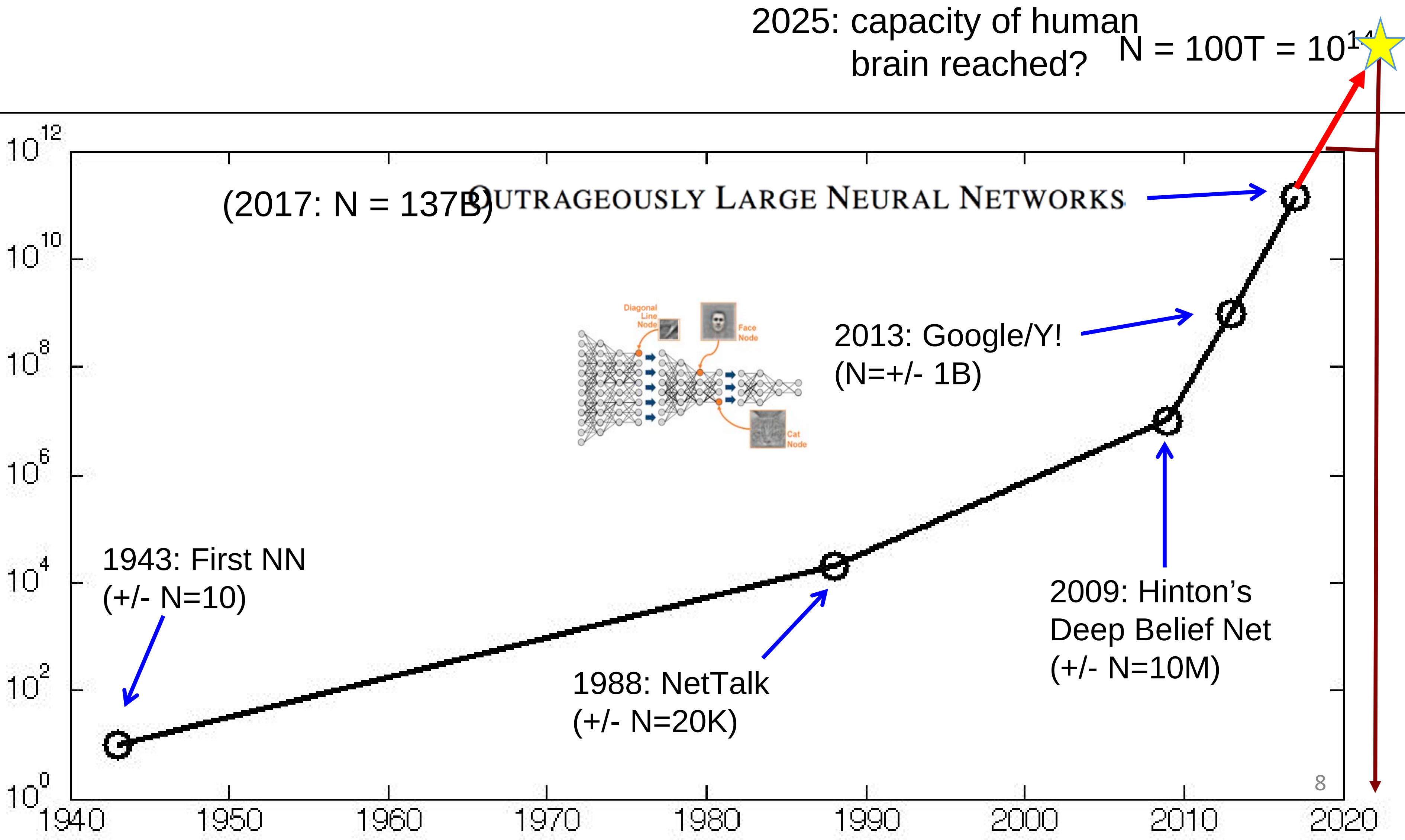
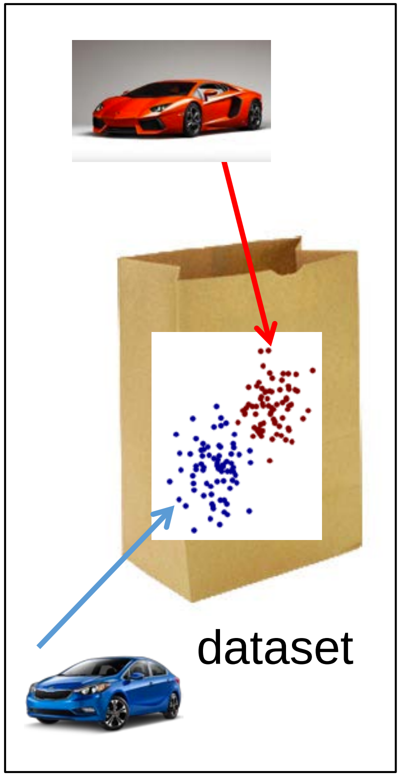


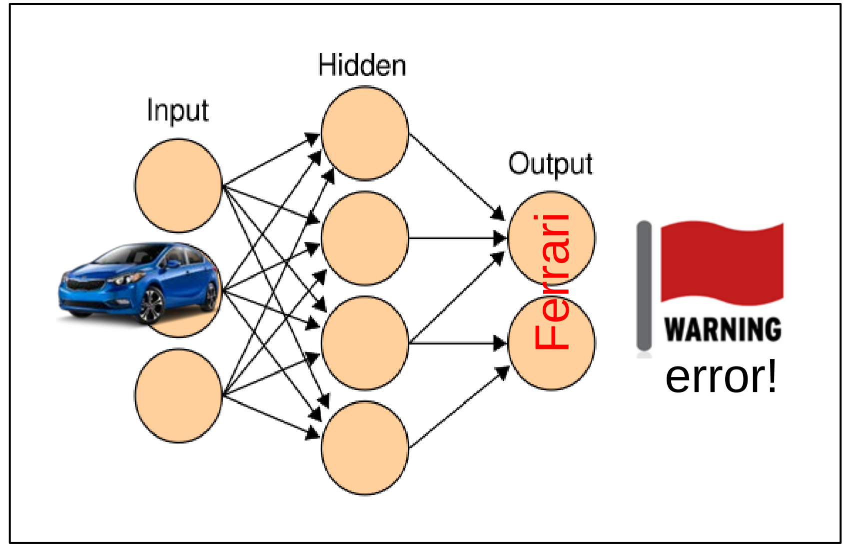
Figure 2. Residual learning: a building block.

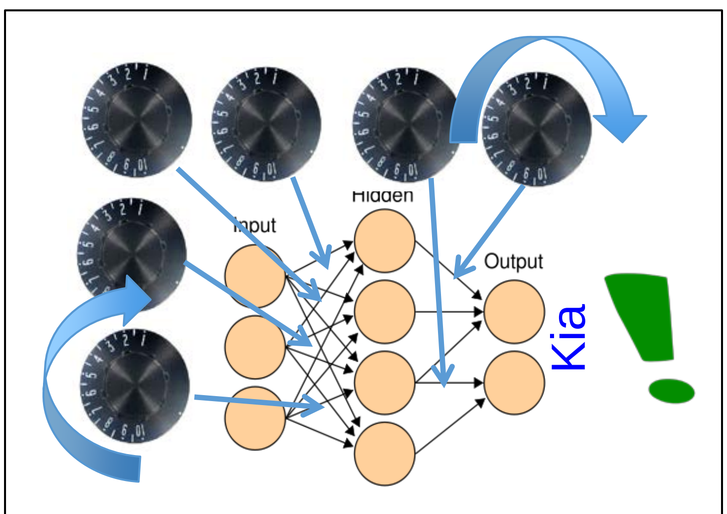
Explosive Growth Neural Network Capacity



How to Train a Computer

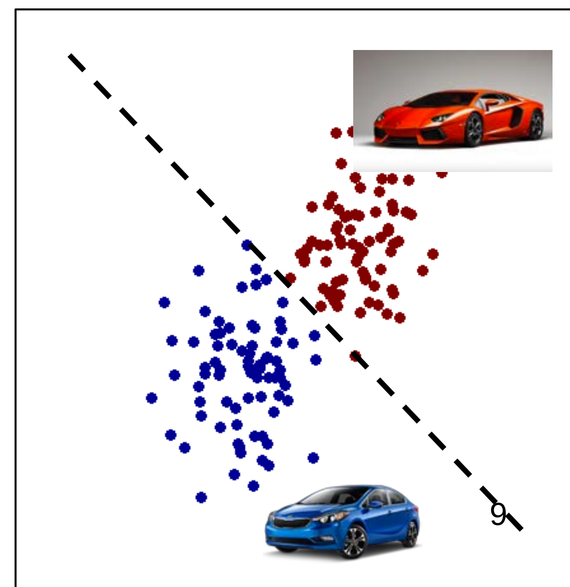






REPEAT

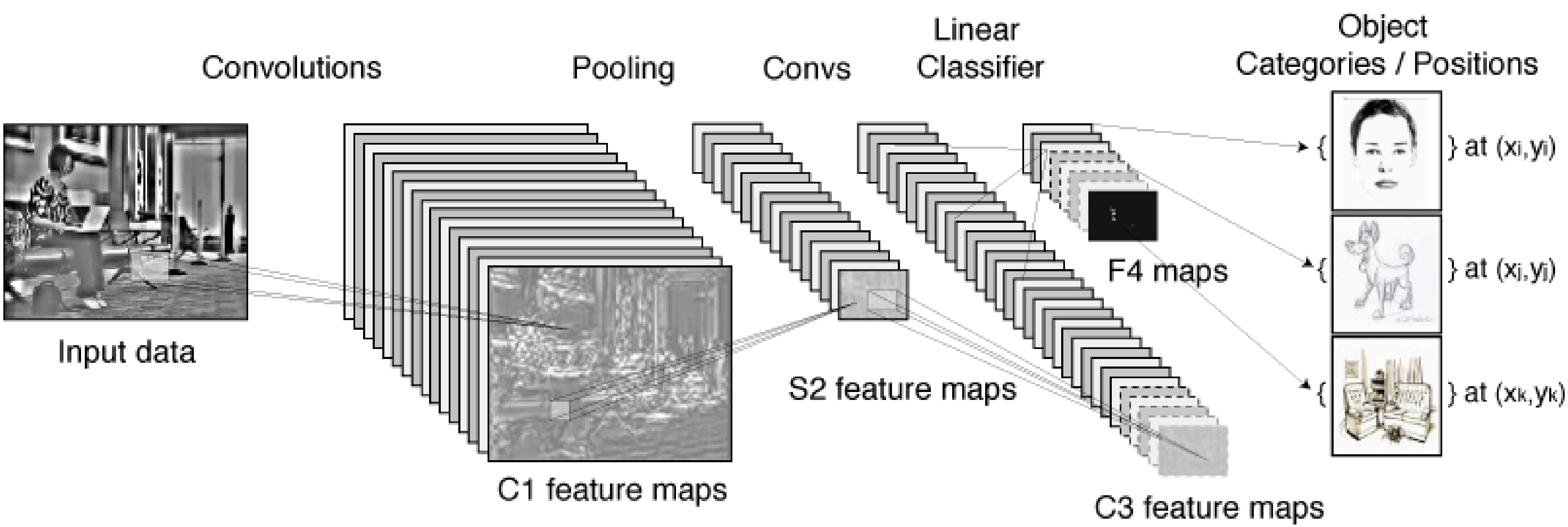
(keep turning the knobs until there are no more errors)



