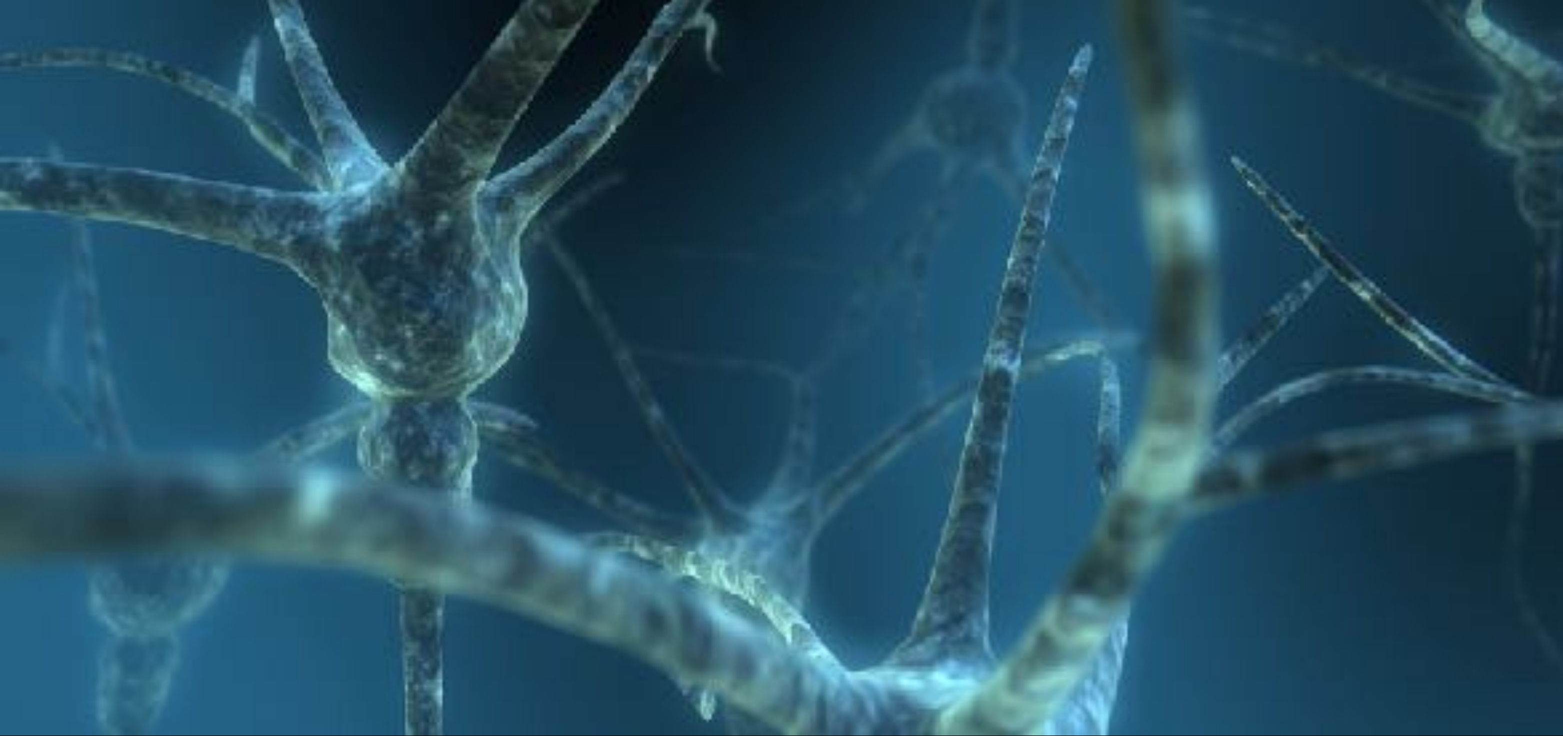
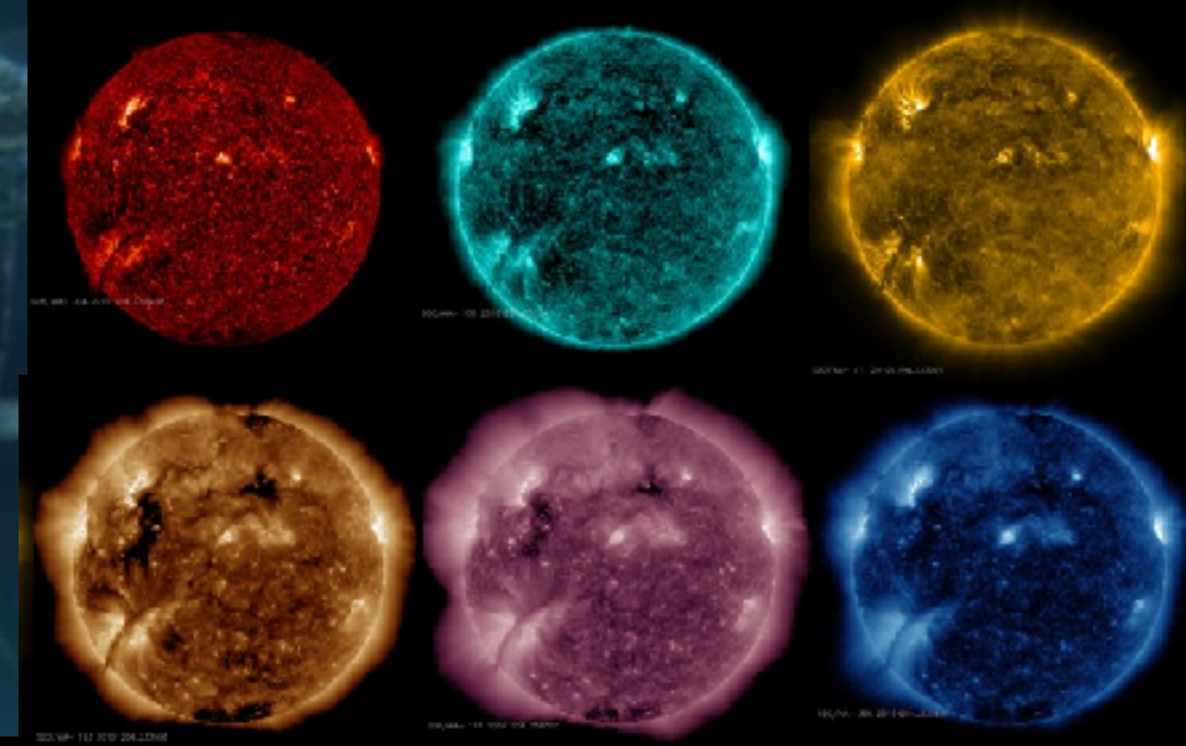
 

BASICS OF DEEP LEARNING

a. asensio ramos
@aasensior
github.com/aasensio

CONTENT

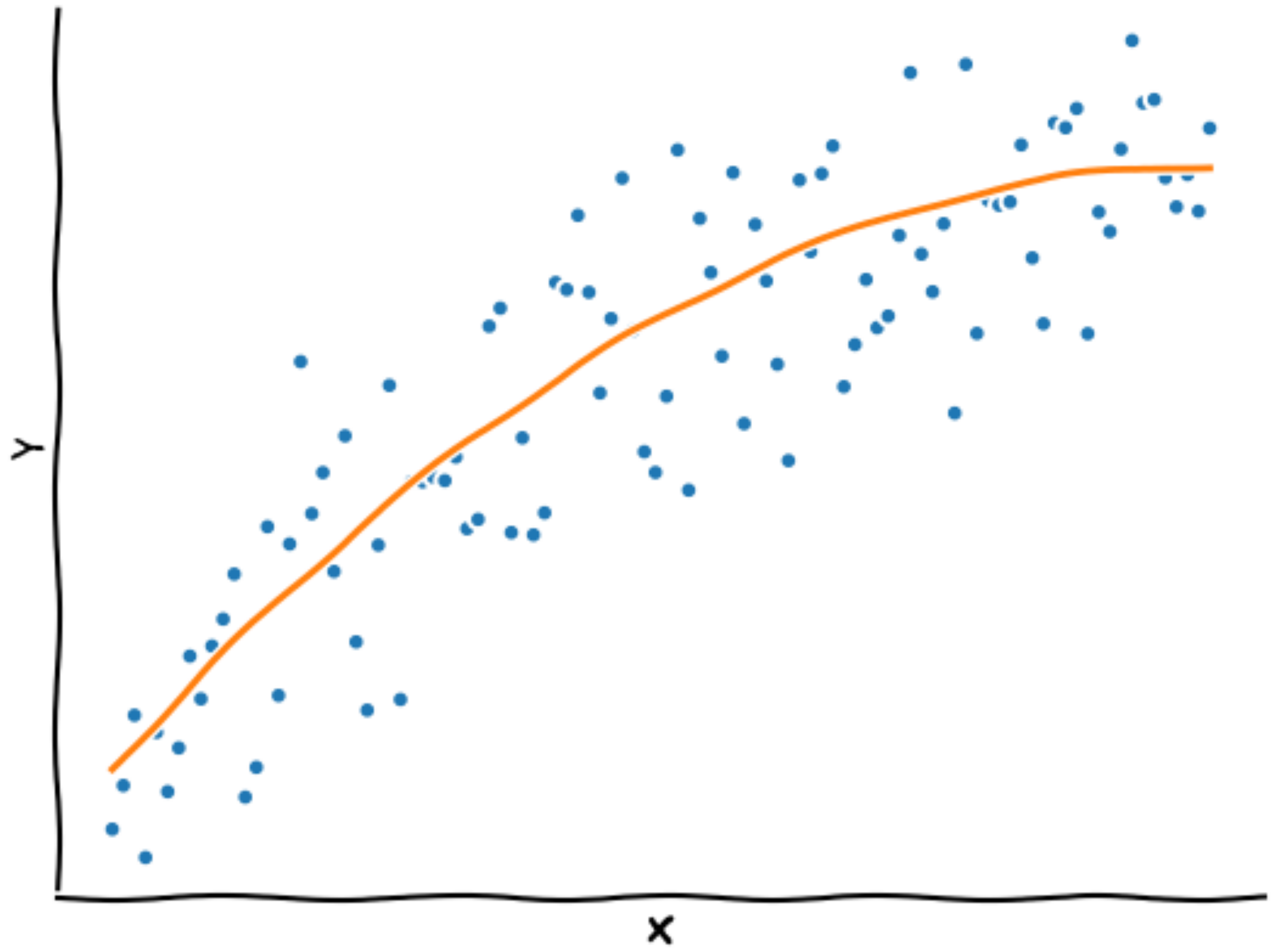
https://github.com/aasensio/solarnet19

Introduction

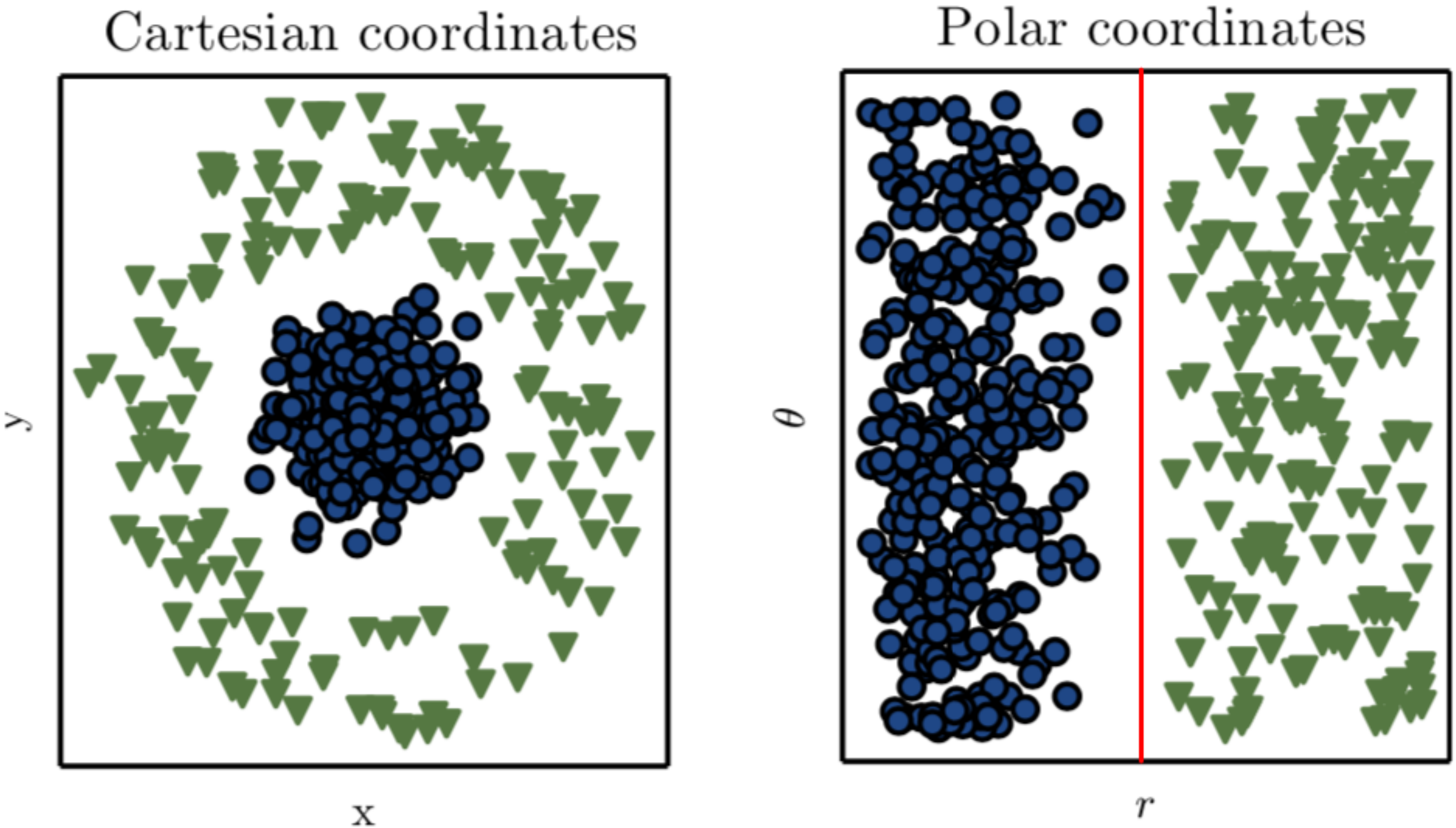
what is machine learning?

the focus of Machine Learning (ML) is to give computers the ability to learn from data, so that they may accomplish tasks that humans have difficulty expressing in pure code