Deep Convolutional Networks

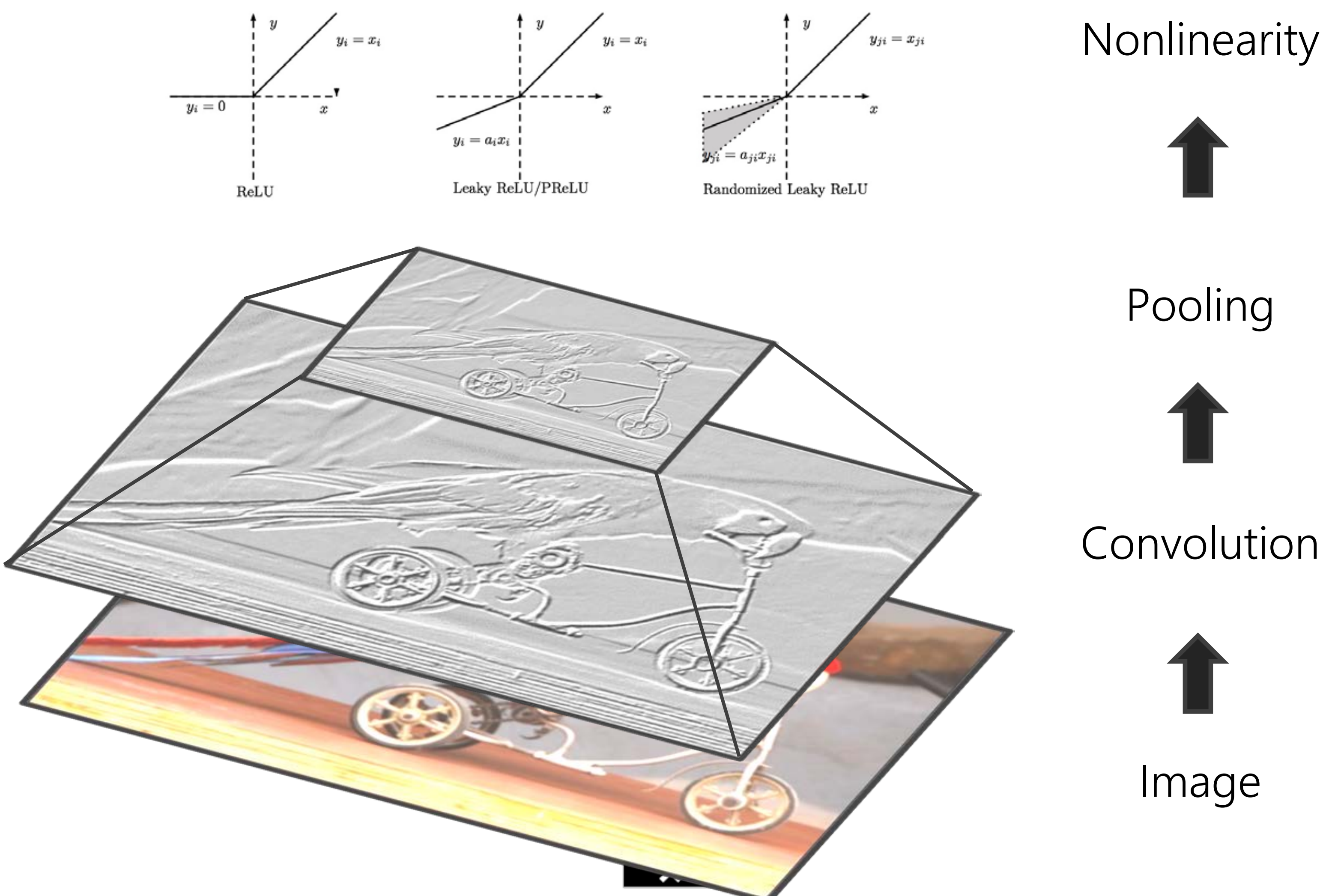
- Input dimensions have "topology”:
  (1D, speech, 2D image, 3D MRI, 2+1D video, 4D fMRI)

Forward: Filter, subsample, filter, nonlinearity, subsample, …., classify



Backward: backpropagation (propagate error signal backward)

Convolutional Network (slide borrowed from Li Deng)



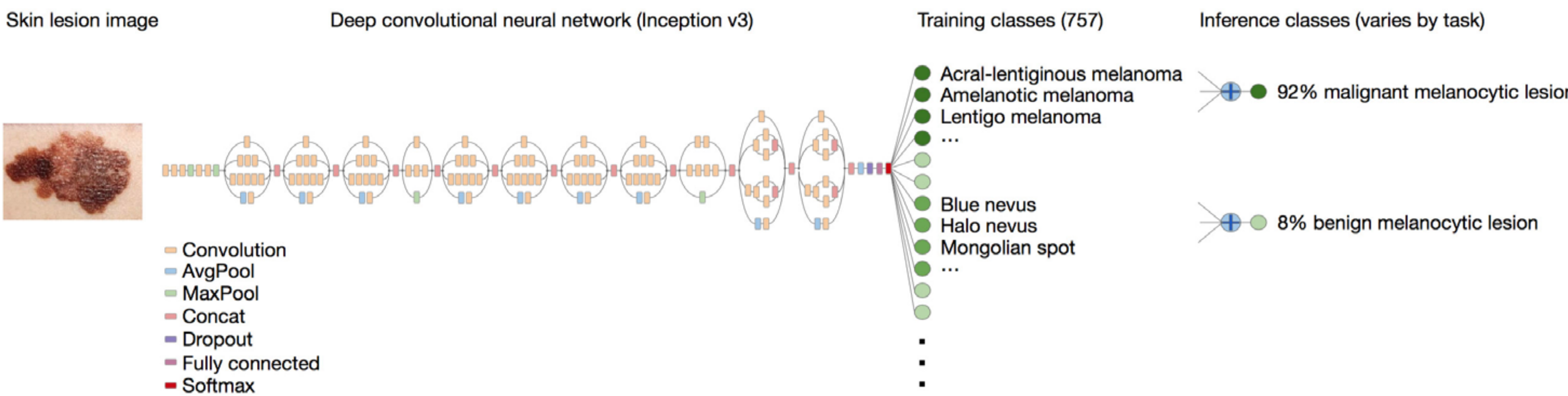
CNN in Action

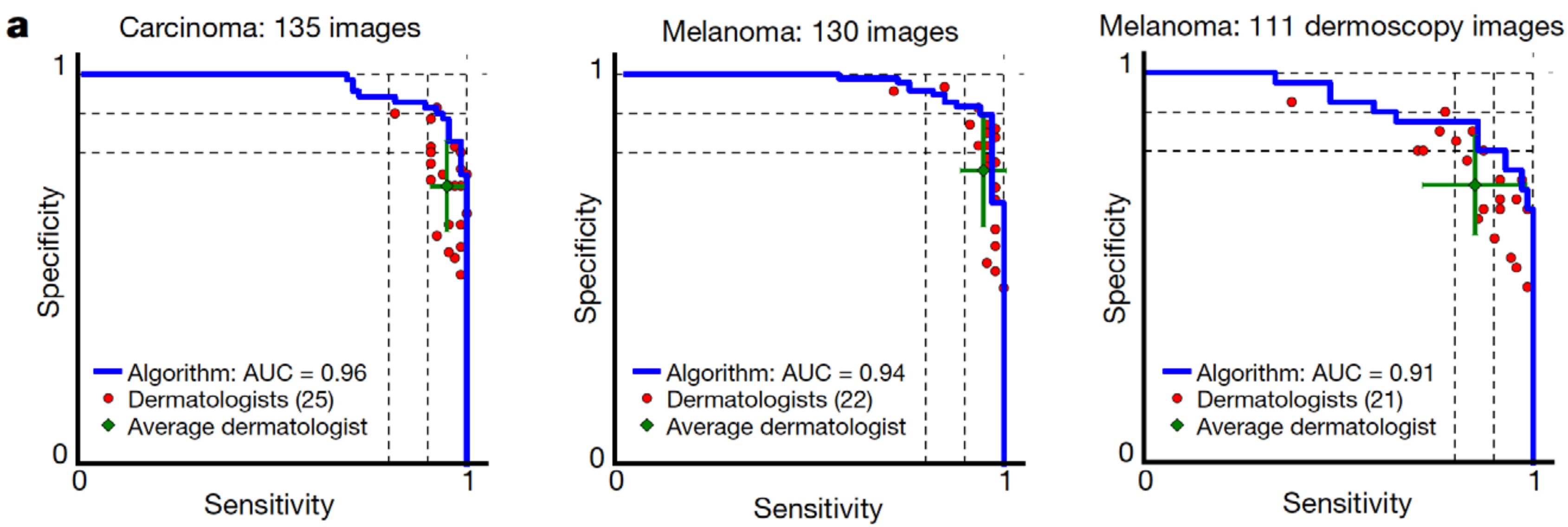
(Andreiy Karpathy's blog)

Example: Dermatology

Dermatologist-level classification of skin cancer with deep neural networks

Andre Esteva[1*], Brett Kuprel[1*], Roberto A. Novoa[2,3], Justin Ko[2], Susan M. Swetter[2,4], Helen M. Blau[5] & Sebastian Thrun[6]



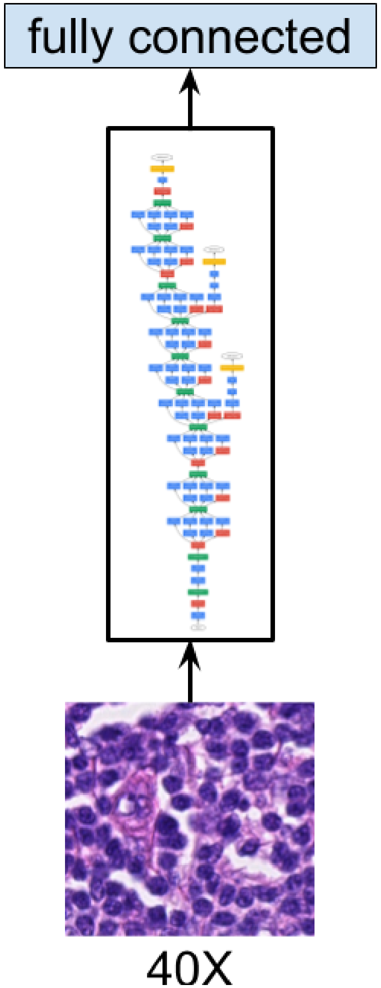

a
Carcinoma: 135 images
Melanoma: 130 images
Melanoma: 111 dermoscopy images
Specificity
Sensitivity
1
0
0
1
Algorithm: AUC = 0.96
Dermatologists (25)
Average dermatologist
Algorithm: AUC = 0.94
Dermatologists (22)
Average dermatologist
Algorithm: AUC = 0.91
Dermatologists (21)
Average dermatologist

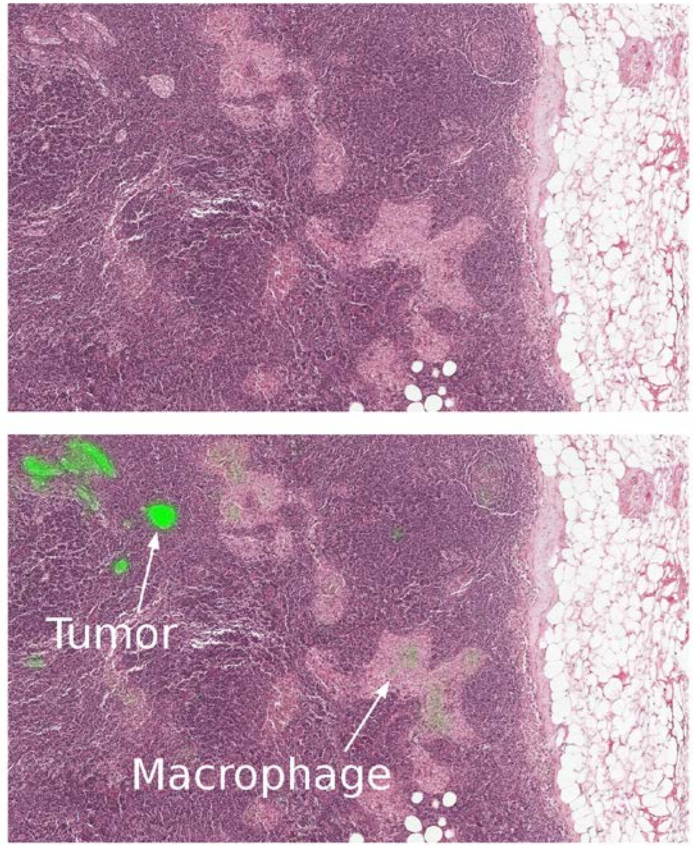
Example: Pathology

NOS | Nieuws | Sport | Uitzendingen | TELETEKST | AEX | 0 km

Computer kan kanker beter herkennen dan patholoog

VRIJDAG, 17:07 BINNENLAND, TECH





Beter dan de patholoog

Datzelfde principe heeft Google nu toegepast op de data van het Radboud. Het algoritme werd geprogrammeerd om kankercellen te vinden op de foto's en vervolgens aan het werk gezet. Volgens de onderzoekers haalde het algoritme een score van 89 procent, terwijl een patholoog gemiddeld 73 procent haalt op dezelfde foto's.