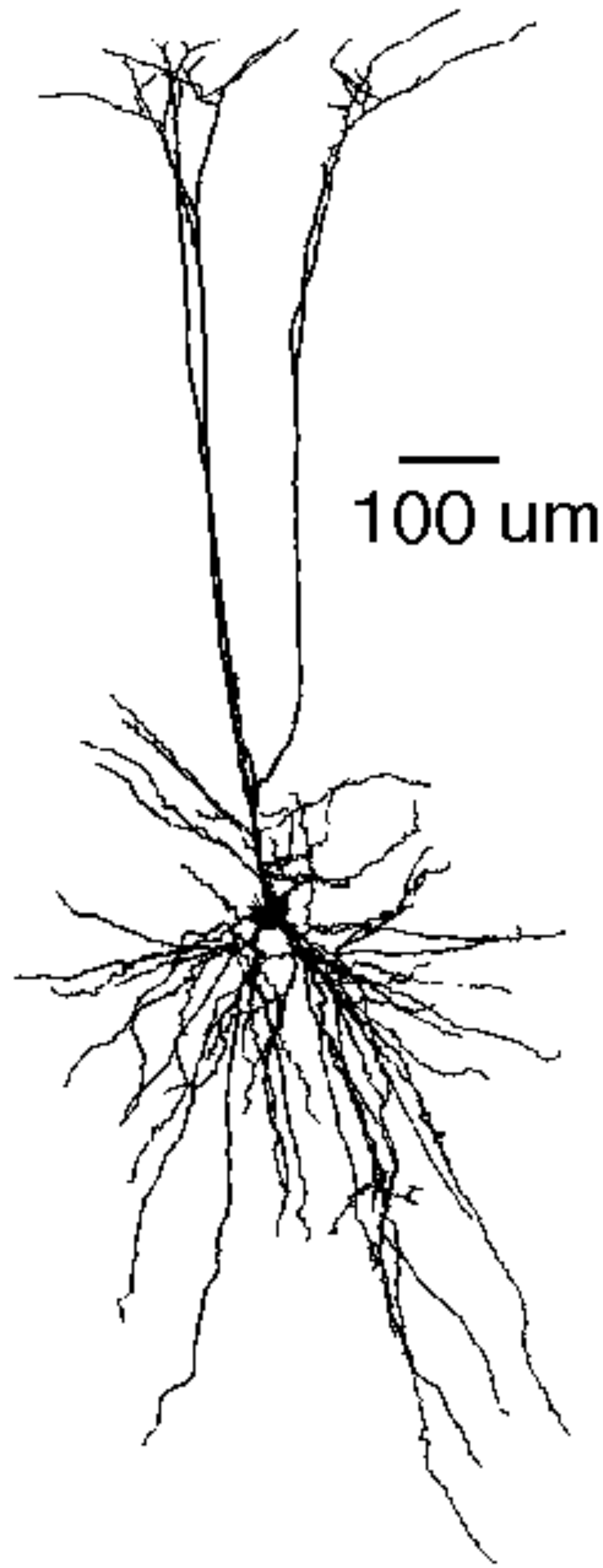
REGRESSION



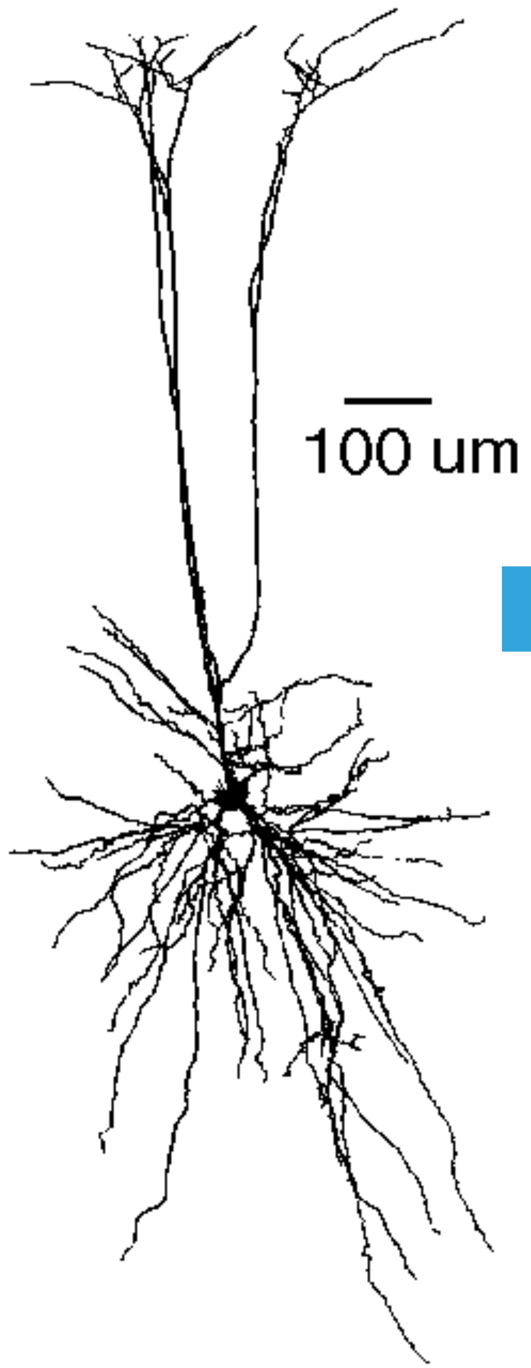
CLASSIFICATION

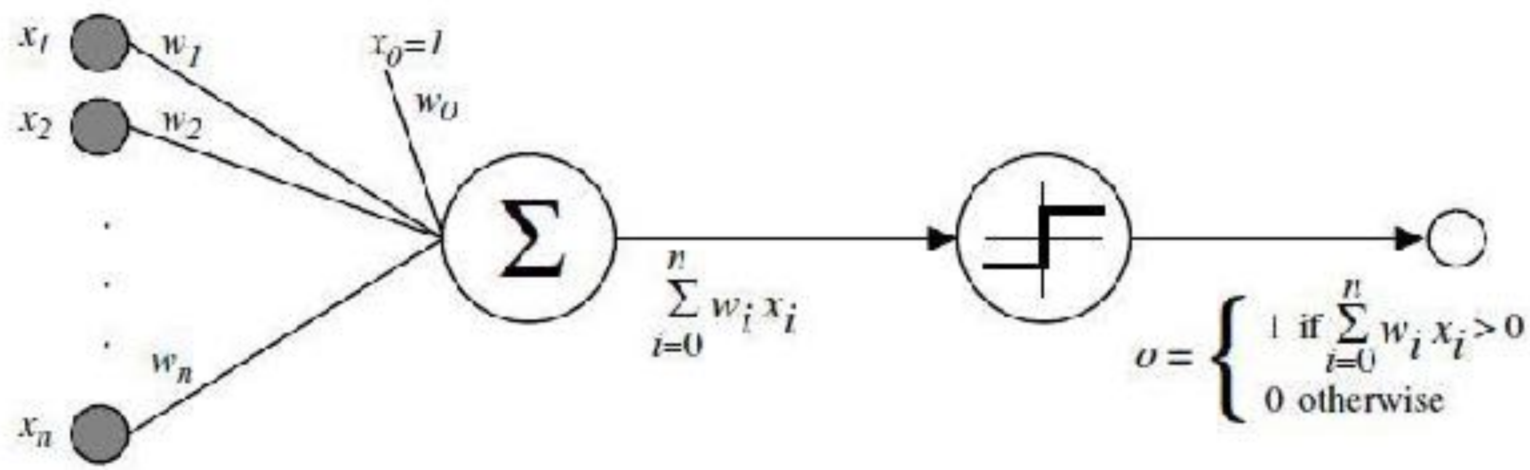


NATURAL NEURAL NETWORKS

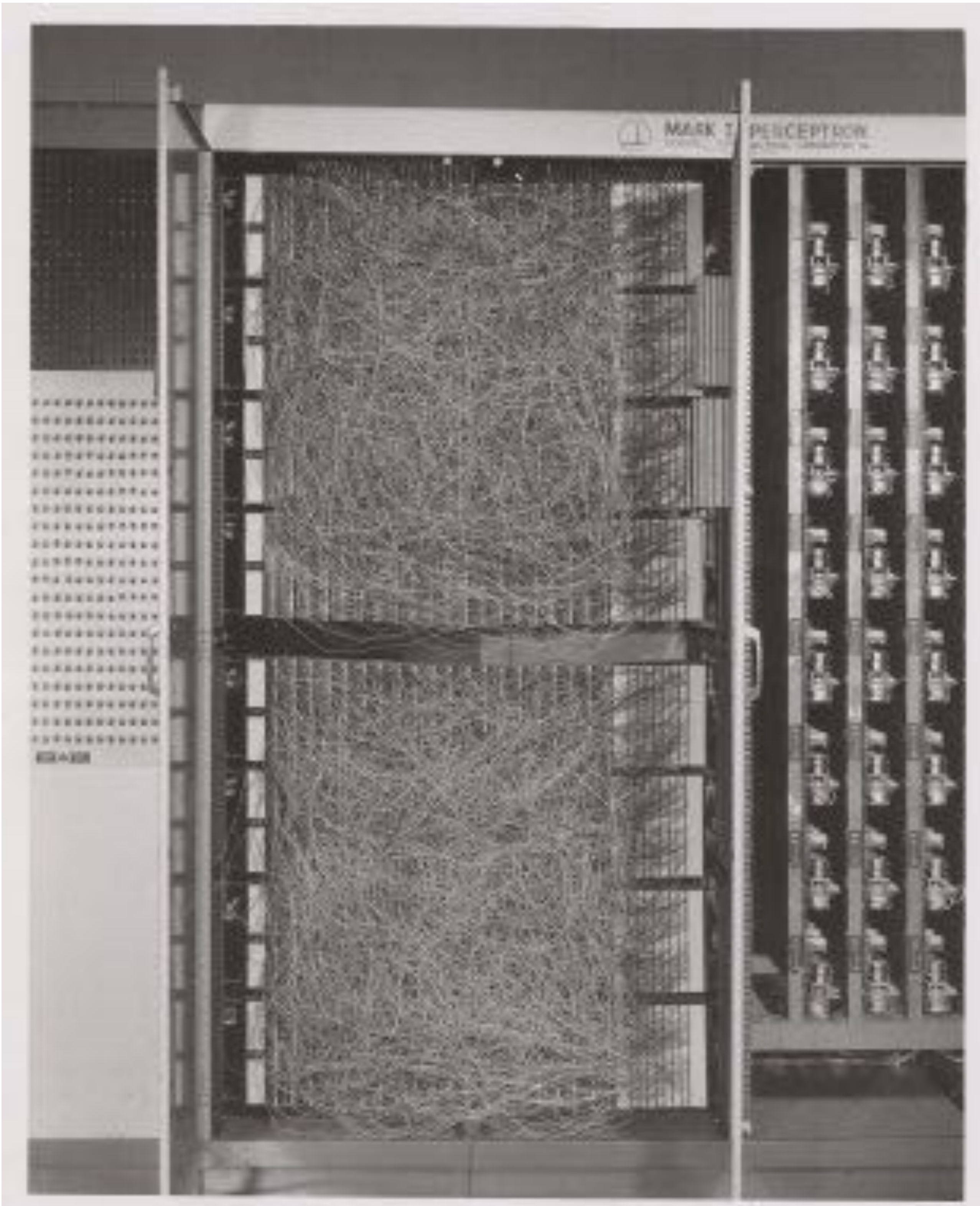


ARTIFICIAL NEURAL NETWORKS