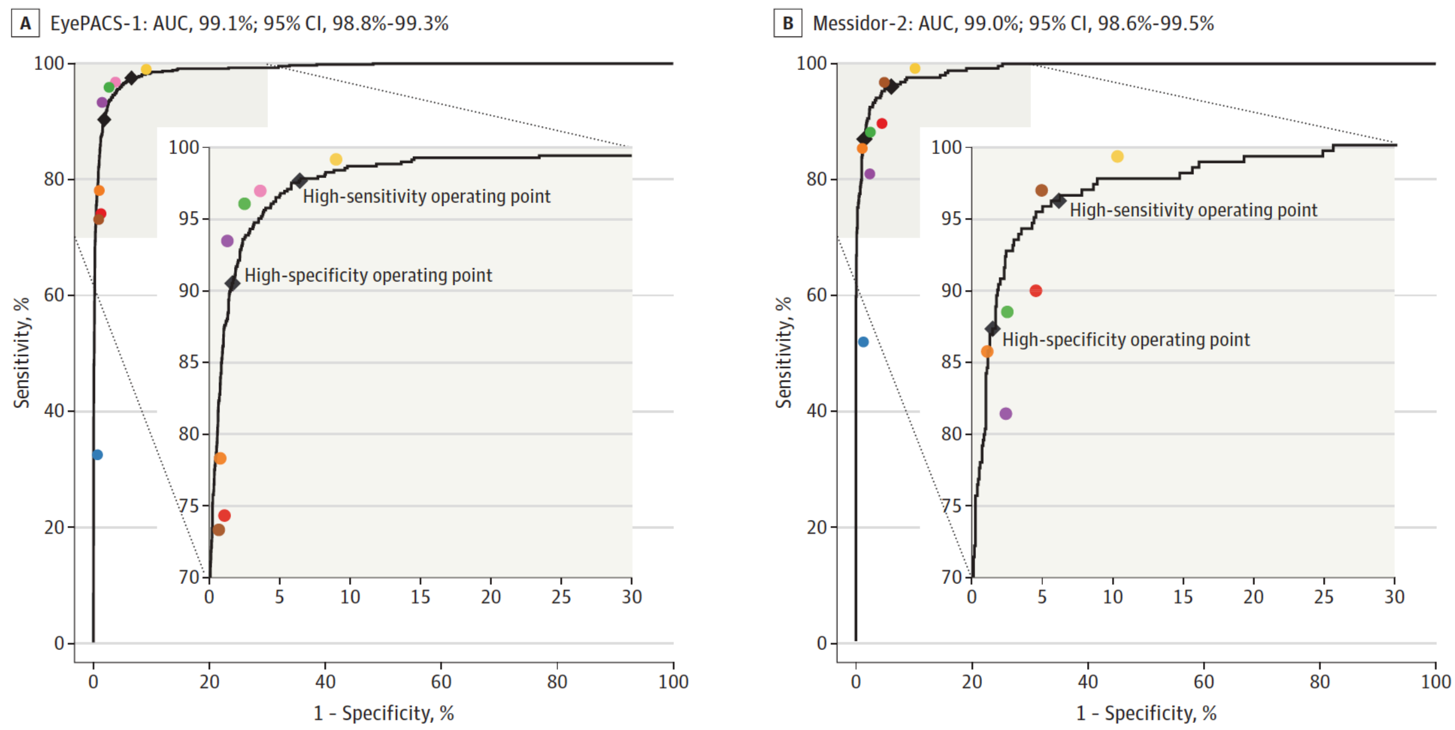
Example: Retinopathy

JAMA | Original Investigation | INNOVATIONS IN HEALTH CARE DELIVERY

Development and Validation of a Deep Learning Algorithm for Detection of Diabetic Retinopathy in Retinal Fundus Photographs

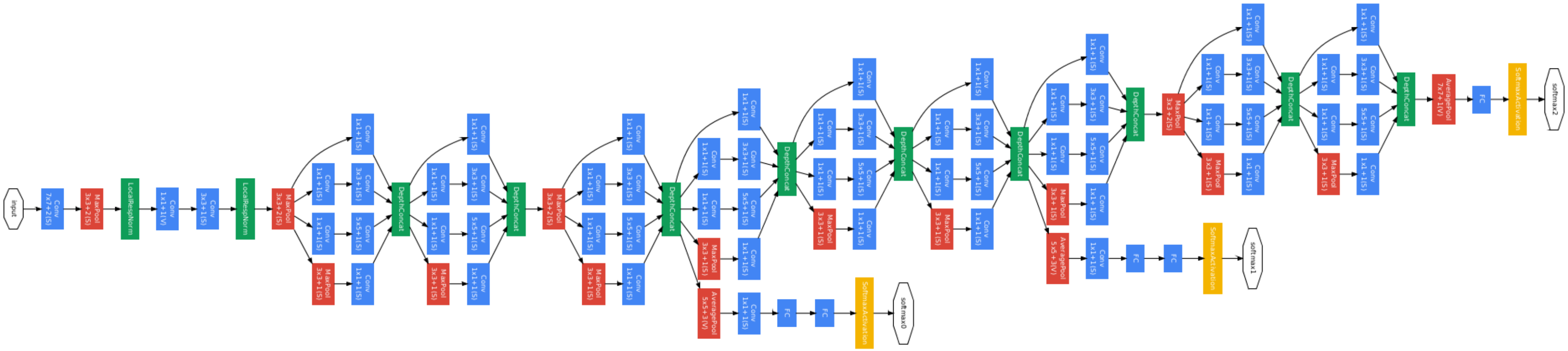
Varun Gulshan, PhD; Lily Peng, MD, PhD; Marc Coram, PhD; Martin C. Stumpe, PhD; Derek Wu, BS; Arunachalam Narayanaswamy, PhD; Subhashini Venugopalan, MS; Kasumi Widner, MS; Tom Madams, MEng; Jorge Cuadros, OD, PhD; Ramasamy Kim, OD, DNB; Rajiv Raman, MS, DNB; Philip C. Nelson, BS; Jessica L. Mega, MD, MPH; Dale R. Webster, PhD

Figure 2. Validation Set Performance for Referable Diabetic Retinopathy

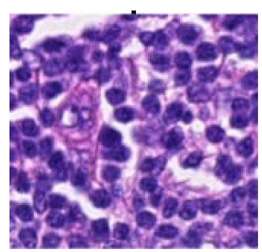
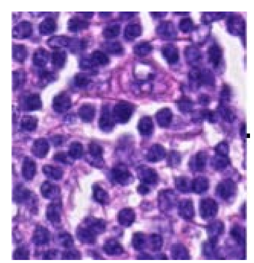


Universiteit van Amsterdam

What do these problems have in common?



1) It's the same CNN in all cases: Inception-v3

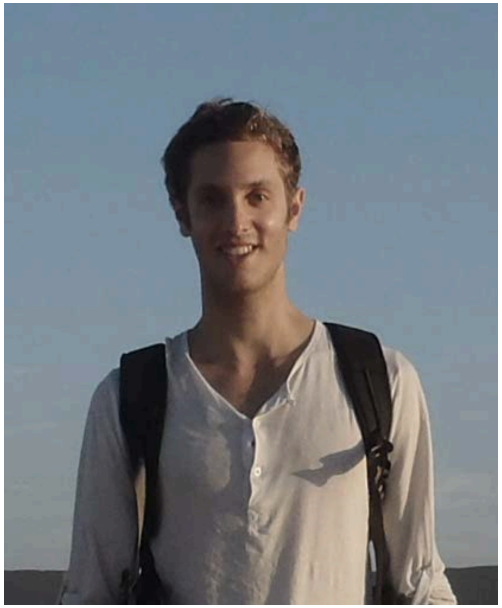
1) Object identity is translations translation, rotation, mirror invariant

Part II: Symmetries

Symmetries

- How can we improve CNNs by exploiting symmetries better ?

(Escher)



Taco Cohen

Equivariance

$$R_{\mu\nu} - \frac{1}{2} R g_{\mu\nu} = 8\pi G T_{\mu\nu}$$

Symmetry in Deep Learning

What makes CNNs so effective?

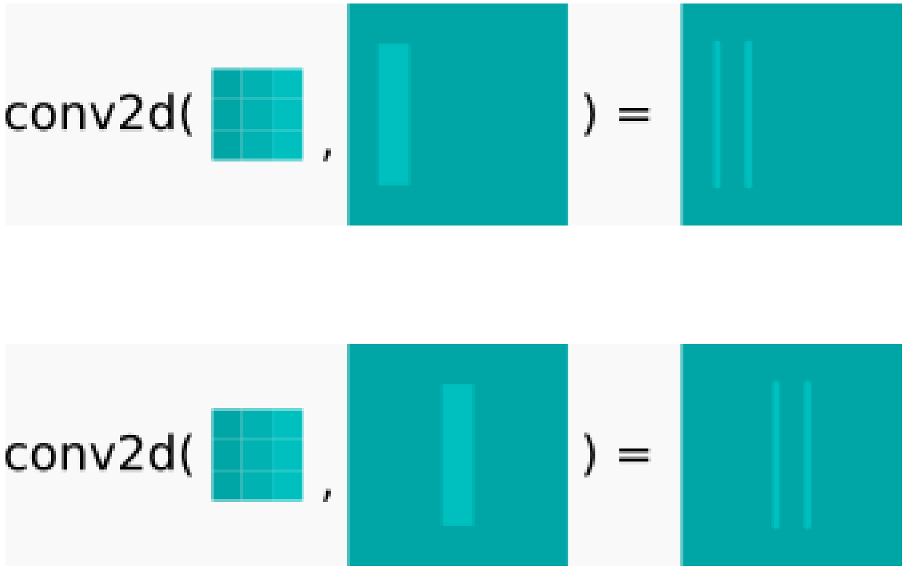
- Weight sharing: exploits translation symmetry
- Depth: exploits equivariance

Network design principle:

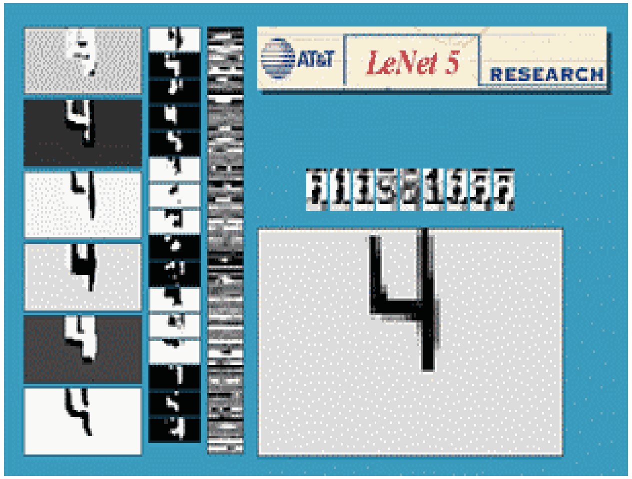
Equivariance to symmetry transformations

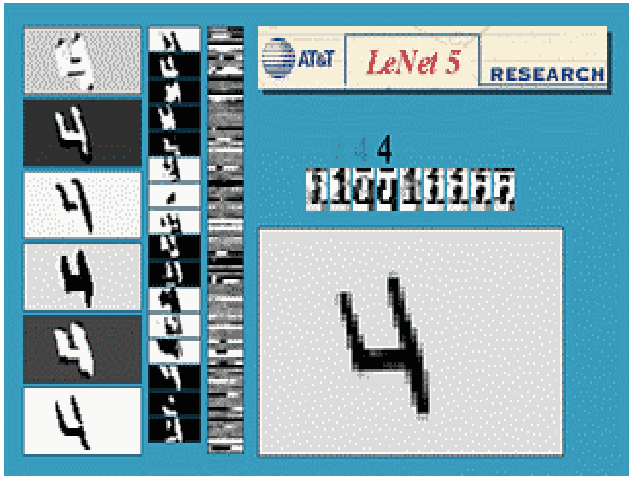
(Picasso effect:
why we do not want to
use invariant features)

Equivariance



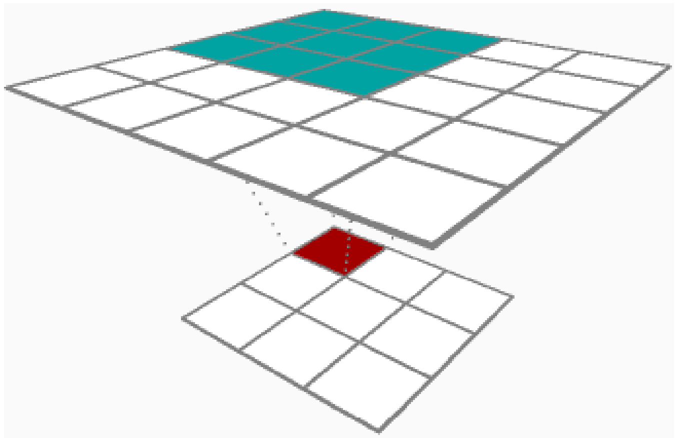
Equivariance

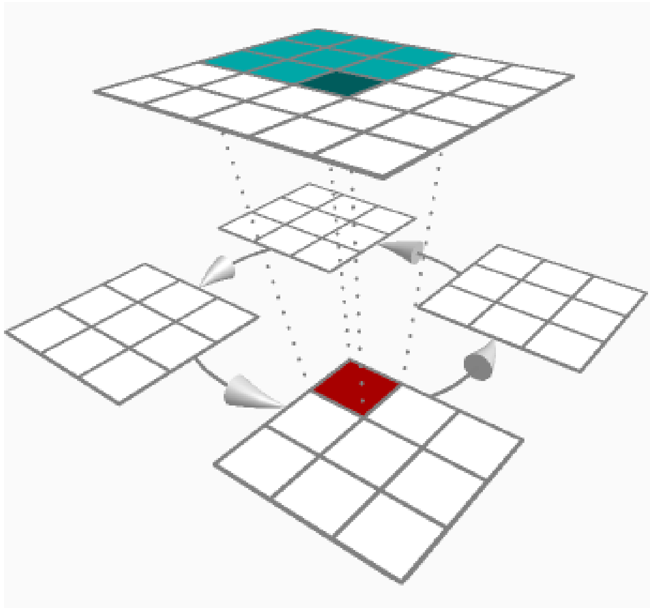




Source: http://yann.lecun.com/exdb/lenet/index.html

Conv vs G-Conv





T.S. Cohen & M. Welling, *Group Equivariant Convolutional Networks.* ICML 2016
J. Peters & T. Cohen, Data-Efficient Deep Learning with G-CNNs, Scyfer Blog, 2016
Sander Dieleman, Jeffrey de Fauw, Koray Kavukcuoglu, Exploiting Cyclic Symmetry in Convolutional Neural Networks, ICML2016

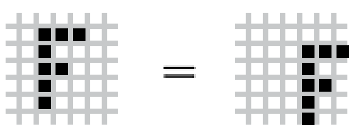
Conv vs G-Conv

Planar Convolution

"translate filter and compute inner product"

Translation

$$T_s f(x) = f(x - s)$$

$T_{(2,1)}$ 

Z^2-Convolution

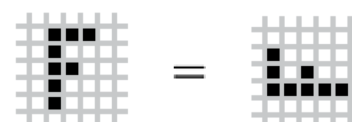
$$[f \star \psi](s) = \sum_{x \in \mathbb{Z}^2} \sum_{k=1}^{K} f_k(x) [T_s \psi]_k(x)$$

Group Convolution

"transform filter and compute inner product"

Transformation

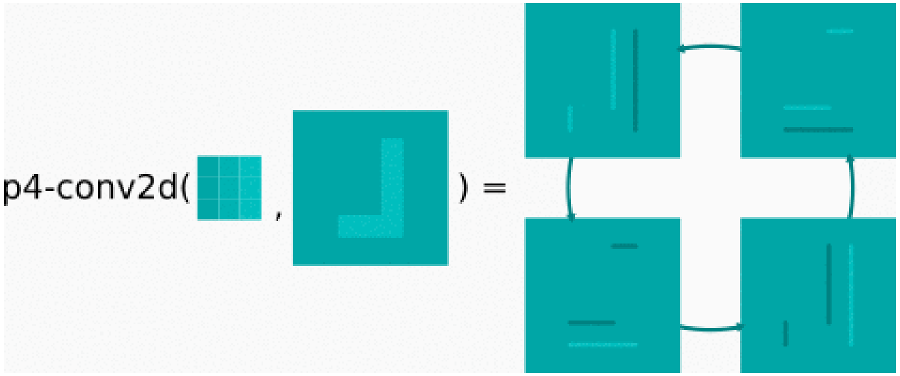
$$T_r f(x) = f(r^{-1} x)$$

T_r 

G-Convolution

$$[f \star \psi](g) = \sum_{x \in \mathbb{Z}^2} \sum_{k=1}^{K} f_k(x) [T_g \psi]_k(x)$$

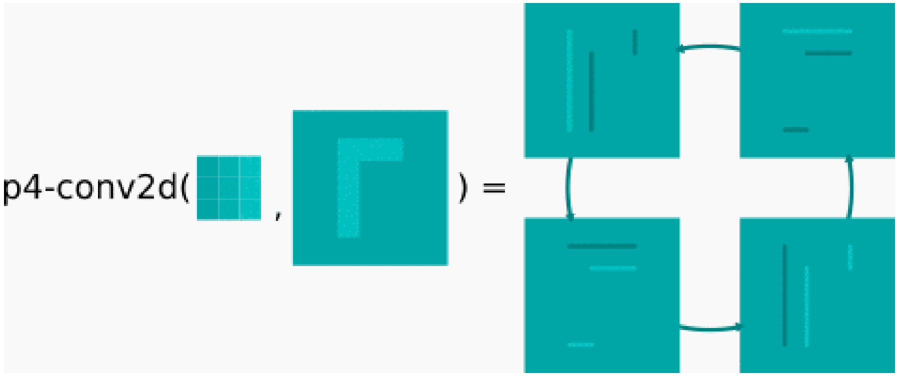
Equivariance of G-Convs



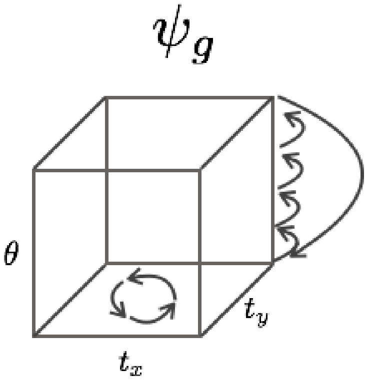
$$[T_g f] \star \psi = T_g [f \star \psi]$$