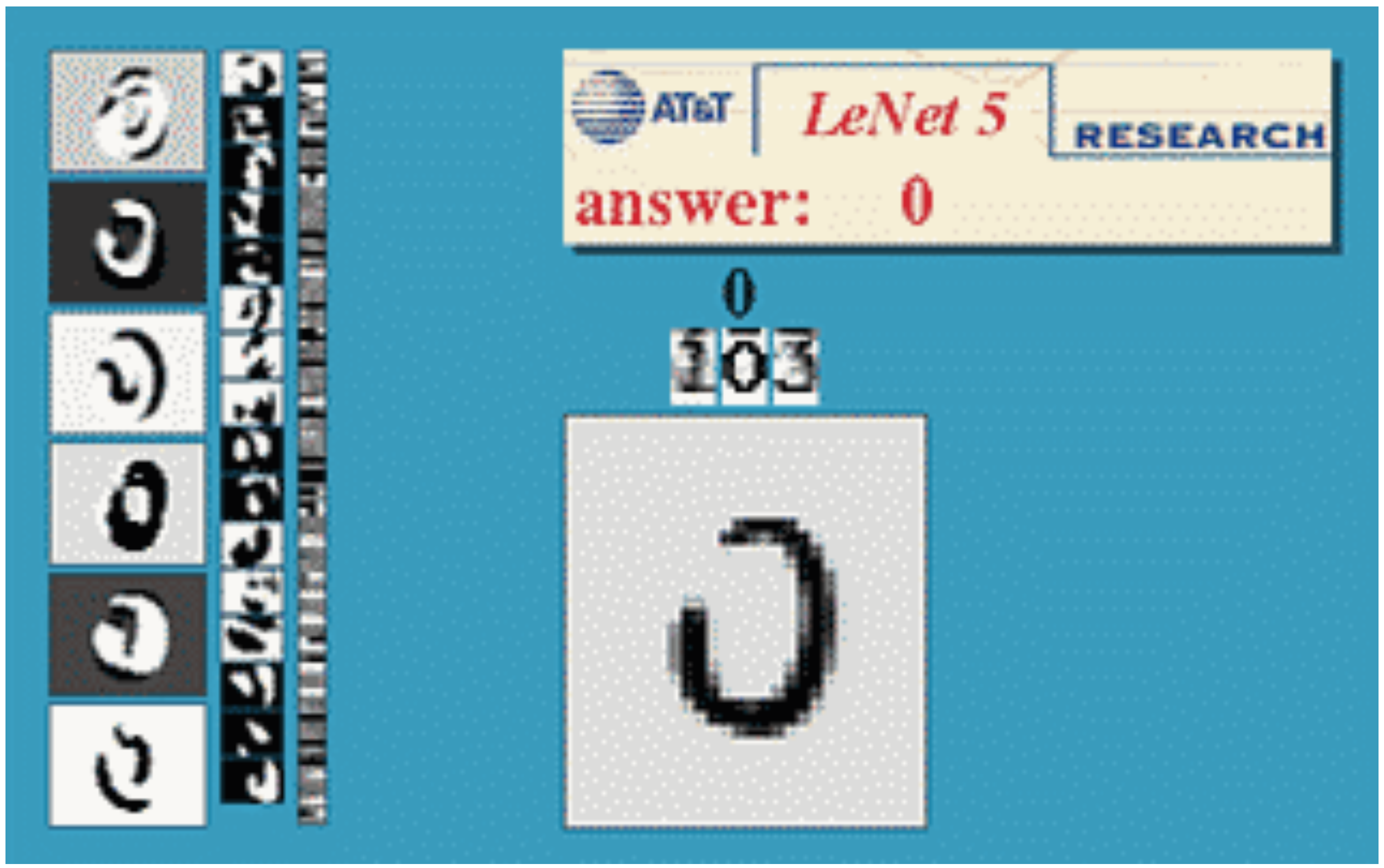


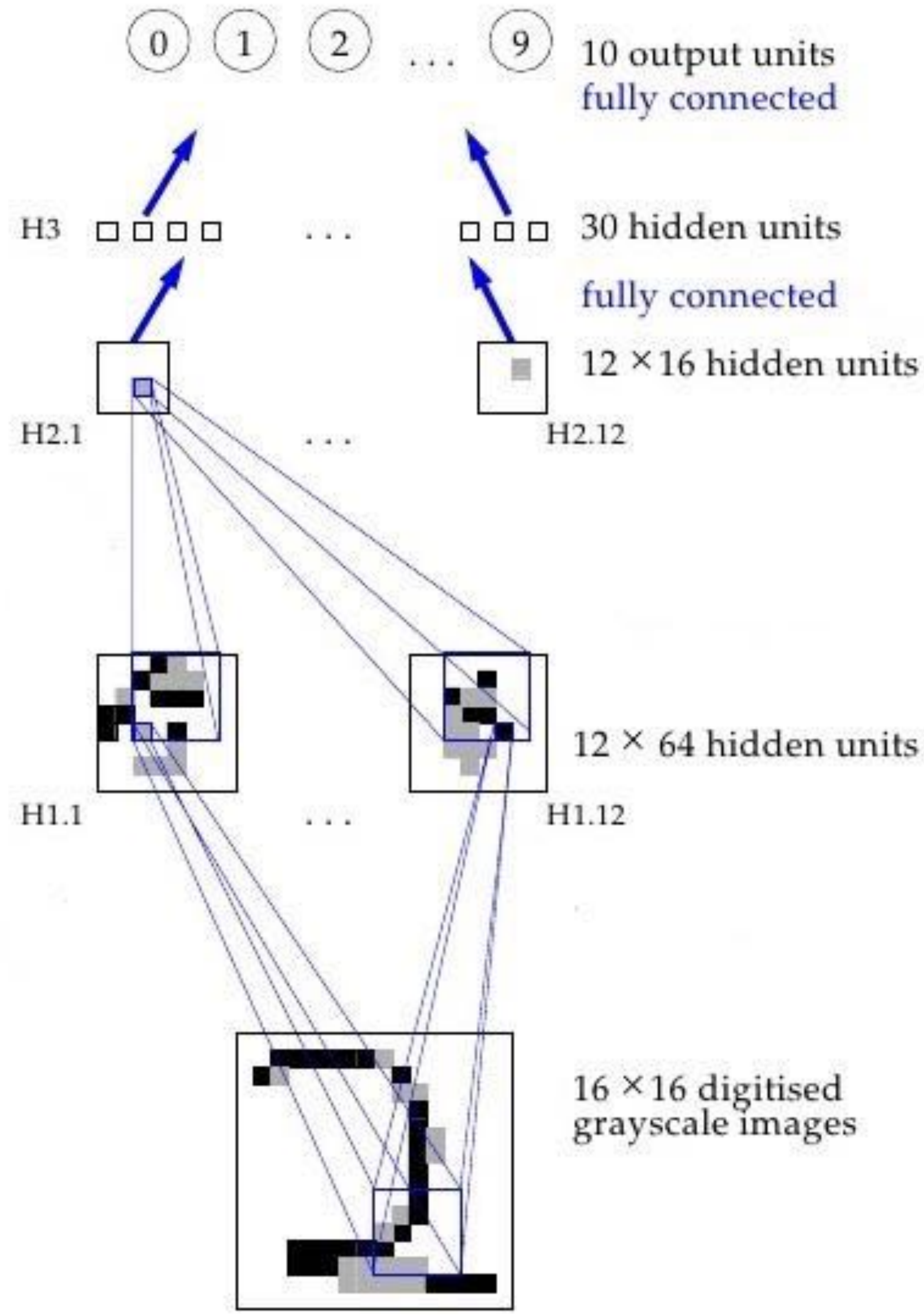
MARK I PERCEPTRON : FRANK ROSENBLATT



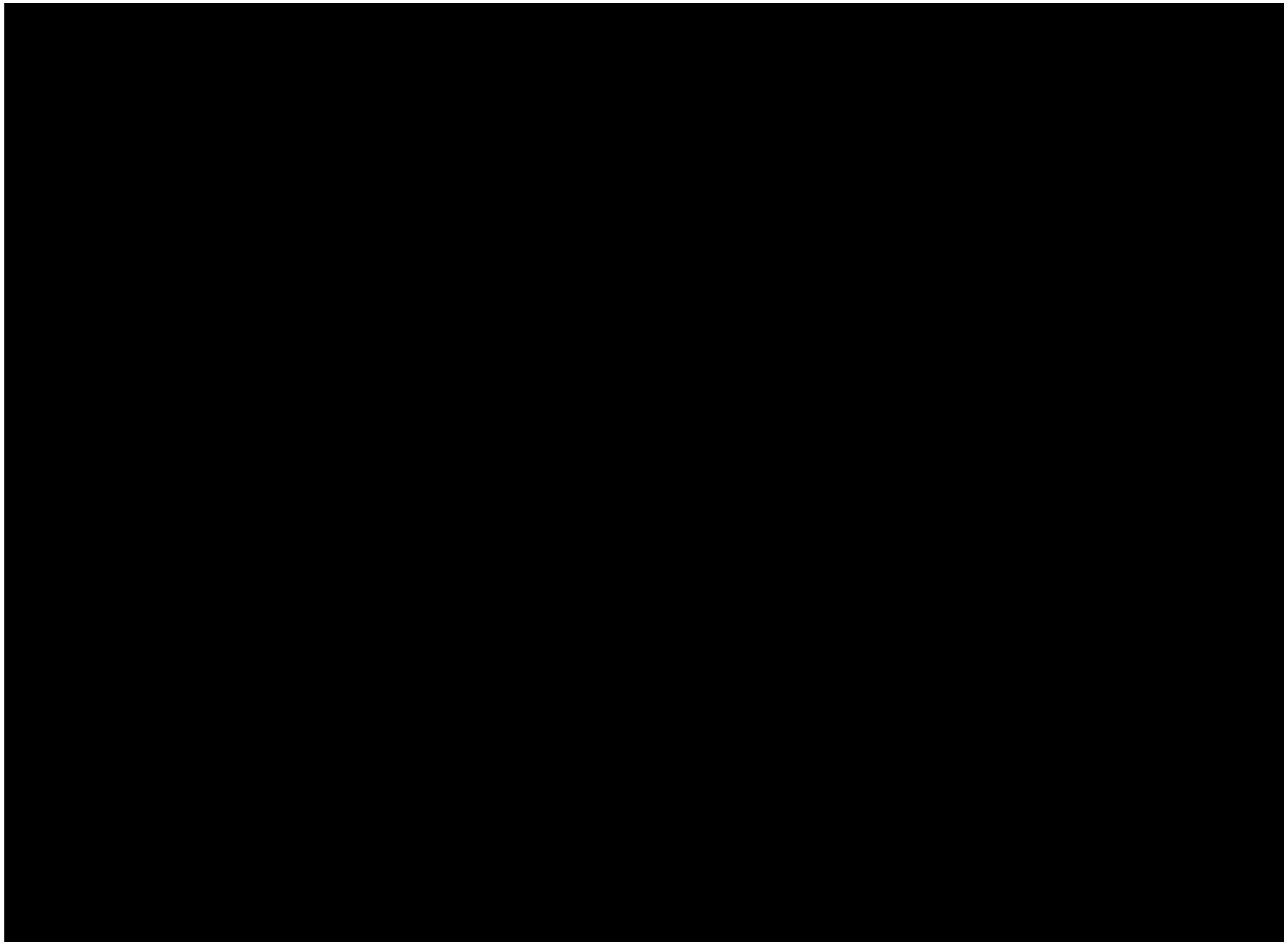
Source: Arvin Calspan Advanced Technology Center; Hecht-Nielsen, R. Neurocomputing (Reading, Mass.: Addison-Wesley, 1990)

CONVOLUTIONAL NEURAL NETWORKS : YANN LACUN





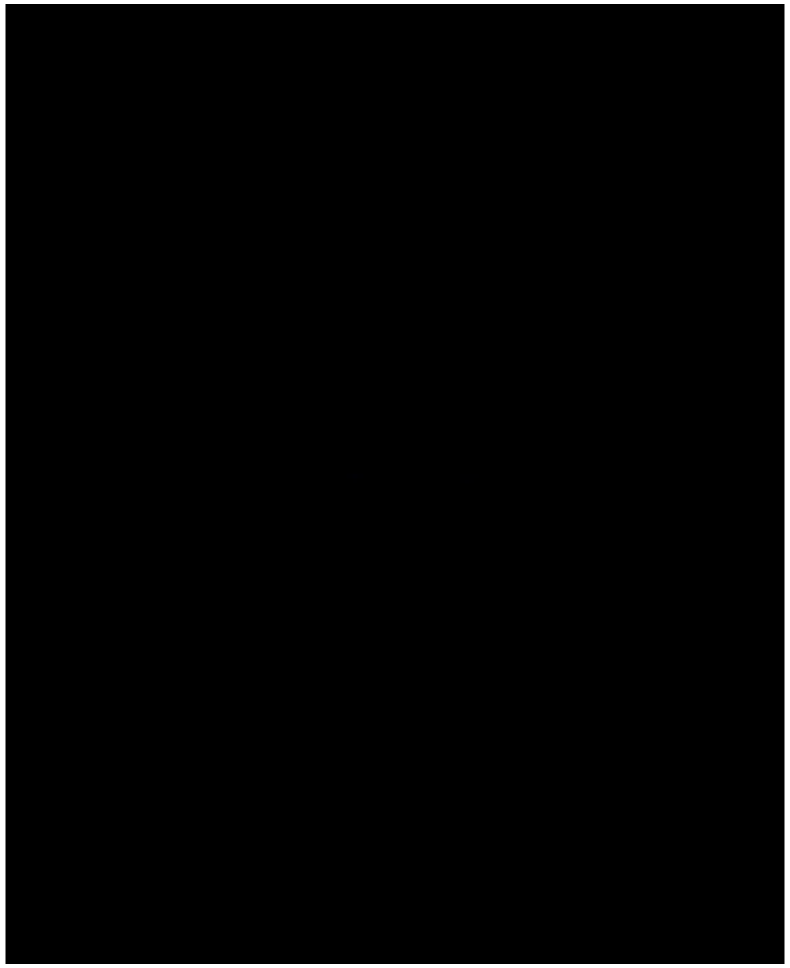
CONVOLUTIONAL NEURAL NETWORKS : YANN LECUN



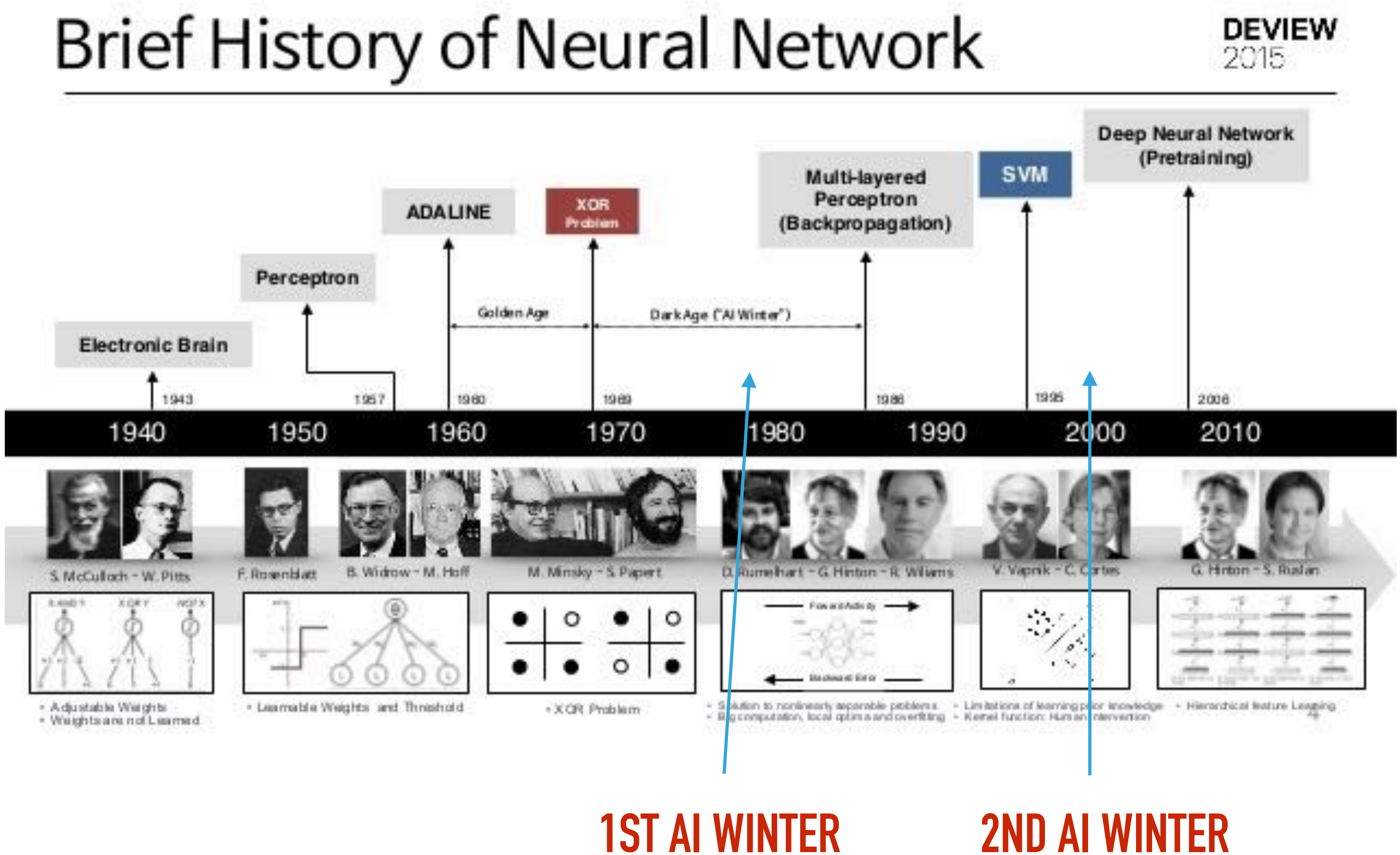
RANDOM PLAYER

TRAINED PLAYER

After 240 min of training



AI WINTERS



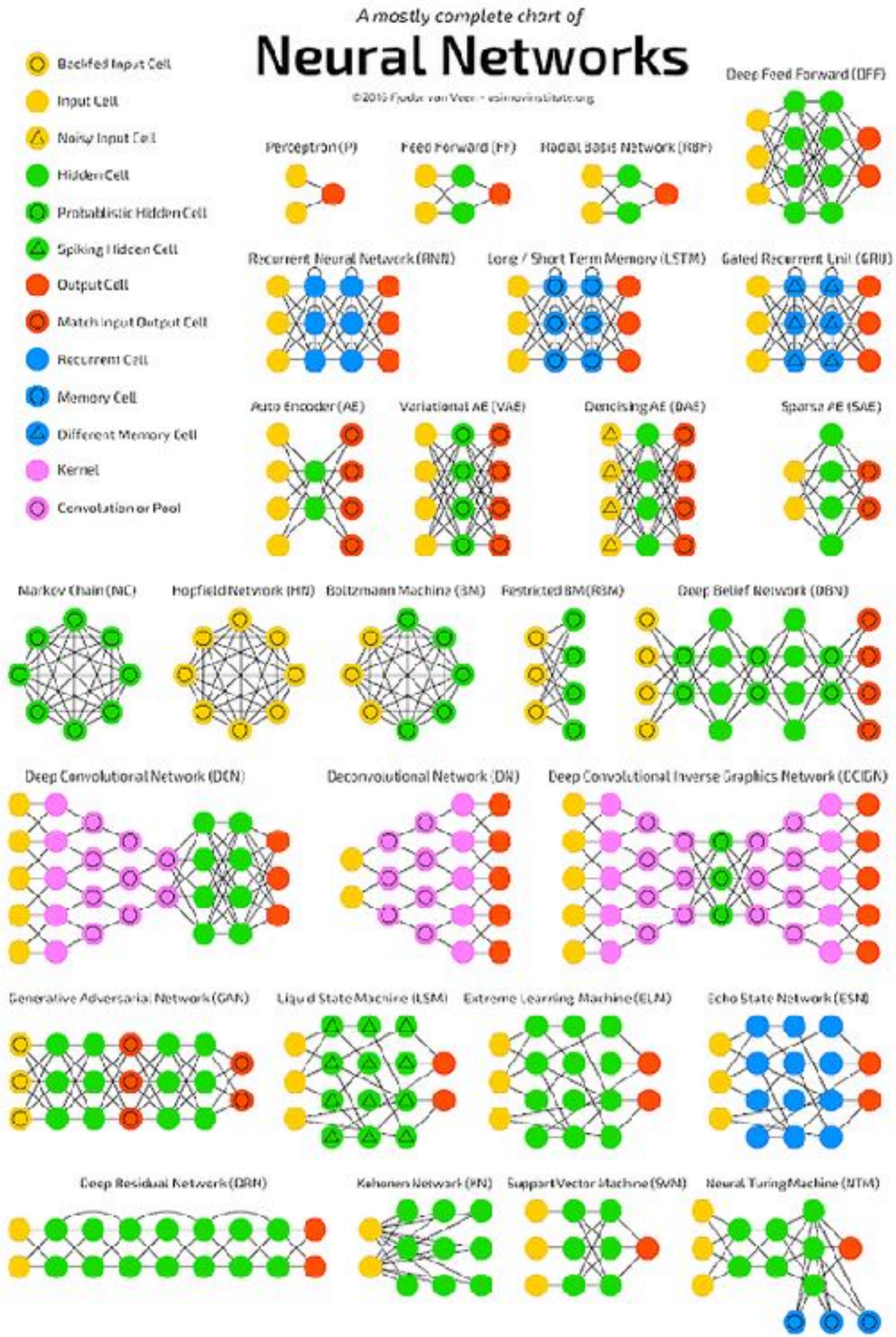
Brief History of Neural Network
DEVIEW 2015
Electronic Brain
Perceptron
ADALINE
XOR Problem
Multi-layered Perceptron (Backpropagation)
SVM
Deep Neural Network (Pretraining)
Golden Age
Dark Age ("AI Winter")
1943
1957
1960
1969
1986
1995
2006
1940
1950
1960
1970
1980
1990
2000
2010
S. McCulloch - W. Pitts
F. Rosenblatt
B. Widrow - M. Hoff
M. Minsky - S. Papert
D. Rumelhart - G. Hinton - R. Williams
V. Vapnik - C. Cortes
G. Hinton - S. Ruslan
• Adjustable Weights
• Weights are not Learned
• Learnable Weights and Threshold
• XOR Problem
1ST AI WINTER
2ND AI WINTER

BE PERSEVERANT

Yoshua Bengio
Montreal University

Yann LeCun
Facebook+NYU

Geoffrey Hinton
Google+Toronto



A mostly complete chart of
Neural Networks
©2016 Fjodor van Veen - asimovinstitute.org
Backfed Input Cell
Input Cell
Noisy Input Cell
Hidden Cell
Probablistic Hidden Cell
Spiking Hidden Cell
Output Cell
Match Input Output Cell
Recurrent Cell
Memory Cell
Different Memory Cell
Kernel
Convolution or Pool
Perceptron (P)
Feed Forward (FF)
Radial Basis Network (RBF)
Deep Feed Forward (DFF)
Recurrent Neural Network (RNN)
Long / Short Term Memory (LSTM)
Gated Recurrent Unit (GRU)
Auto Encoder (AE)
Variational AE (VAE)
Denoising AE (DAE)
Sparse AE (SAE)
Markov Chain (MC)
Hopfield Network (HN)
Boltzmann Machine (BM)
Restricted BM (RBM)
Deep Belief Network (DBN)
Deep Convolutional Network (DCN)
Deconvolutional Network (DN)
Deep Convolutional Inverse Graphics Network (DCIGN)
Generative Adversarial Network (GAN)
Liquid State Machine (LSM)
Extreme Learning Machine (ELM)
Echo State Network (ESN)
Deep Residual Network (DRN)
Kohonen Network (KN)
Support Vector Machine (SVM)
Neural Turing Machine (NTM)

but is deep learning really a hype?

Baidu's Andrew Ng on Deep Learning and Innovation in Silicon Valley

Nervana Systems raises $3.3M to build hardware designed for deep learning

by Derrick Harris Aug. 21, 2014 - 5:48 AM PST

Artificially Intelligent Robot Scientists Could Be Next Project for Google's AI Firm

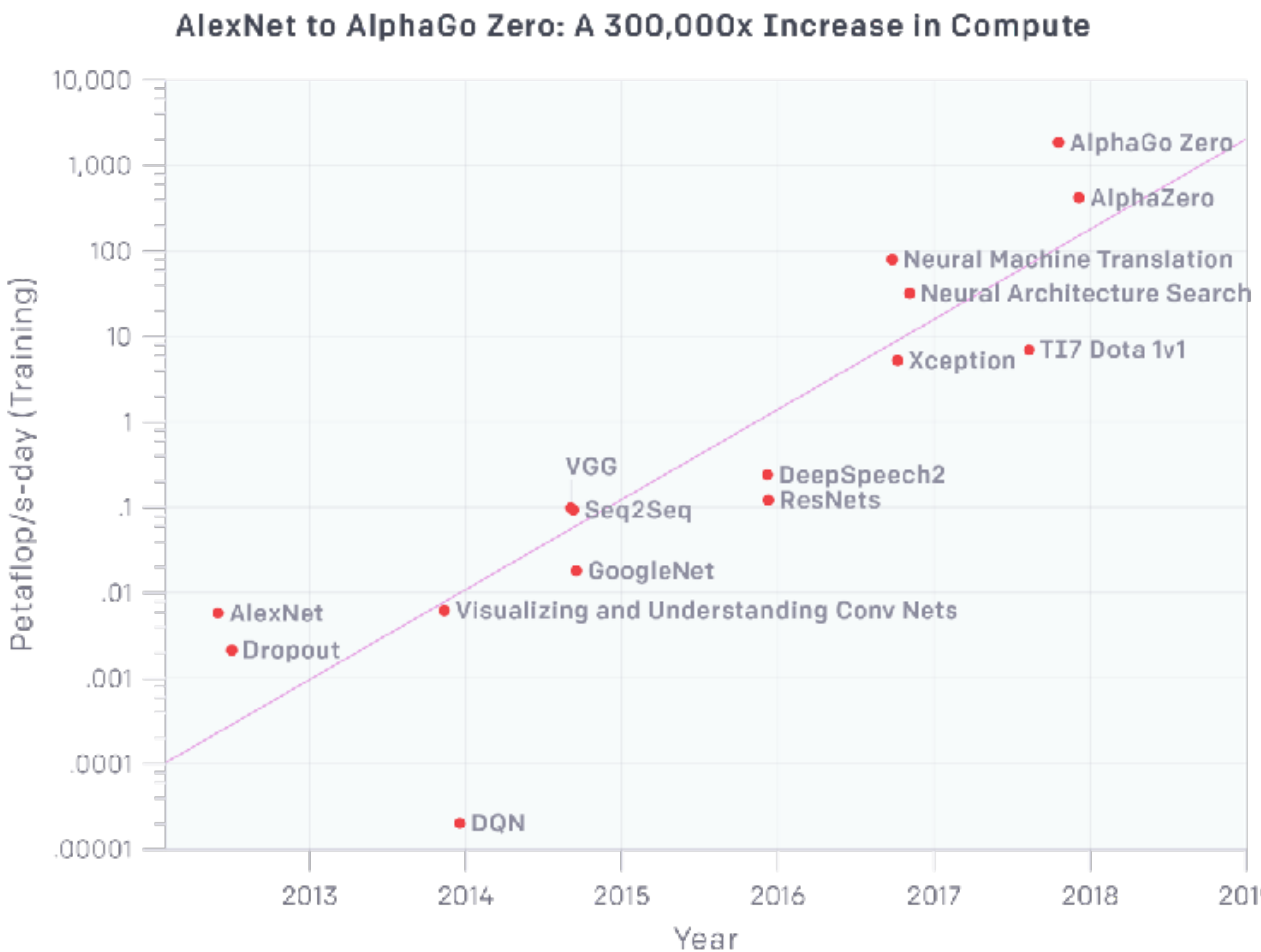
Deep learning might help you get an ultrasound at Walgreens

A Googler's Quest to Teach Machines How to Understand Emotions

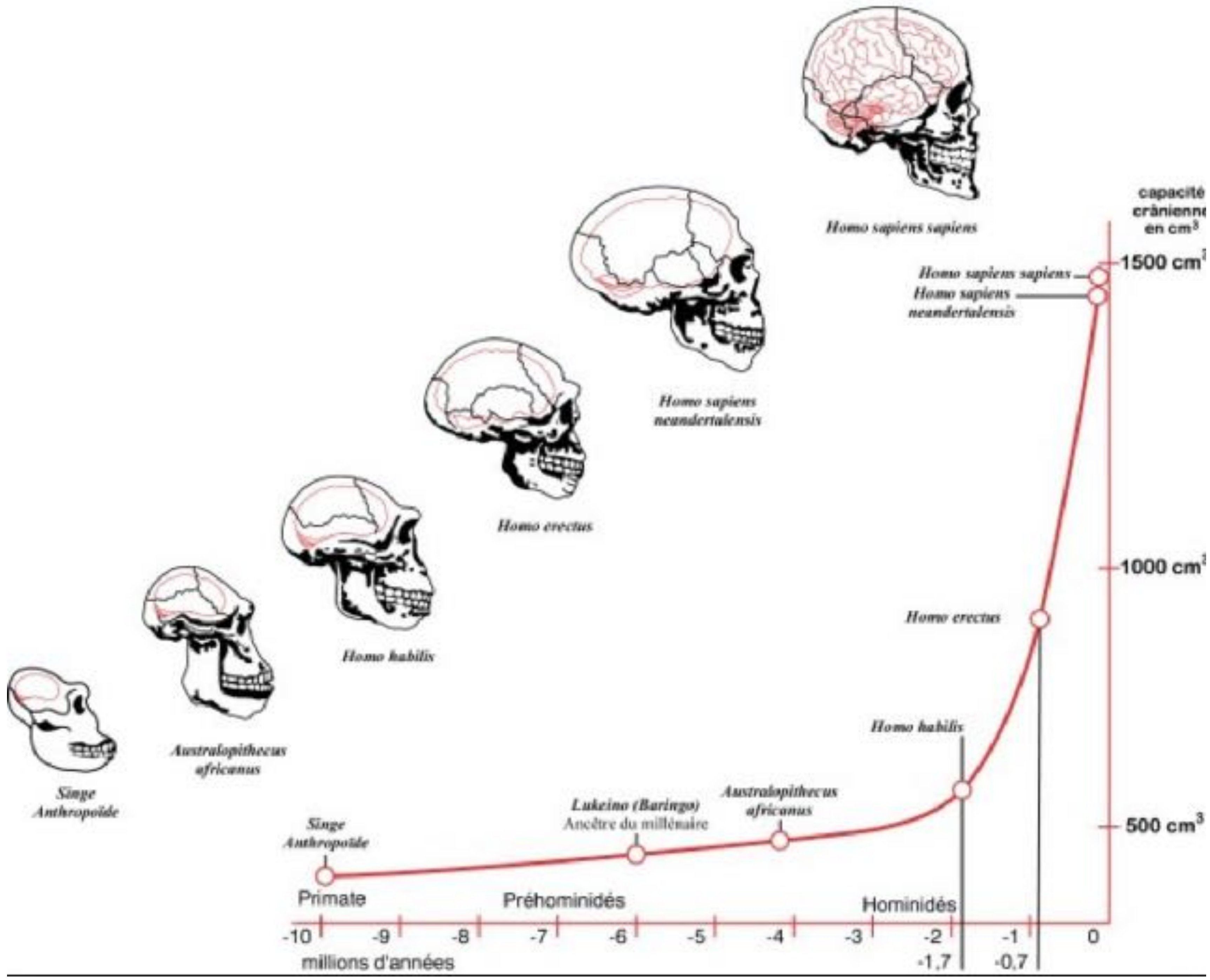
Google, Spotify, & Pandora bet a computer could generate a better playlist than you can

Butterfly Network Hopes to Bring Deep Learning AI to Medicine

Enlitic picks up $2M to help diagnose diseases with deep learning



https://blog.openai.com/ai-and-compute



Homo sapiens sapiens
Homo sapiens neandertalensis
Homo erectus
Homo habilis
Australopithecus africanus
Singe Anthropoïde
capacité crânienne en cm³
1500 cm³
1000 cm³
500 cm³
Homo sapiens sapiens
Homo sapiens neandertalensis
Homo erectus
Homo habilis
Australopithecus africanus
Lukeino (Baringo)
Ancêtre du millénaire
Singe Anthropoïde
Primate
Préhominidés
Hominidés
-10
-9
-8
-7
-6
-5
-4
-3
-2
-1
0
-1,7
-0,7
millions d'années