T.S. Cohen & M. Welling,
Group Equivariant Convolutional Networks.
ICML 2016

Equivariance of G-Convs

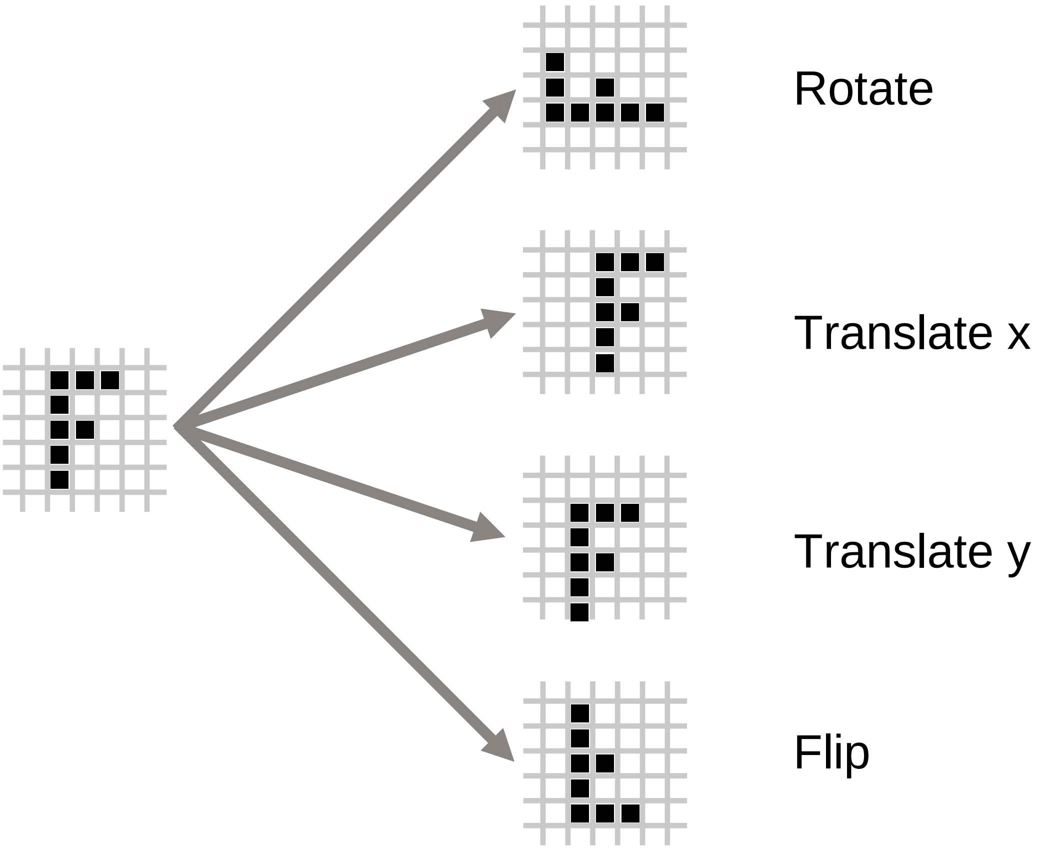


$$[T_g f] \star \psi = T_g [f \star \psi]$$

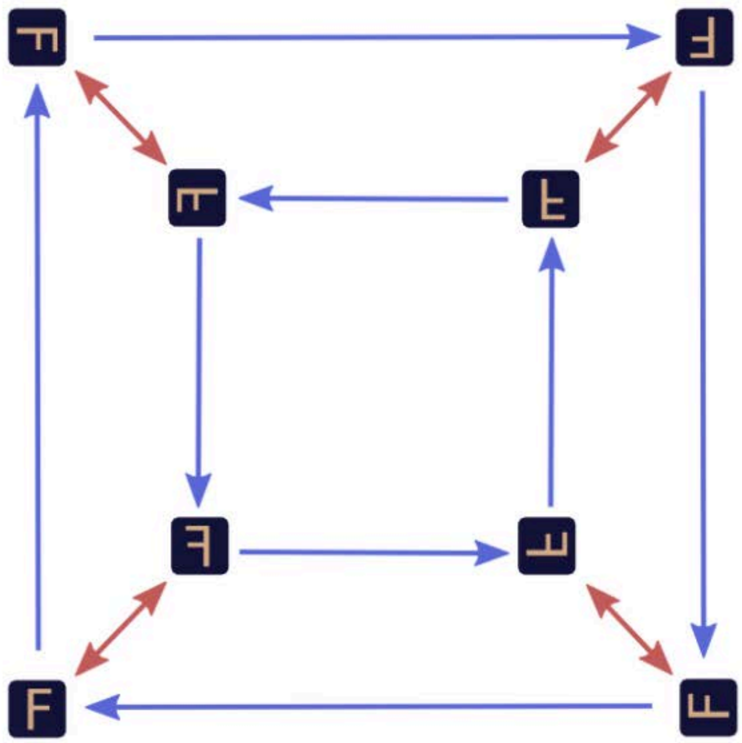


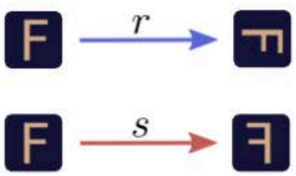
T.S. Cohen & M. Welling,
Group Equivariant Convolutional Networks.
ICML 2016

The Groups p4 & p4m



Cayley Diagrams

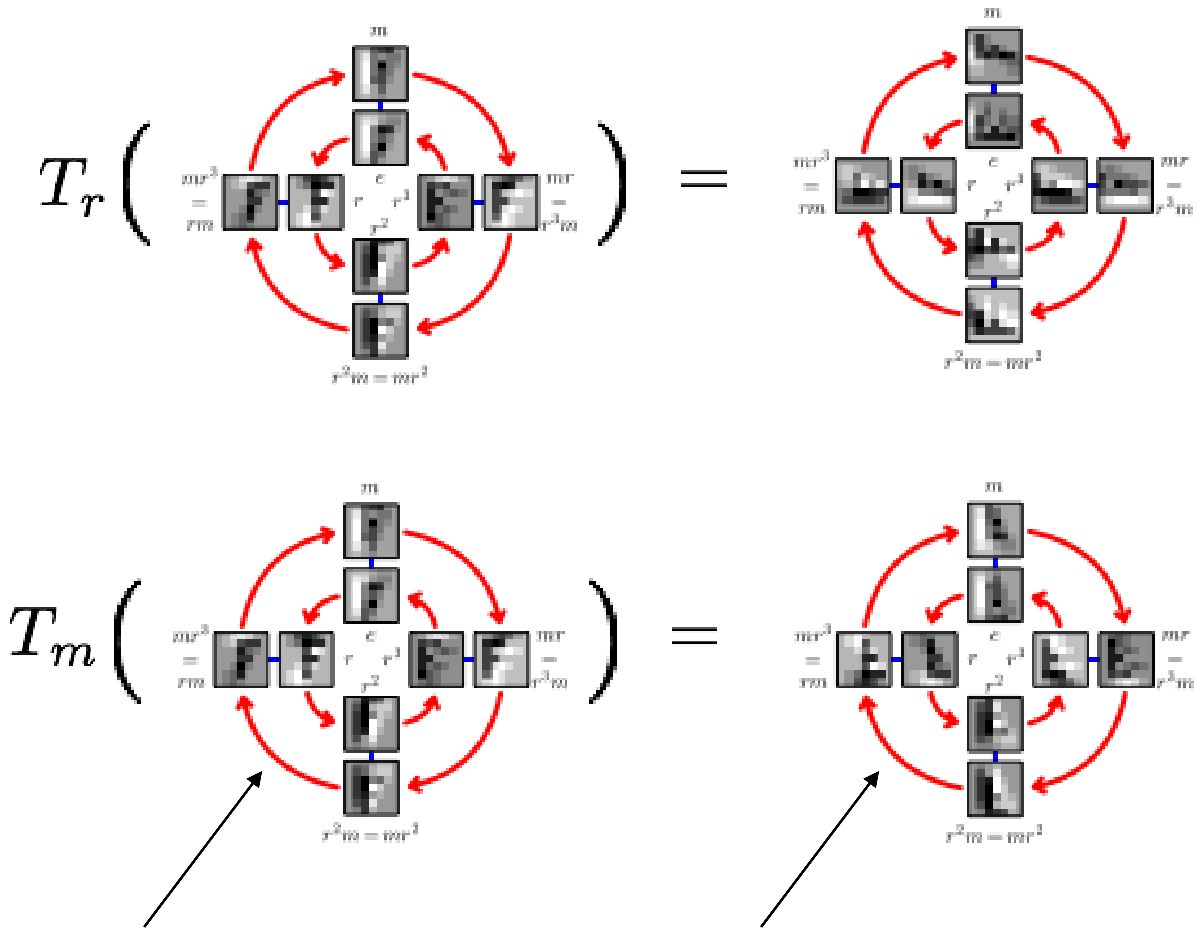




(from Olah’s blog)

Equivariance of G-Convs

$$[T_g f] \star \psi = T_g [f \star \psi]$$



(result of gconv on ”F”)

(The filter maps transform and permute)

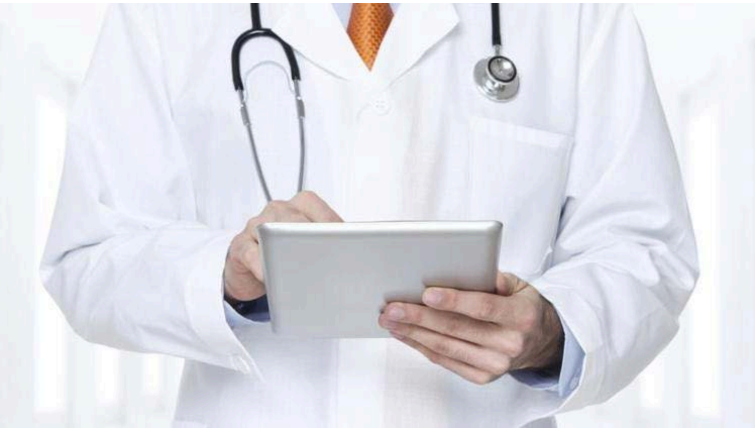
Some Results

Network	Group	CIFAR10	CIFAR10+
All-CNN	Z_2	9.44	8.86
	$p4$	8.84	7.67
	$p4m$	7.59	7.04
ResNet44	Z_2	9.45	5.61
	$p4m$	6.46	4.94

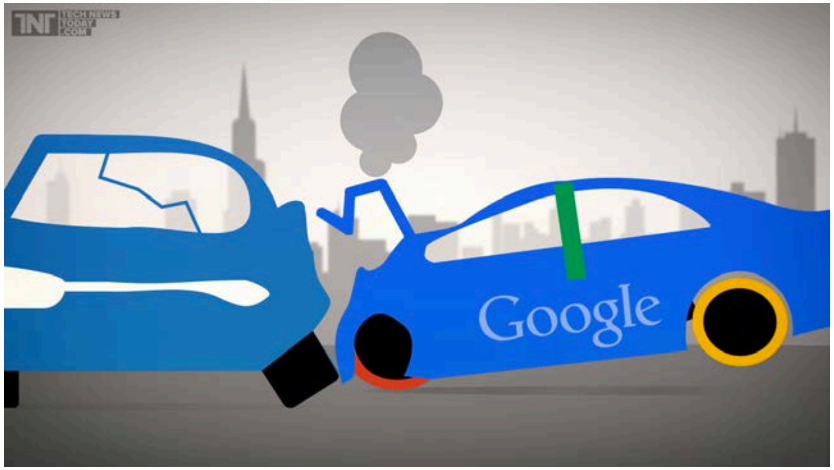
PART III: Bayesian Deep Learning

Reasons for Bayesian Deep Learning

- Automatic model selection / pruning
- Automatic regularization
- Realistic prediction uncertainty (important for decision making)