CURSE OF DIMENSIONALITY

Machine learning needs to fight the curse of dimensionality

CURSE OF DIMENSIONALITY

CURSE OF DIMENSIONALITY

CURSE OF DIMENSIONALITY

Dimension: 1

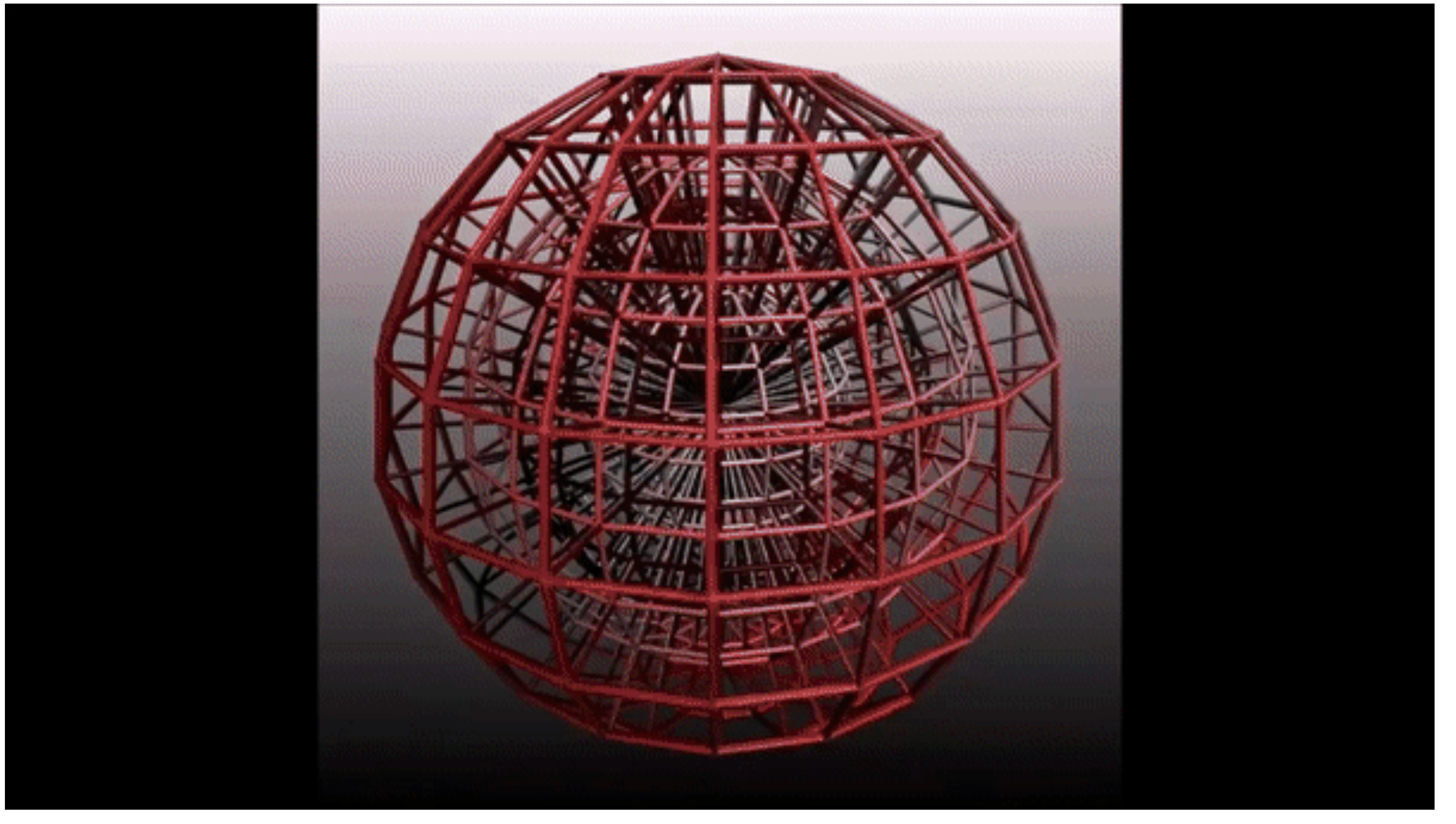
CURSE OF DIMENSIONALITY

Dimension: 2

CURSE OF DIMENSIONALITY

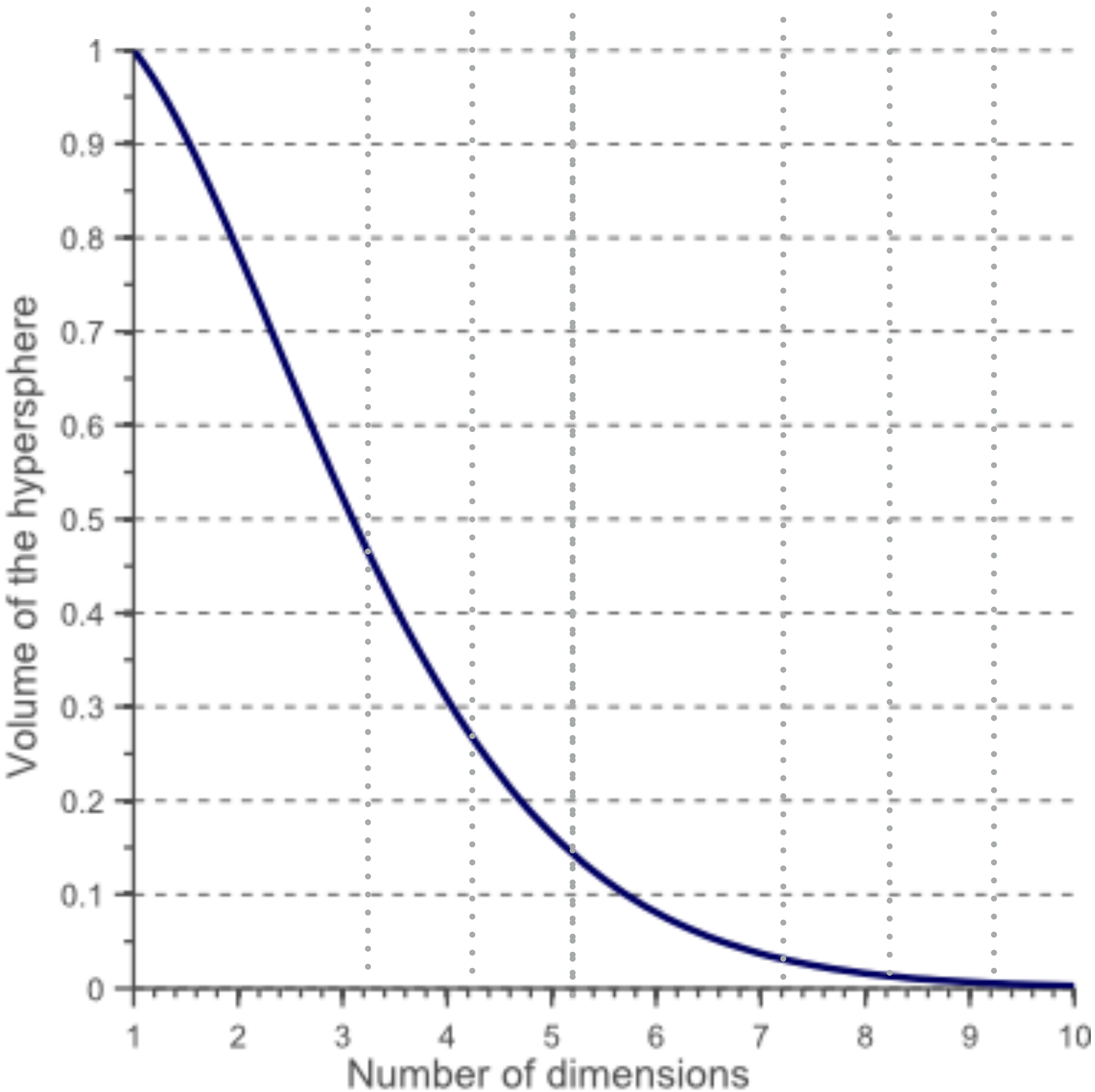
Dimension: 3

CURSE OF DIMENSIONALITY



Dimension: 4

HOW MUCH VOLUME CAN I FILL?

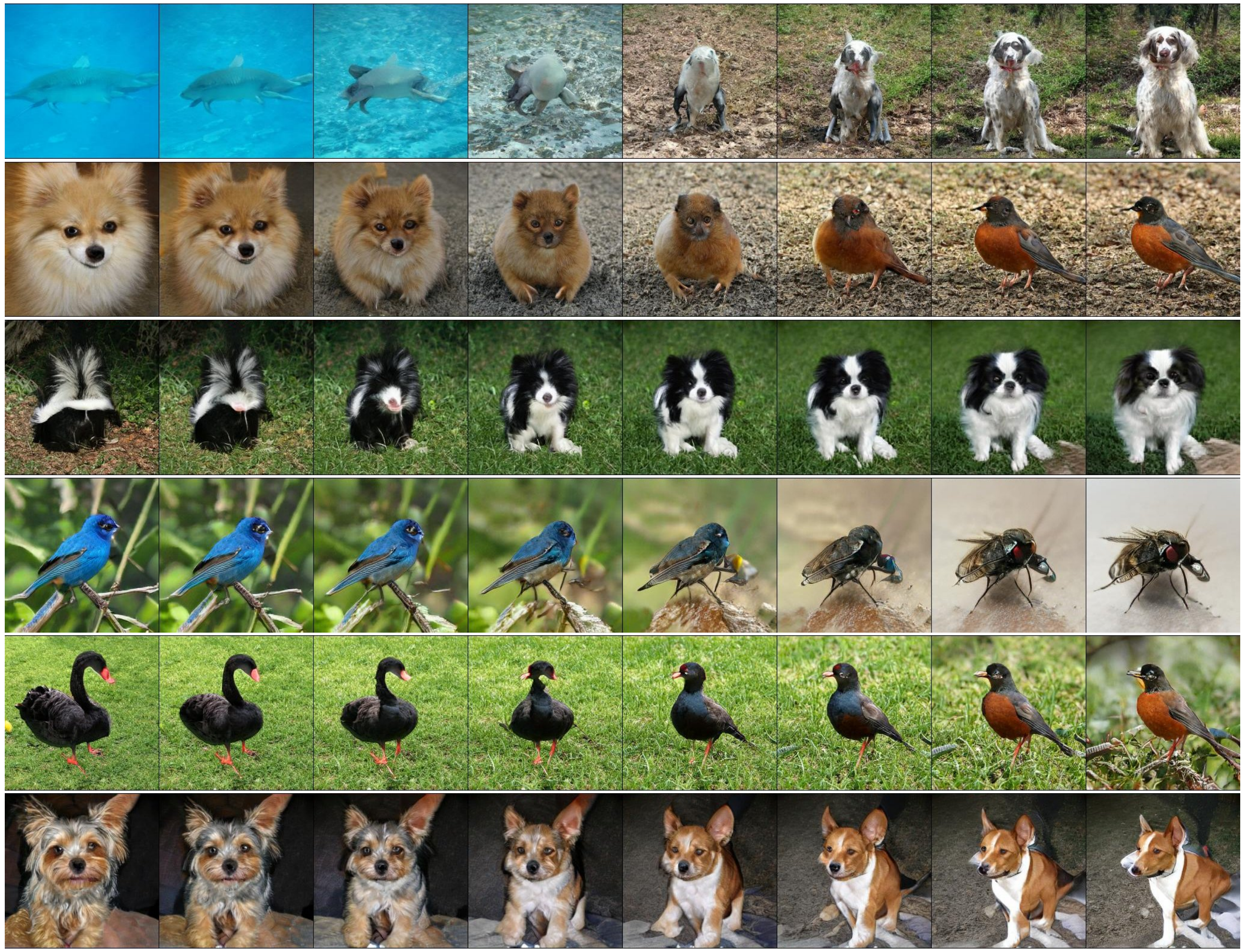


$$V(d) = \frac{\pi^{d/2}}{\Gamma\left(\frac{d}{2}+1\right)}\left(\frac{1}{2}\right)^d$$

The volume is on the borders!

Neural networks are specially suited to adapt to the data manifold

WHERE ARE WE NOW?

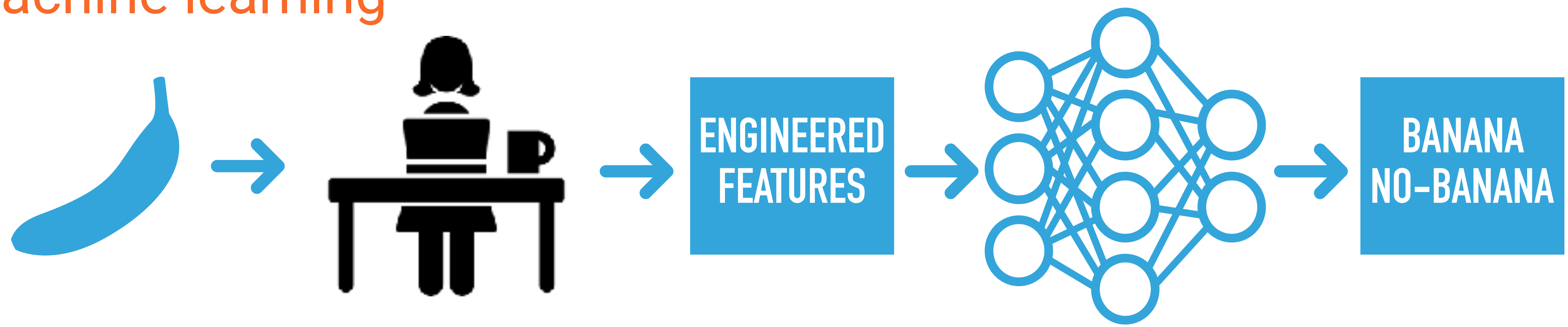


Brock et al. (2018)

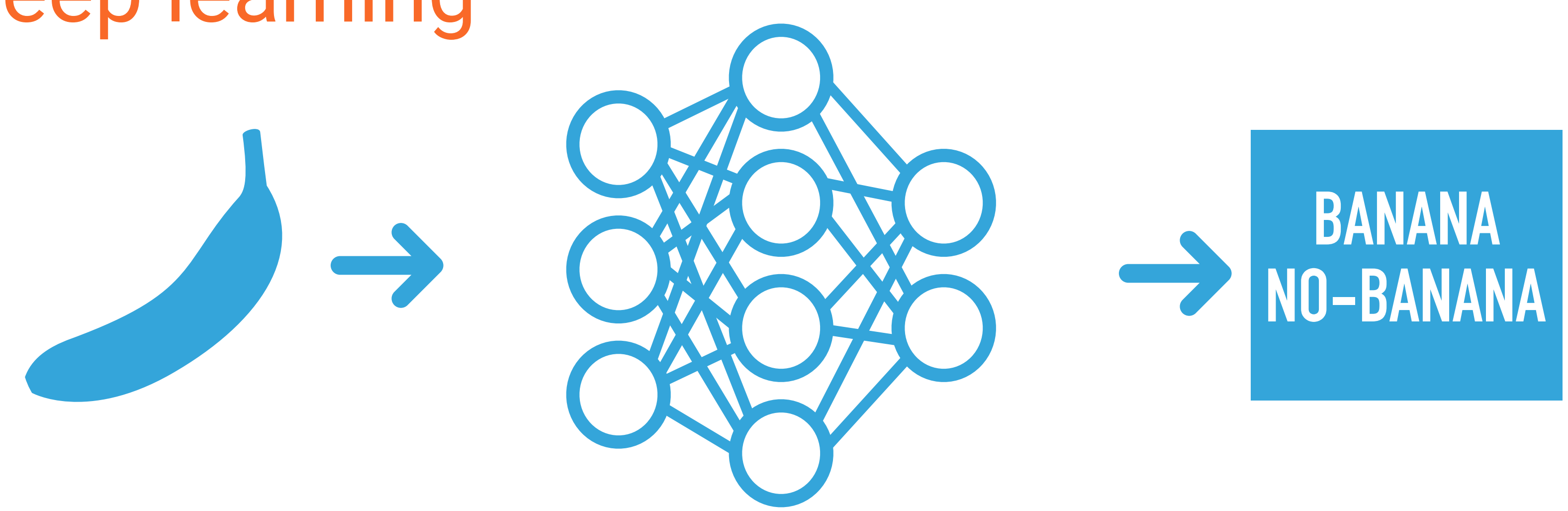
Basics

CLASSICAL MACHINE LEARNING VS. DEEP LEARNING

Machine learning

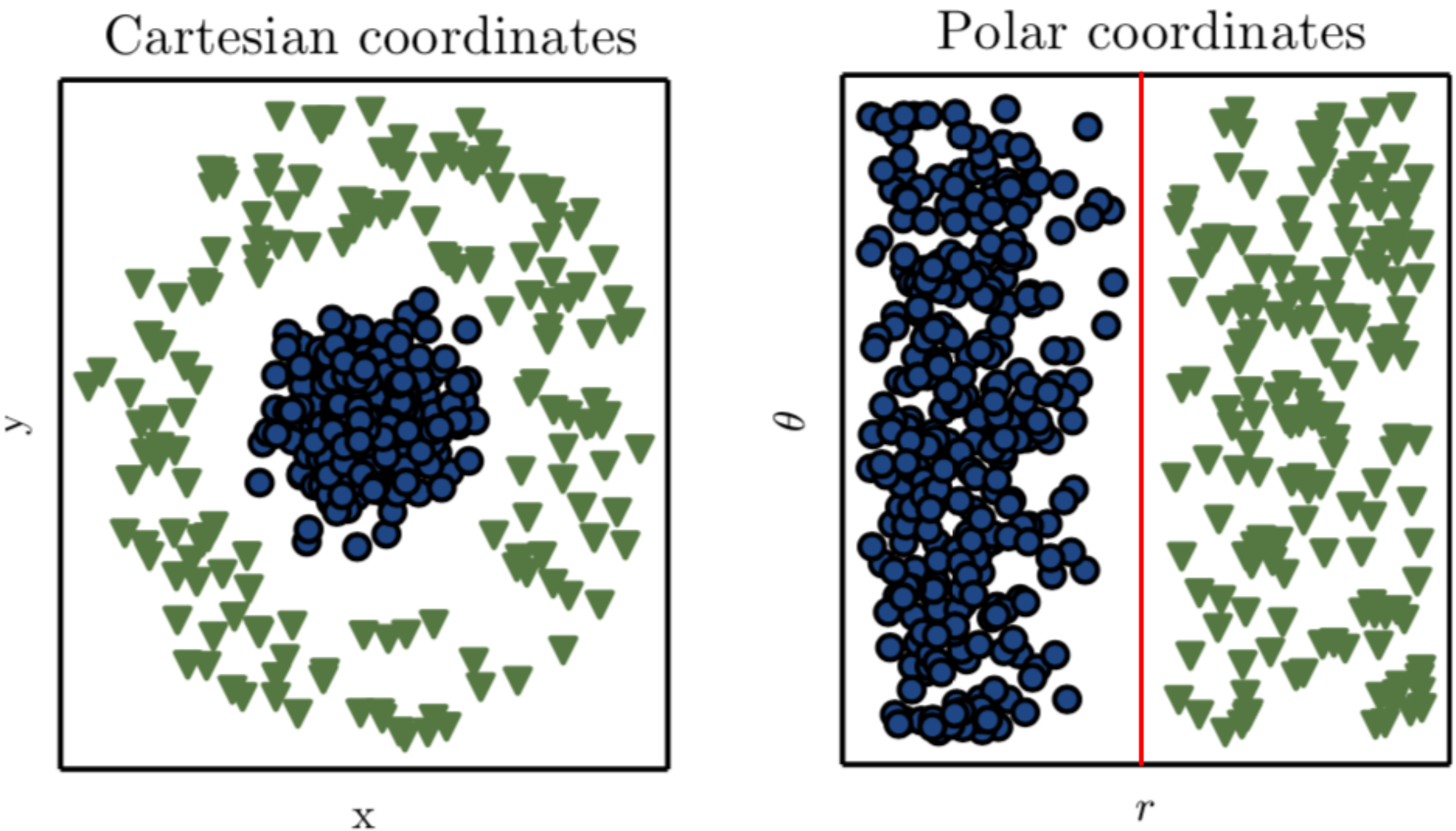


Deep learning

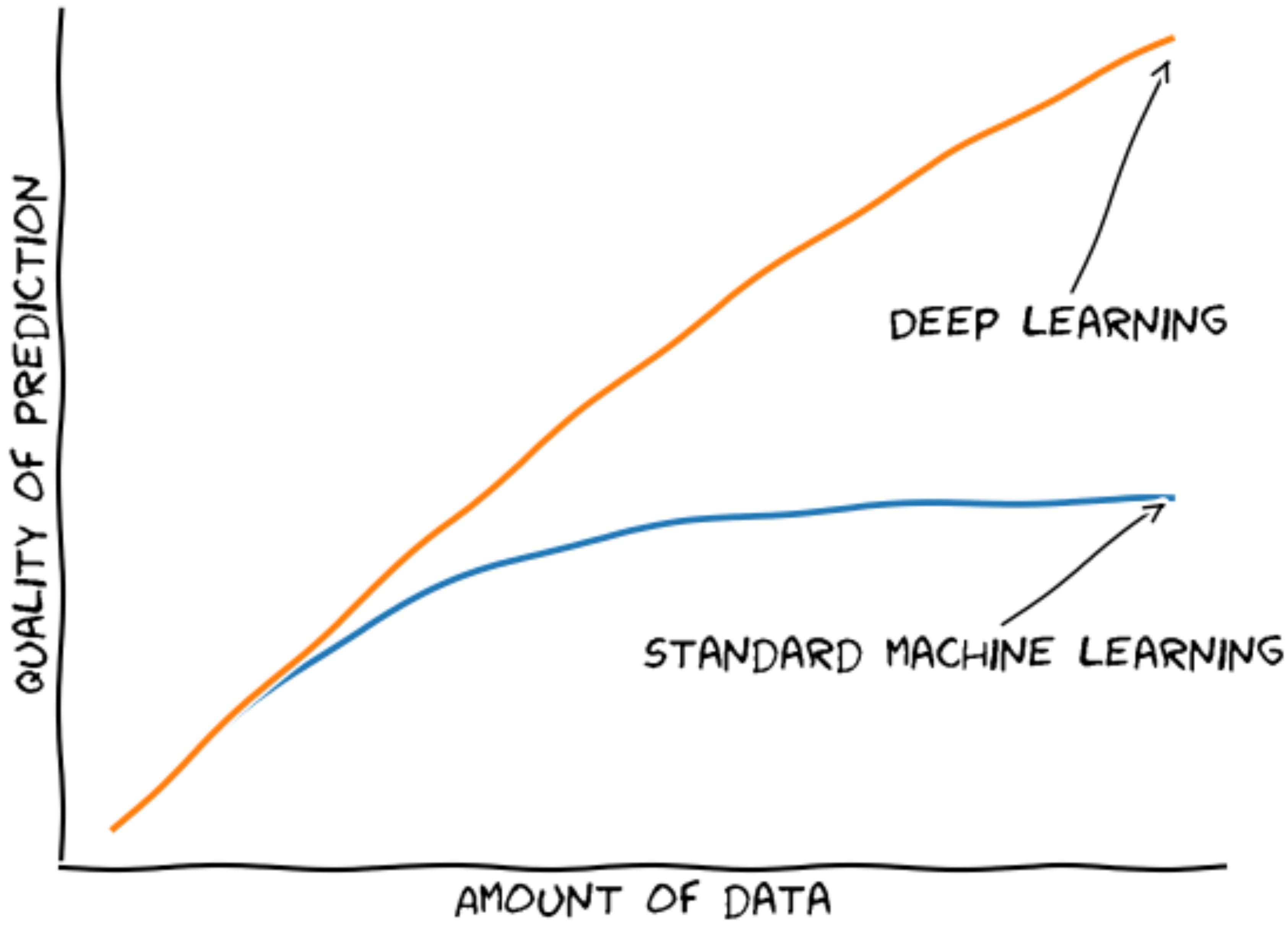


Credits: freepik.com

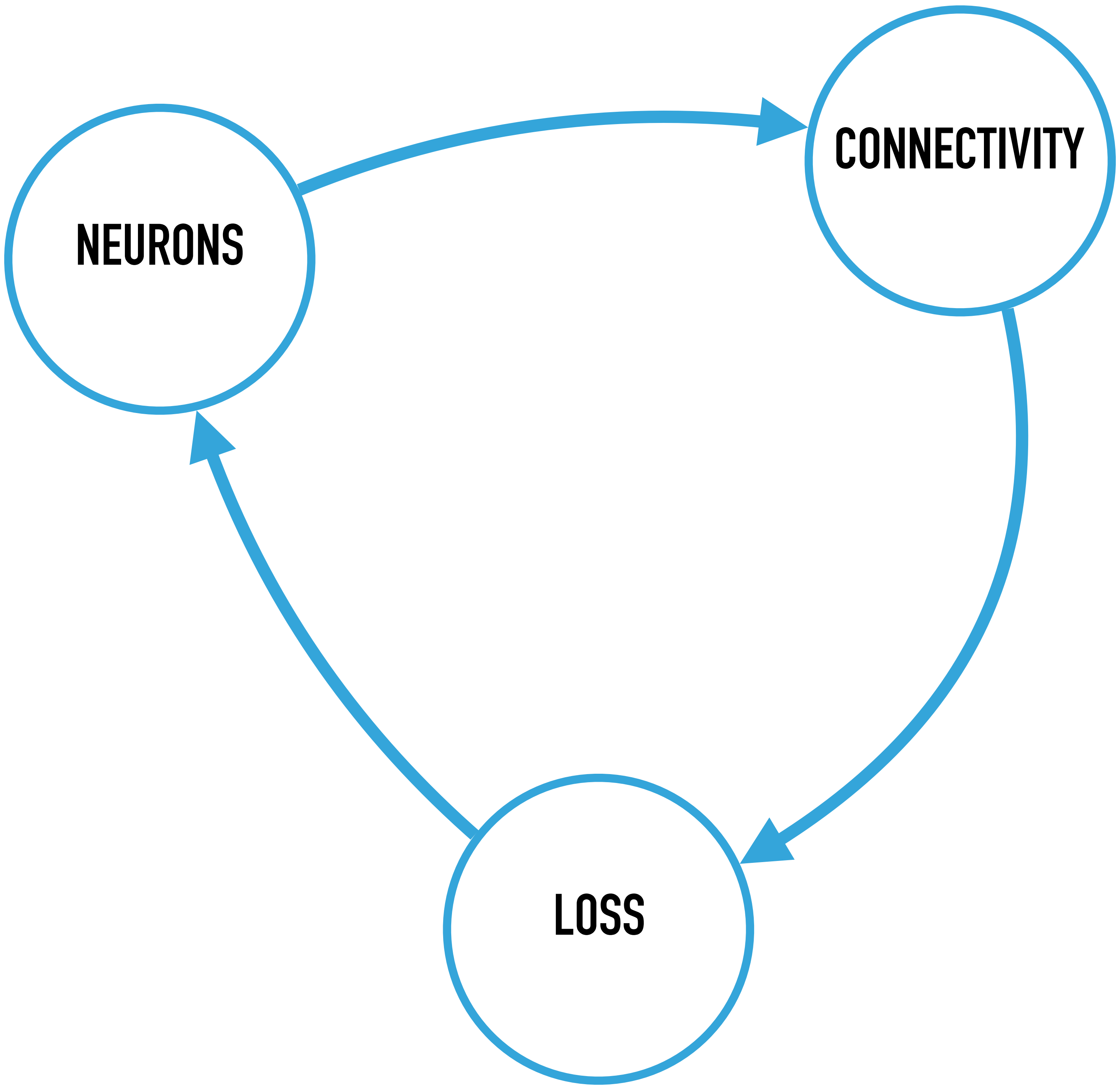
CLASSICAL MACHINE LEARNING VS. DEEP LEARNING



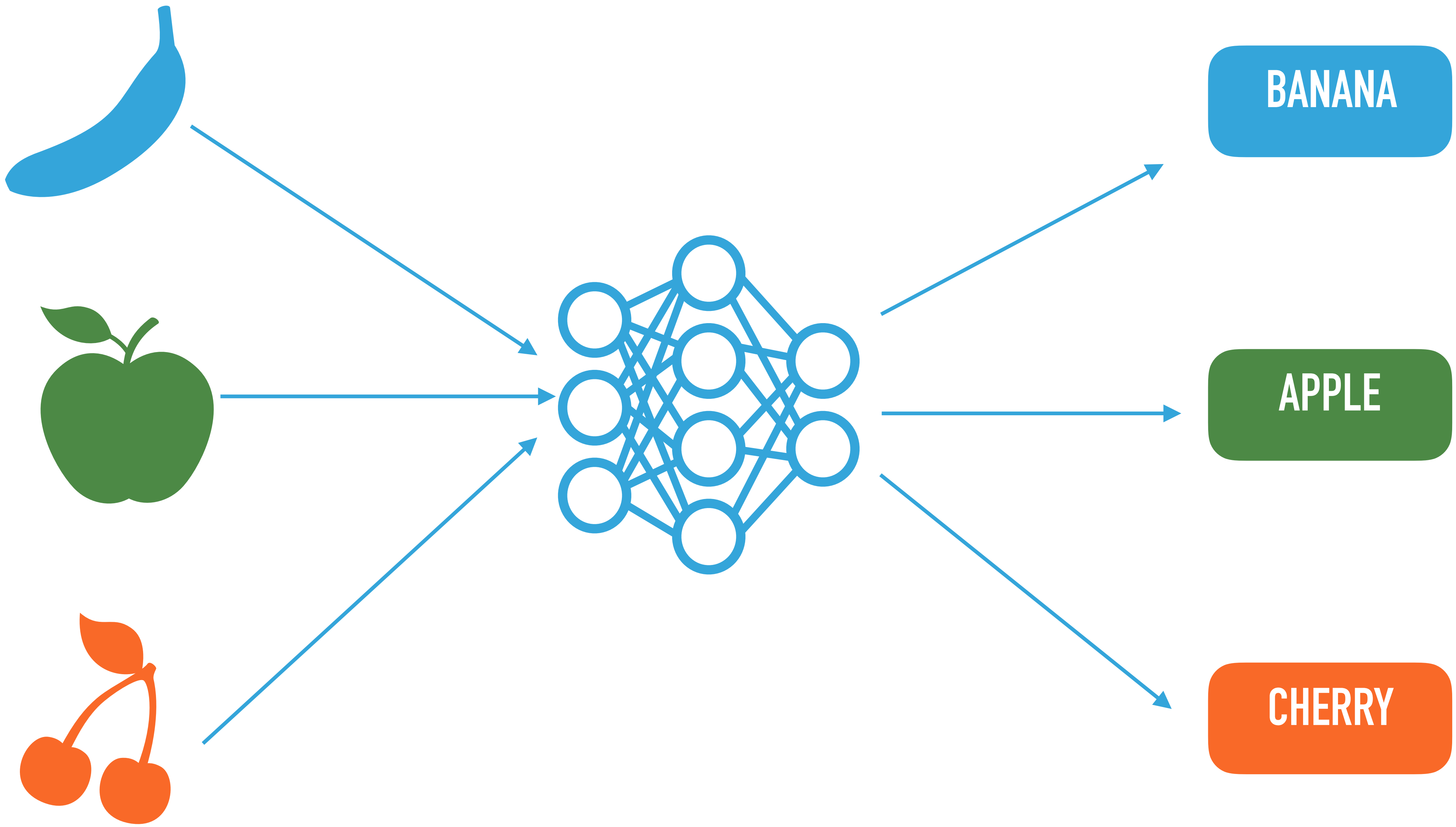
WHY DEEP LEARNING?