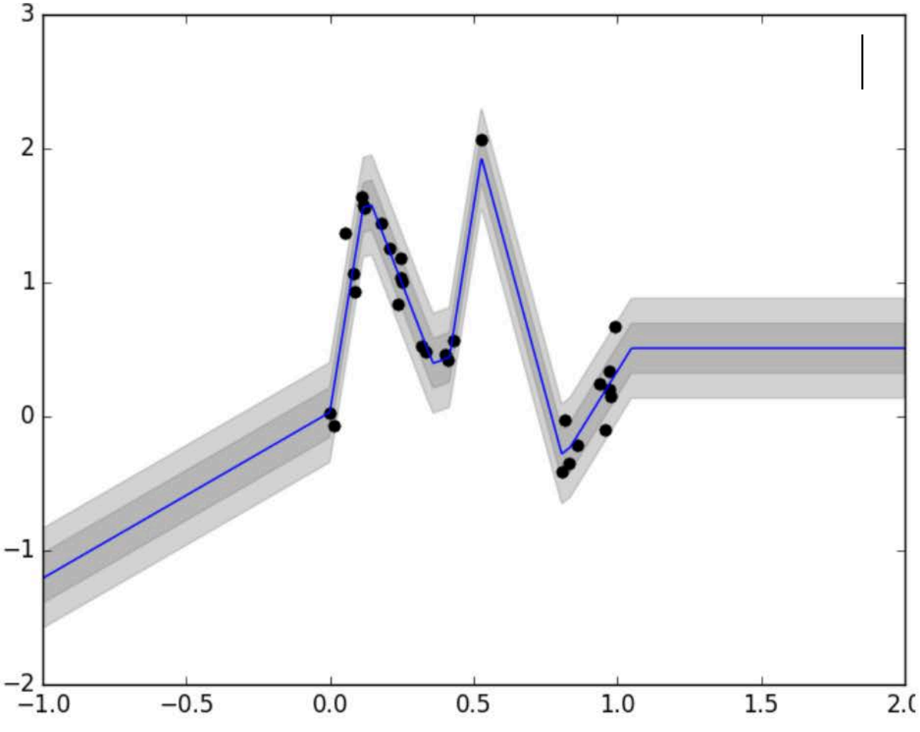
Computer Aided Diagnosis



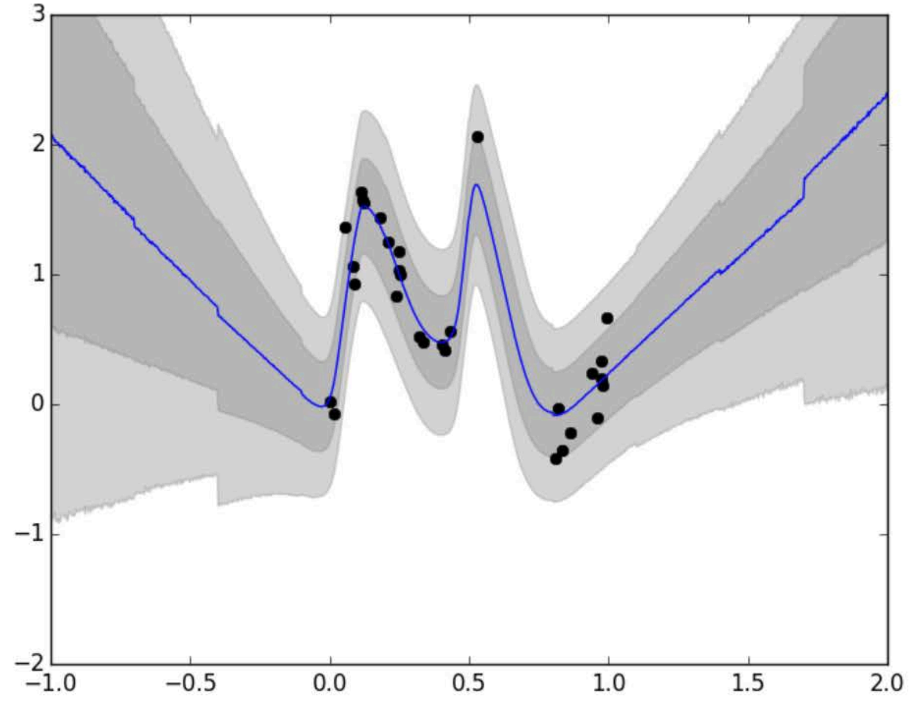
Autonomous Driving

Example

Increased uncertainty away from data



MAP estimate of parameters

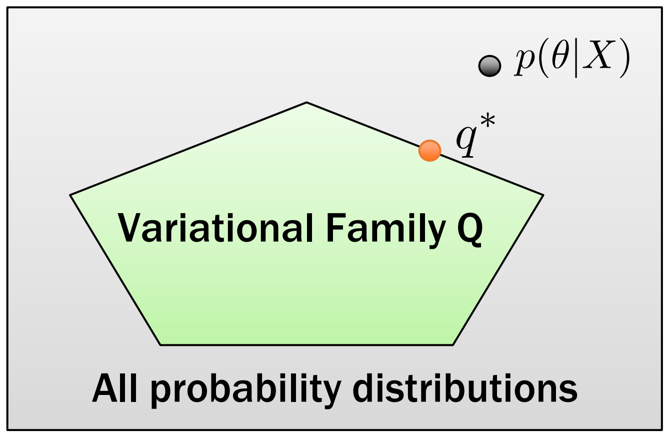


Bayesian estimate of parameters

Bayesian Variational Posterior Inference

Variational Inference

$$q^* = \min_{q \in Q} \mathrm{KL}[q(\theta) || p(\theta | X)]$$



- Deterministic
- Biased
- Local minima
- Easy to assess convergence

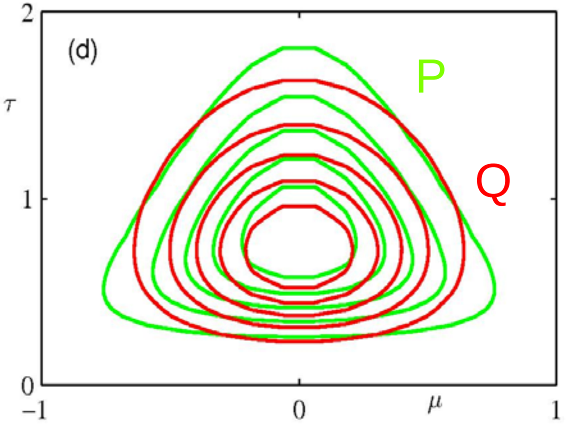
Deep Learning as Statistical Physics

Energy E　　　　Entropy H

$$-F(Q(\Theta)|X) = \int d\Theta\ Q(\Theta)\left[\log(P(X|\Theta)P(\Theta)) - \log Q(\Theta)\right]$$

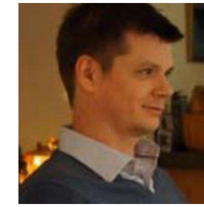
$$= \log P(X) - KL\left[Q(\Theta)||P(\Theta|X)\right]$$

$$\leq \log P(X)$$



(Bishop, Pattern Recognition and Machine Learning)

Sparsifying & Compressing CNNs

w/ Karen Ullrich and Ted Meeds

- DNNs are vastly overparameterized (e.g. distillation, Bucilua et al 2006).
- Interpret variational bound as coding cost for data (minimum description length)

$$\mathcal{L}(q, \mathbf{w}) = -\mathbb{E}_q\left[\log\left(\frac{p(\mathcal{D}|\mathbf{w})p(\mathbf{w})}{q}\right)\right] = \underbrace{\mathbb{E}_q\left[-\log p(\mathcal{D}|\mathbf{w})\right]}_{L^E} + \underbrace{\mathbf{KL}(q||p(\mathbf{w}))}_{L^C}$$

error loss ~N

complexity loss ~const.

Empirical Bayes

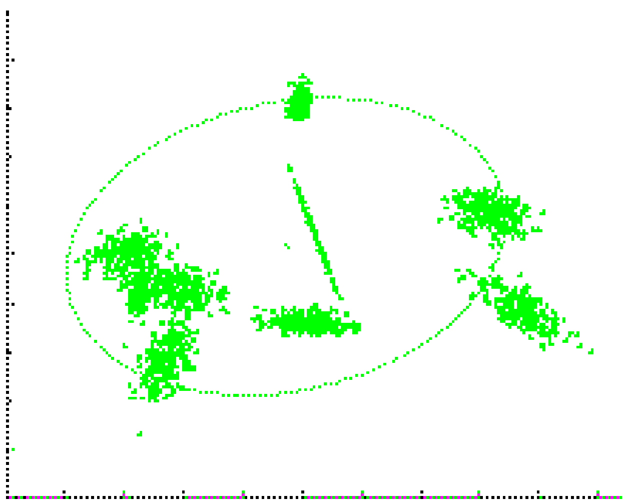
- Simple idea: learn a soft weight sharing prior (Nowlan & Hinton 1991, Gong et al 2014)

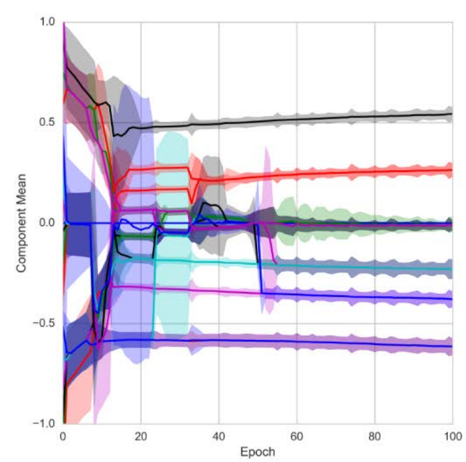
- Fit ”Mixture of Gaussians” prior to the distribution of weights (Nowlan & Hinton 1991).

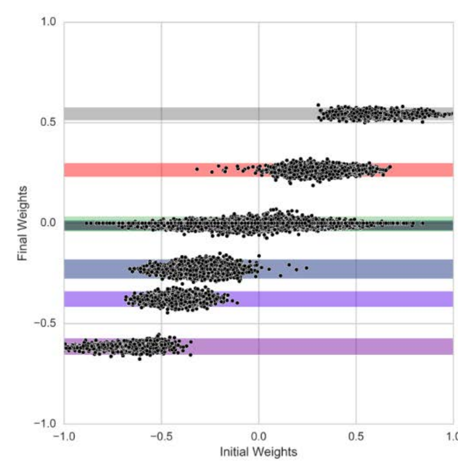
$$p(\mathbf{w}) = \prod_{i=1}^{I} \sum_{j=0}^{J} \pi_j \mathcal{N}(w_i | \mu_j, \sigma_j^2)$$

- Fixed component at w=0 encouraged to be very large (large π_0).

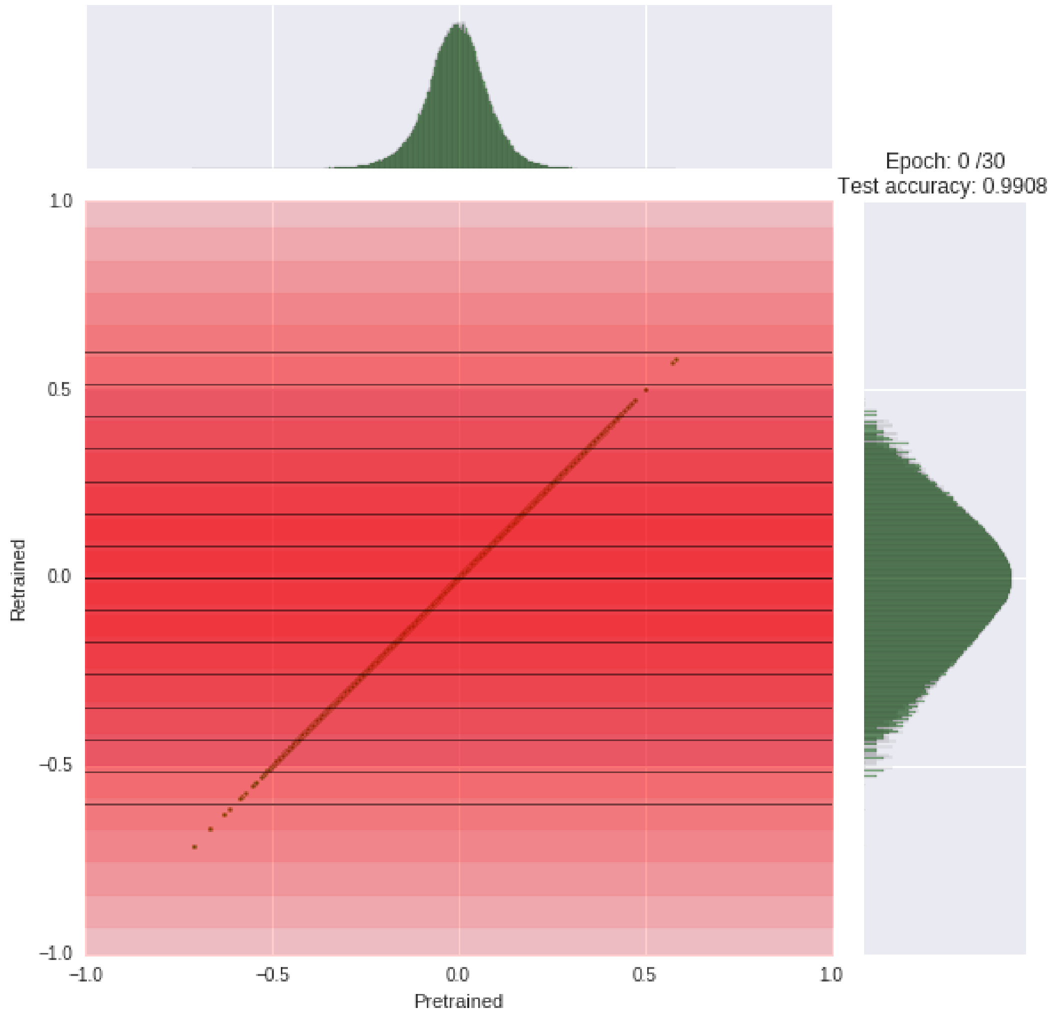
- When training likelihood and prior jointly, the weights cluster.