NEURAL NETWORKS : INGREDIENTS

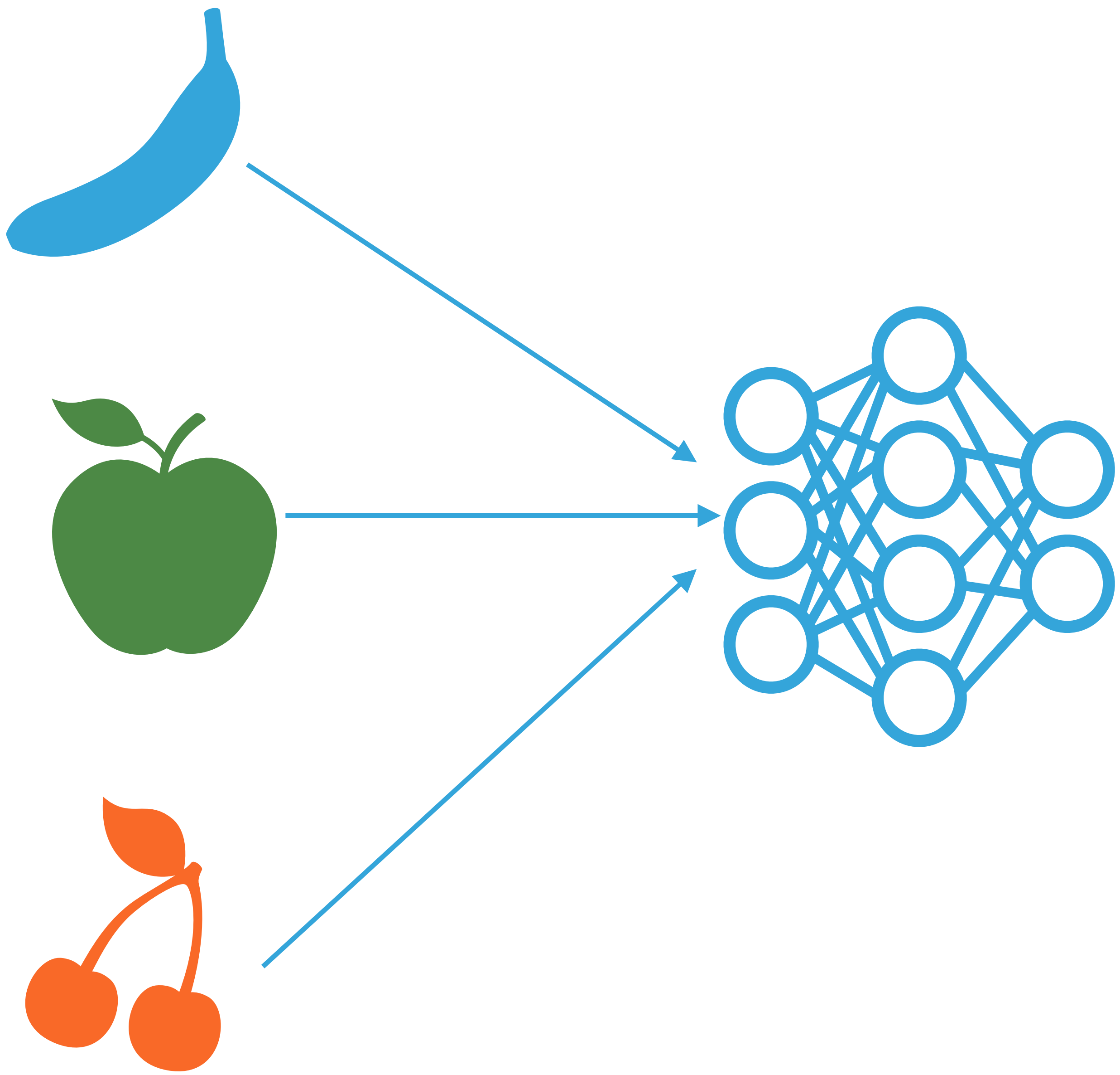


SUPERVISED TRAINING



Prediction, classification, regression, image2image, …

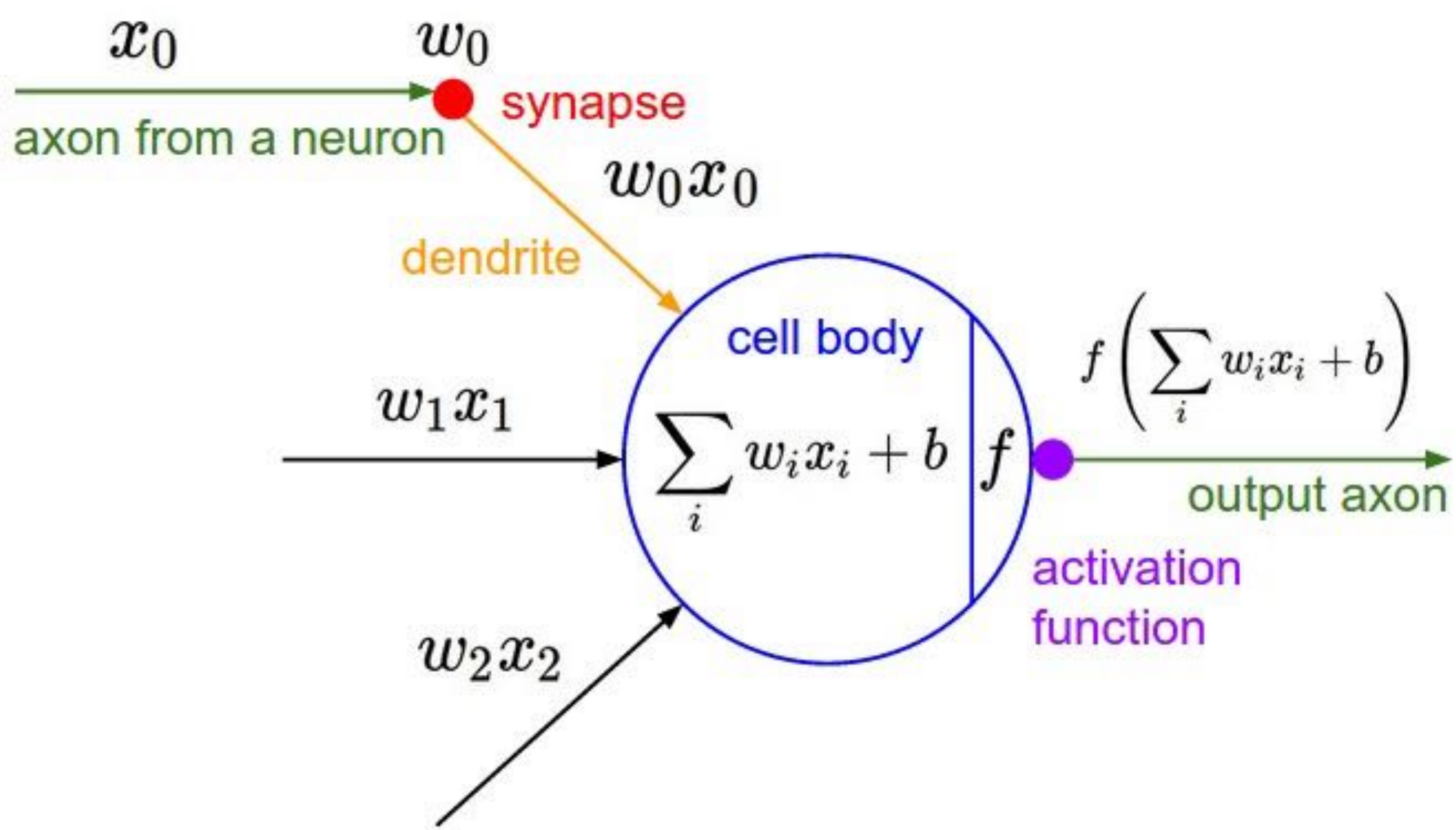
UNSUPERVISED TRAINING



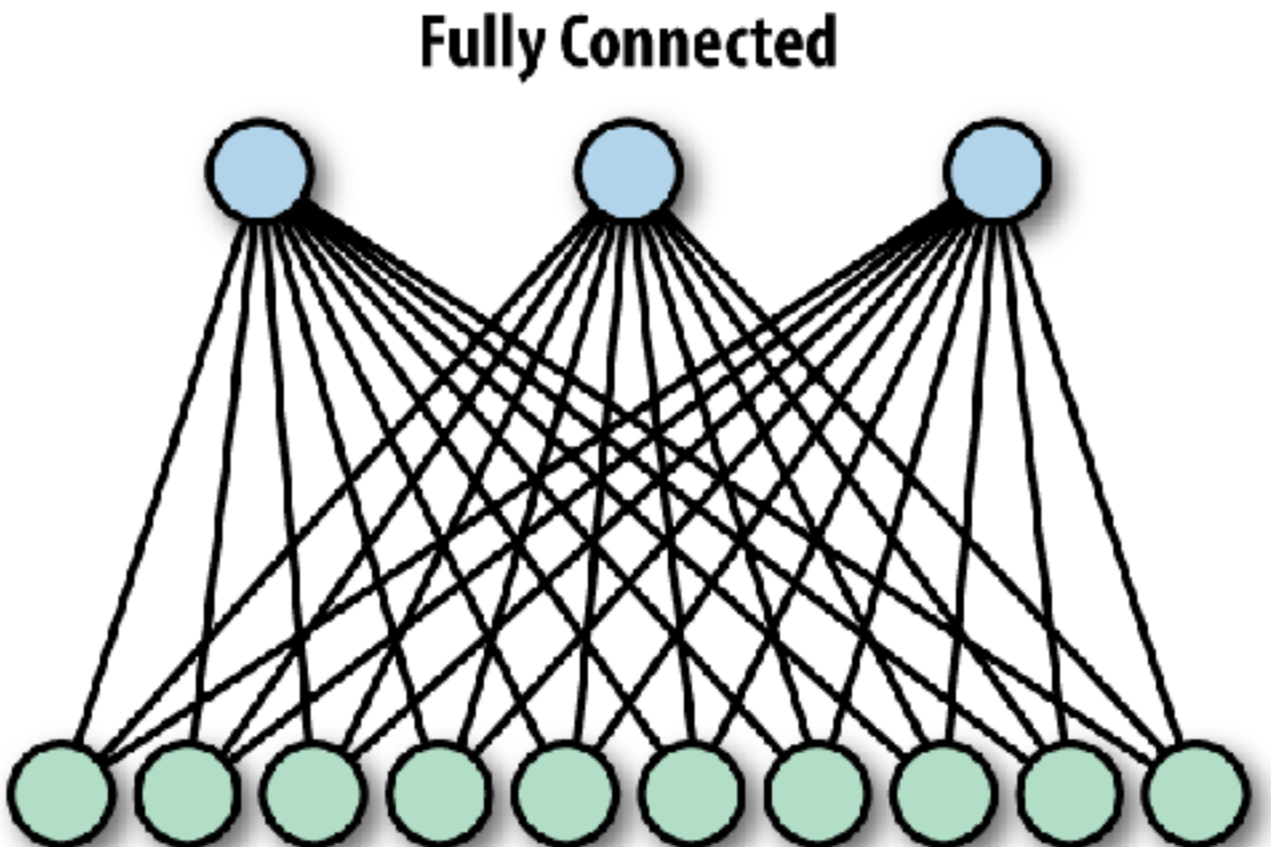
Clustering, feature extraction, generative models,…