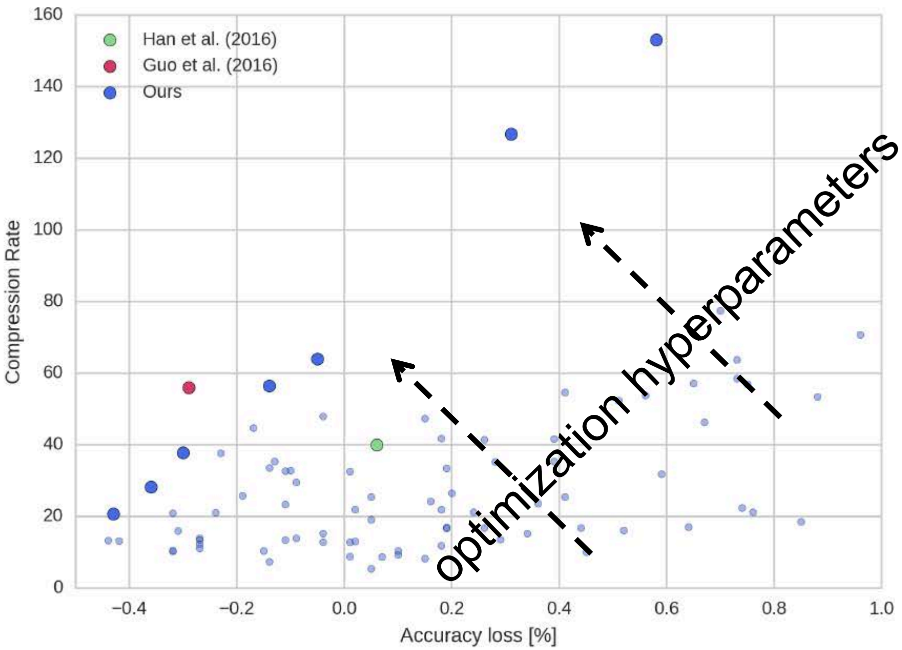
Clustering and Sparsification of the Network Weights



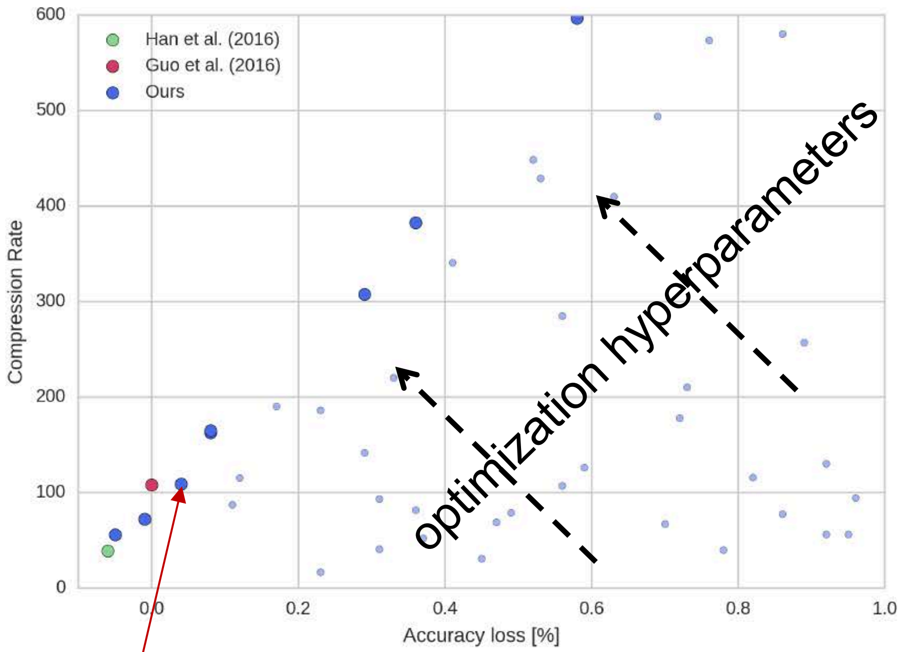
Some Results

Encode cluster means
Encode for each weight to which cluster it belongs

LeNet-300-100

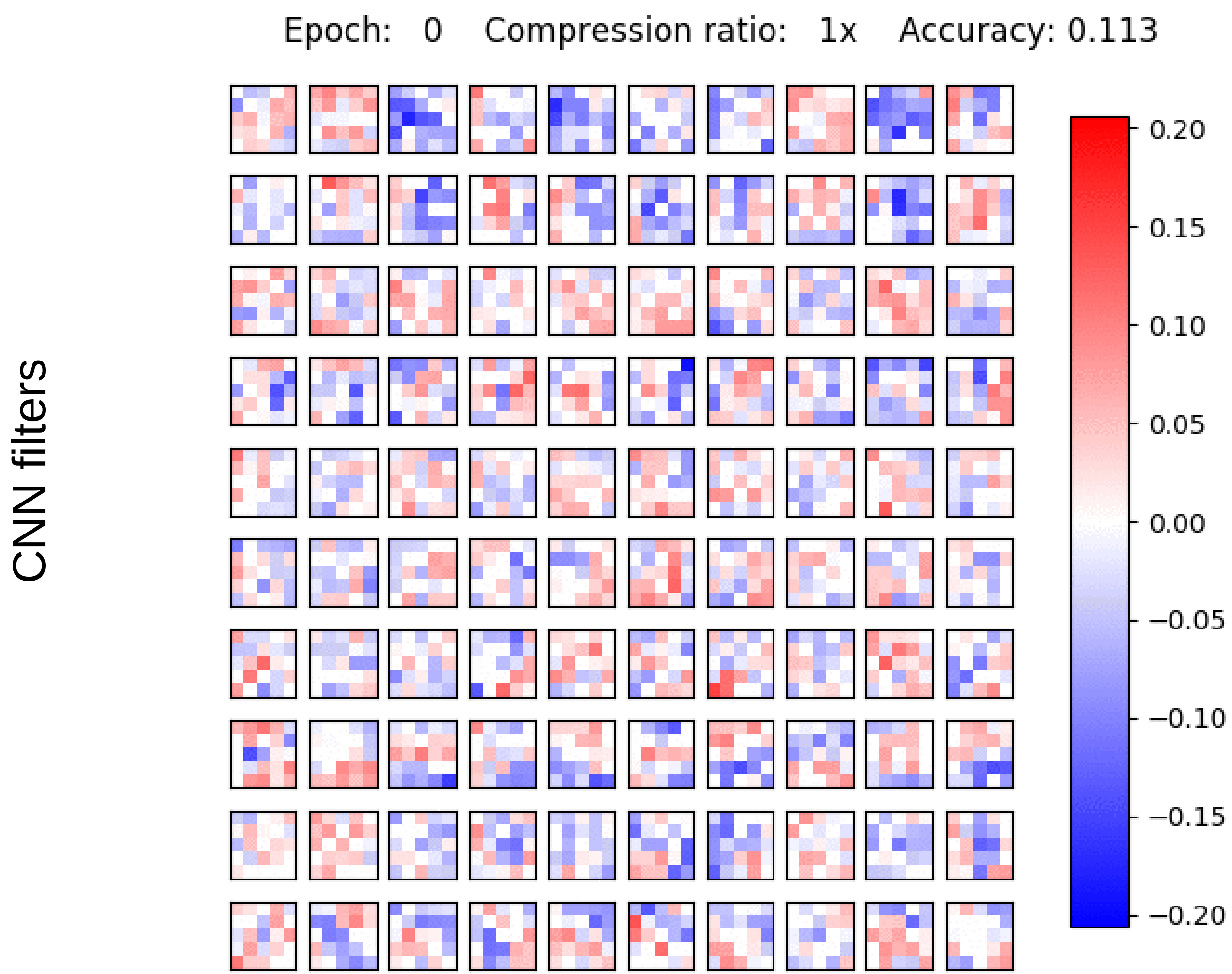


LeNet-5-Caffe



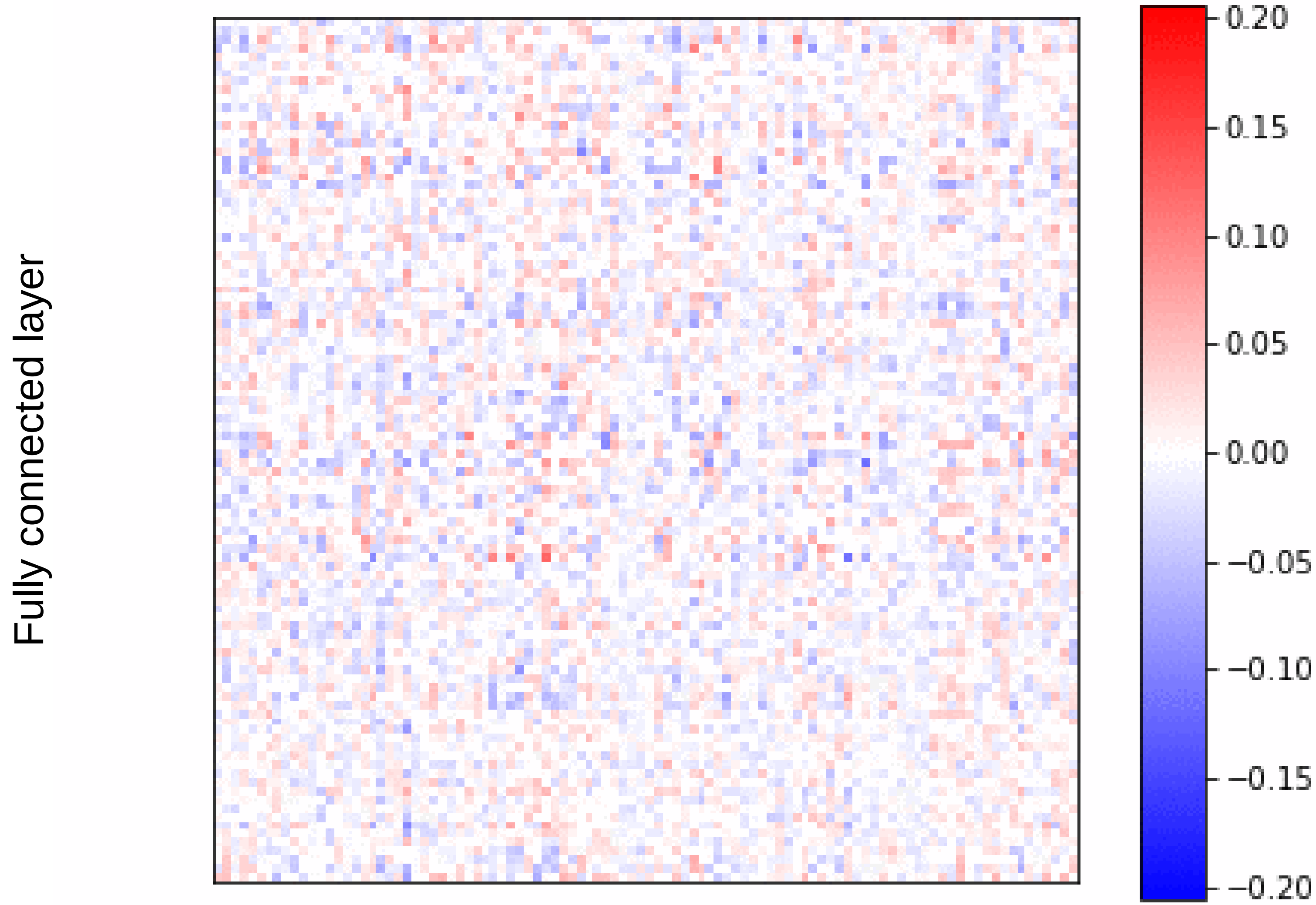
100 fold compression with almost no loss in accuracy.

Variational Dropout



Animation: Molchanov, D., Ashukha, A. and Vetrov, D.

Epoch: 0 Compression ratio: 1x Accuracy: 0.113



Animation: Molchanov, D., Ashukha, A. and Vetrov, D.

Preliminary Results

(Louizos, Ullrich, Molchanov, Vetrov, Welling 2017, unpublished)

Network & size	Method	Pruned architecture	Bit-precision
LeNet-300-100	Sparse VD	512-114-72	8-11-14
784-300-100	GNJ	278-98-13	8-9-14
	GHS	311-86-14	13-11-10
LeNet-5-Caffe	Sparse VD	14-19-242-131	13-10-8-12
	GD	7-13-208-16	-
20-50-800-500	GL	3-12-192-500	-
	GNJ	8-13-88-13	18-10-7-9
	GHS	5-10-76-16	10-10–14-13
VGG	GNJ	63-64-128-128-245-155-63- -26-24-20-14-12-11-11-15	10-10-10-10-8-8-8- -5-5-5-5-5-6-7-11
(2× 64)-(2× 128)- -(3×256)-(8× 512)	GHS	51-62-125-128-228-129-38- -13-9-6-5-6-6-6-20	11-12-9-14-10-8-5- -5-6-6-6-8-11-17-10

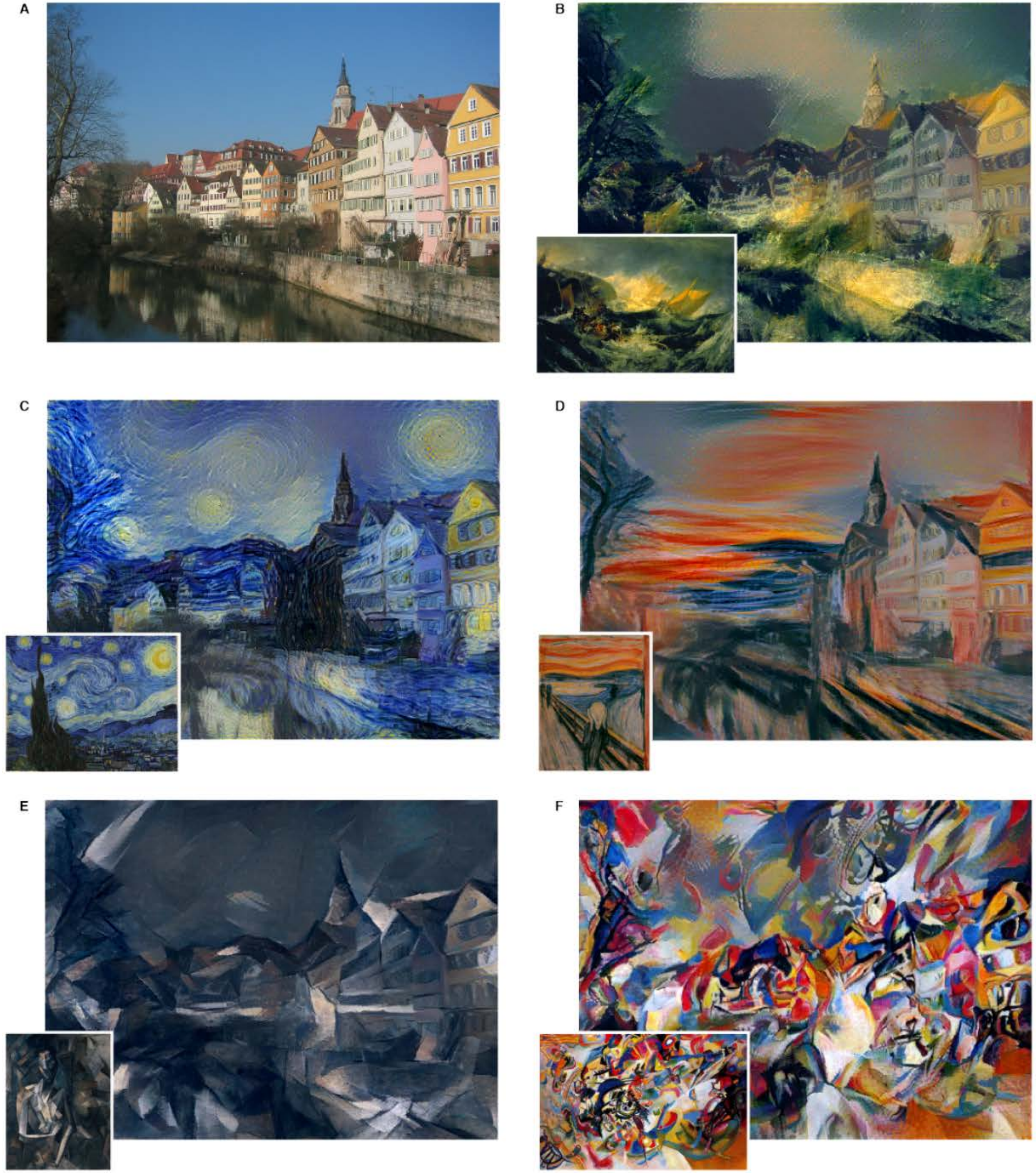
Additional Bayesian Bonus:
By monitoring posterior fluctuations of weights one can determine their fixed point precision.

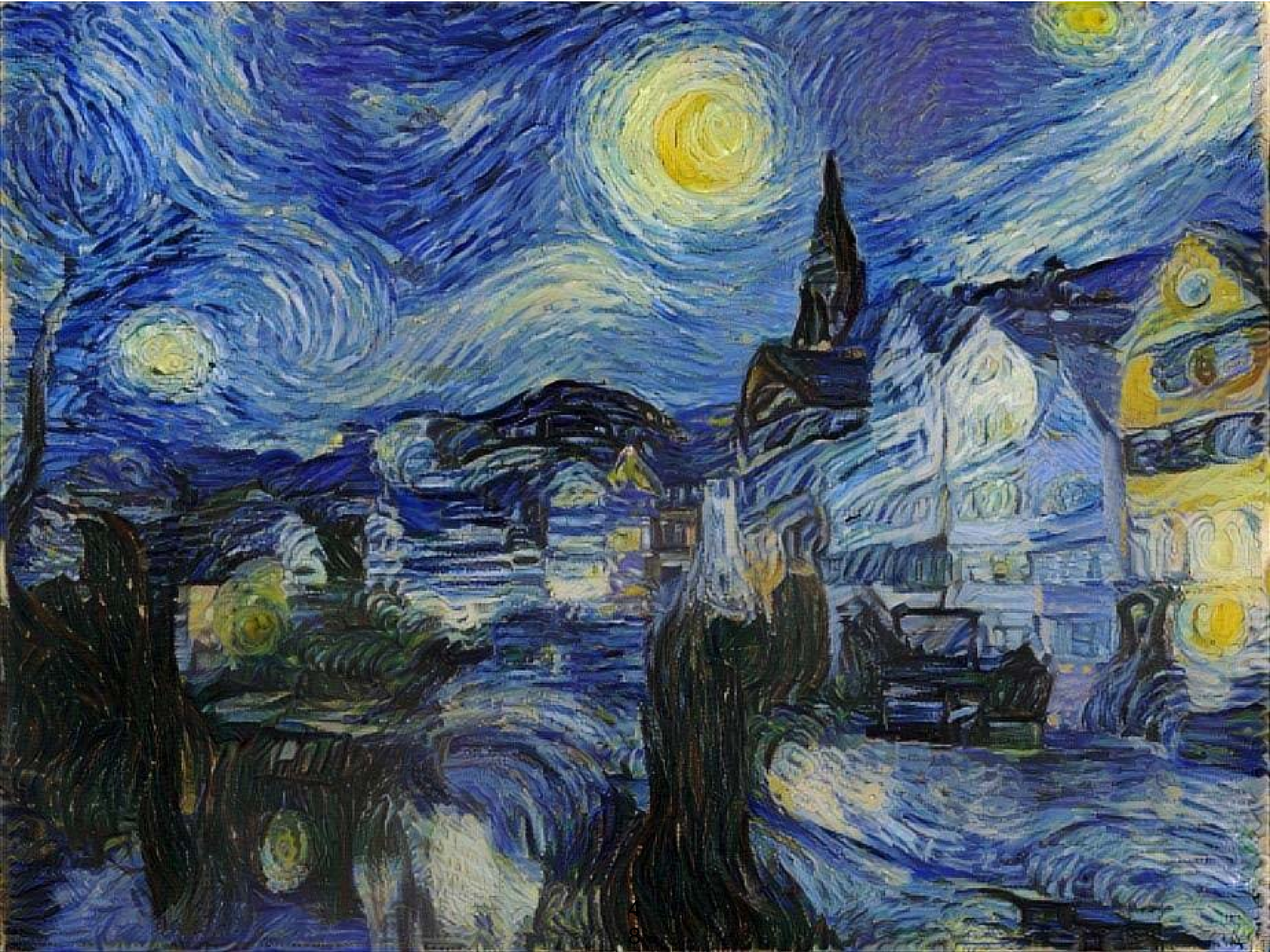
Model			Compression Rates (Error %)		
Original Error %	Method	$\frac{\lvert \mathbf{w} \neq 0 \rvert}{\lvert \mathbf{w} \rvert}\%$	Pruning	Fast Prediction	Maximum Compression
LeNet-300-100	DC	8.0	6 (1.6)	-	40 (1.6)
	DNS	1.8	28* (2.0)	-	-
1.6	SWS	4.3	12* (1.9)	-	64(1.9)
	Sparse VD	2.2	21(1.8)	84(1.8)	113 (1.8)
	GNJ	10.8	9(1.8)	36(1.8)	58(1.8)
	GHS	10.6	9(1.8)	23(1.9)	59(2.0)
LeNet-5-Caffe	DC	8.0	6*(0.7)	-	39(0.7)
	DNS	0.9	55*(0.9)	-	108(0.9)
0.9	SWS	0.5	100*(1.0)	-	162(1.0)
	Sparse VD	0.7	63(1.0)	228(1.0)	365(1.0)
	GNJ	0.9	108(1.0)	361(1.0)	573(1.0)
	GHS	0.6	156(1.0)	419(1.0)	771(1.0)
VGG	GNJ	6.7	14(8.6)	56(8.8)	95(8.6)
8.4	GHS	5.5	18(9.0)	59(9.0)	116(9.2)

Compression rate of a factor 700x with no loss in accuracy!

Part IV: Deep Art

AI & Kunst





De Visuele Turing Test

VS

van Gogh et al.

CNN

De Visuele Turing Test

van Gogh et
al.

CN
N

John Singer Sargent
"White Ships"

De Visuele Turing Test

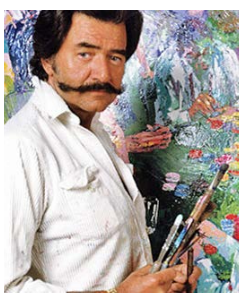
van Gogh et
al.

CN
N

De Visuele Turing Test

van Gogh et
al.

CN
N



Leroy Neiman
"Mickey Mantle"

Conclusions

Deep Learning is fun! (Deepdream)

- Deep Learning is a huge hammer that could be interesting to physics…

- Physics technology is now making inroads into deep learning
  (it's a good time to enter the field)

- We discussed:

1. Lie groups and symmetry transformations to understand equivariance
2. Variational free energies to for probabilistic / Bayesian deep learning

Acknowledgements

QUALCOMM

BOSCH

SAP

Google

PHILIPS

NWO

STW

facebook