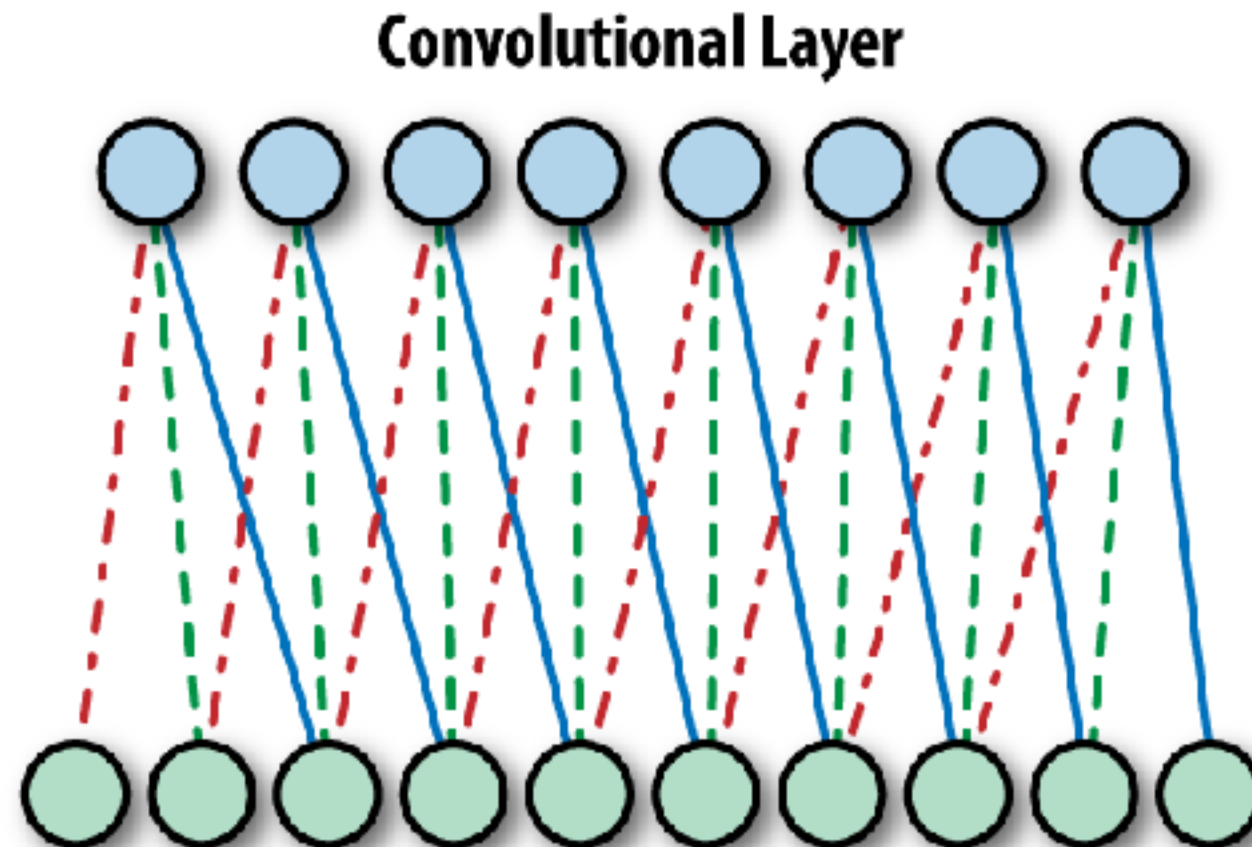
Arquitecture of a neural network

THE BASICS : A NEURON



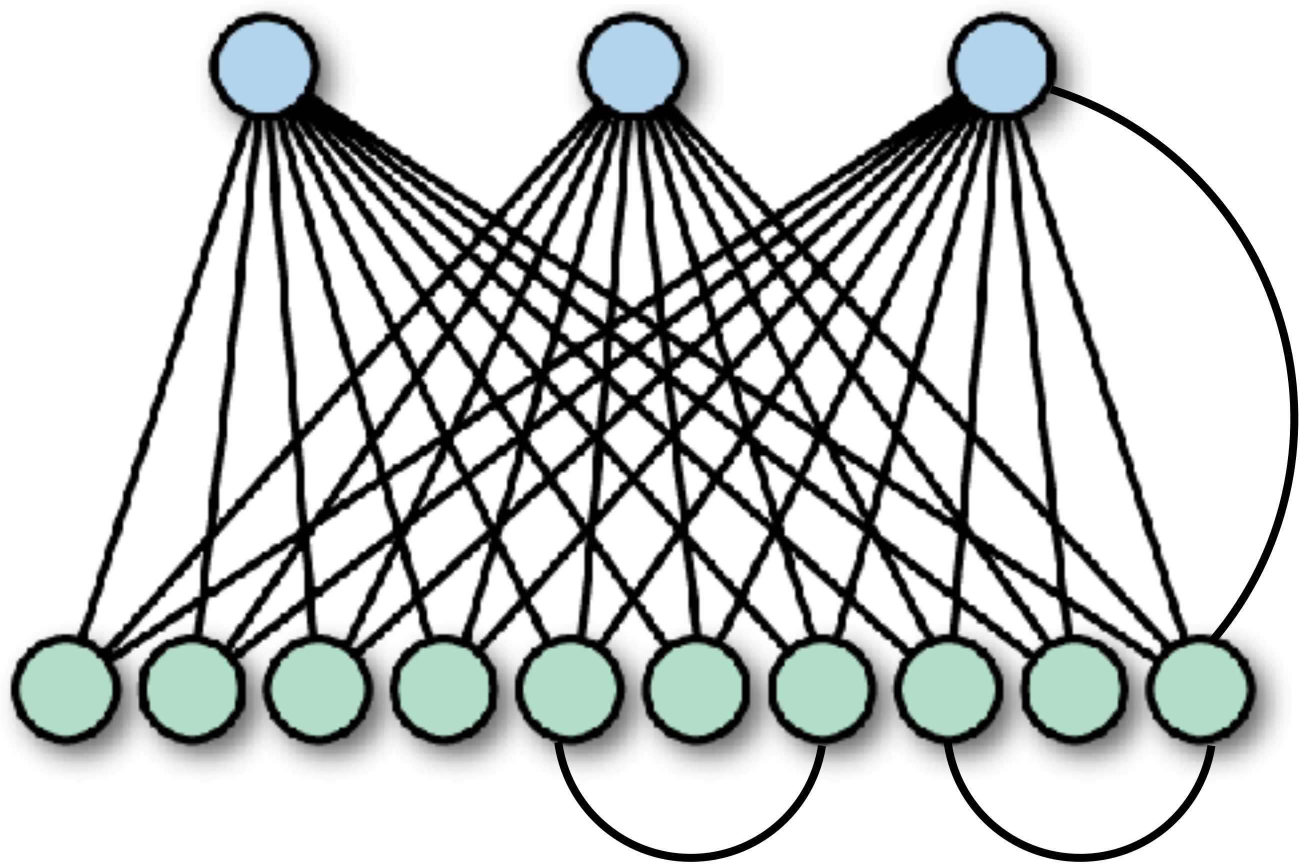
TYPES OF NEURAL NETWORKS



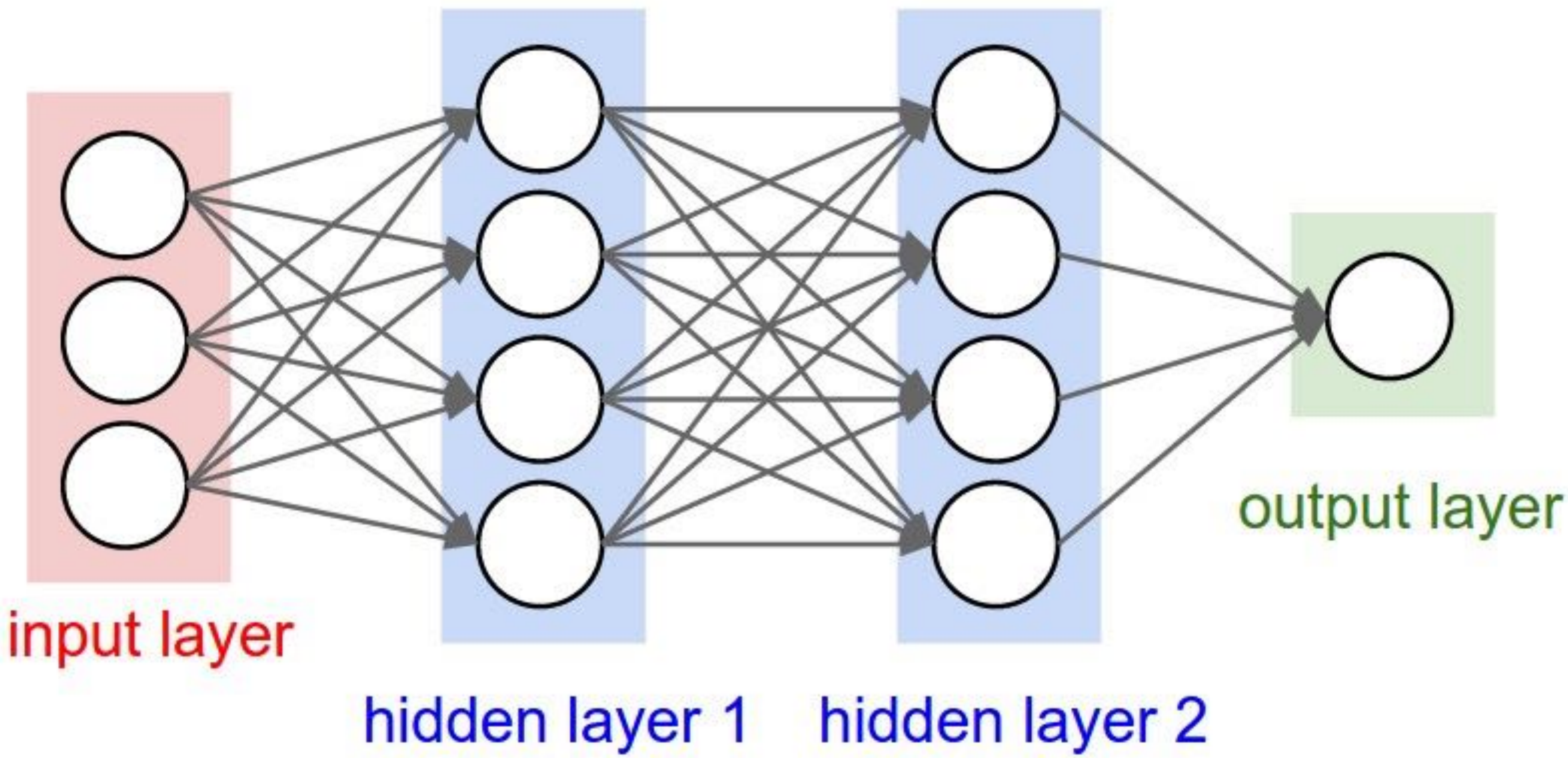


TYPES OF NEURAL NETWORKS

Recurrent network

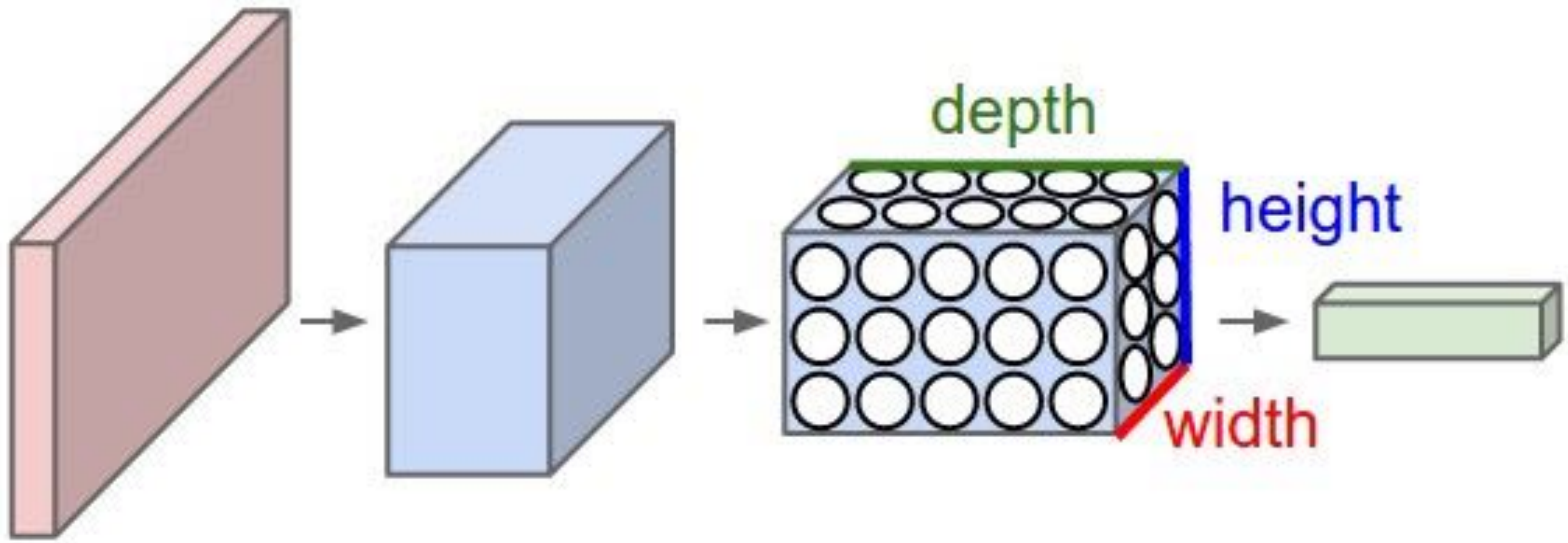


FULLY CONNECTED NEURAL NETWORK



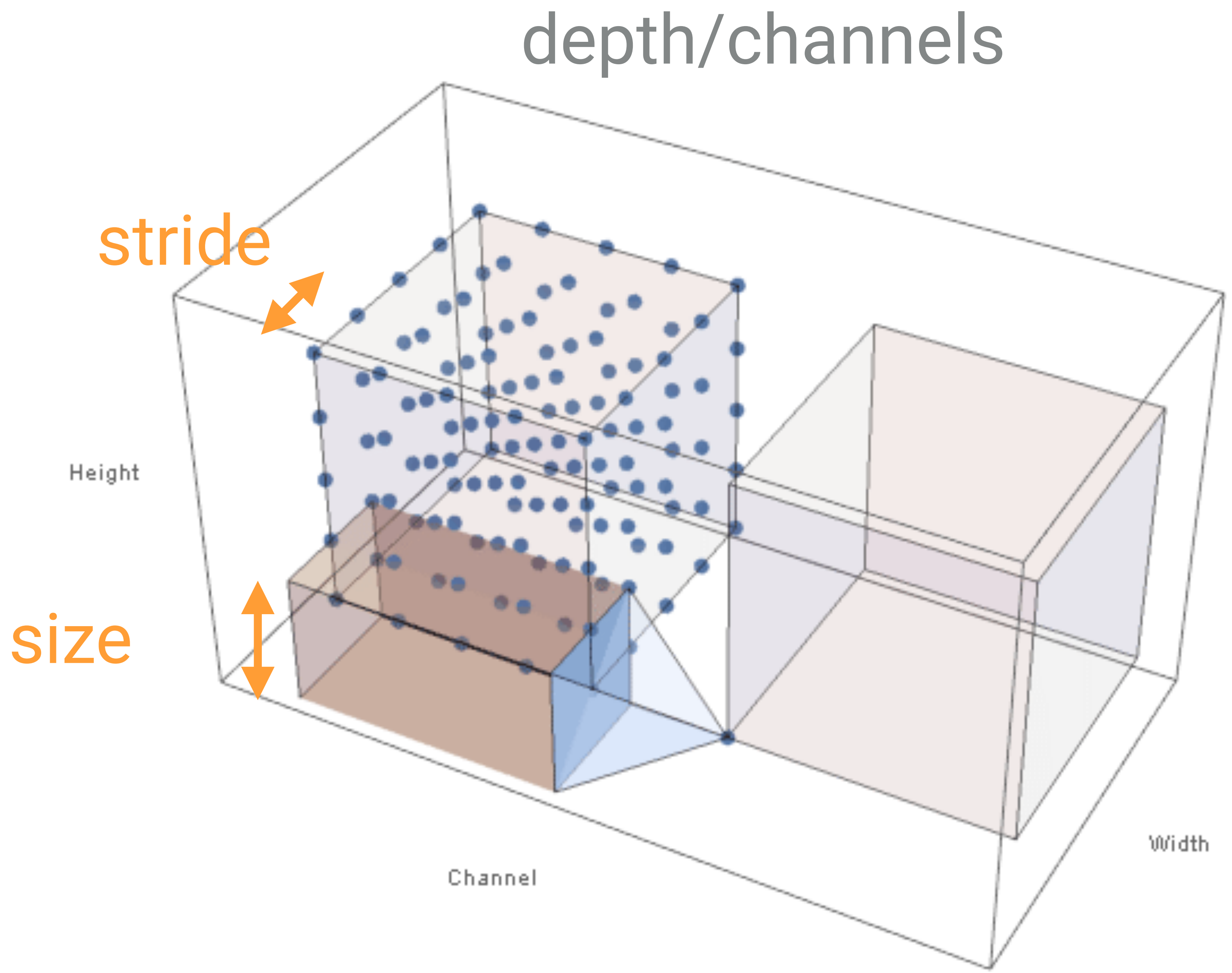
$$N = N_{\mathrm{in}} N_{\mathrm{hid1}} + N_{\mathrm{hid1}} N_{\mathrm{hid2}} + N_{\mathrm{hid2}} N_{\mathrm{out}}$$

CONVOLUTIONAL NEURAL NETWORK

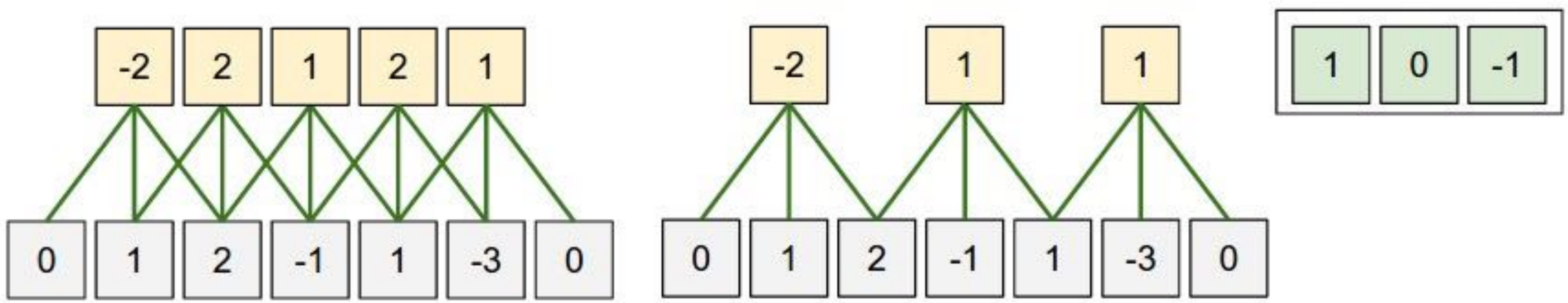


$$N = N_{\rm in} N_{\rm ker} d_{\rm ker}^2$$

CONVOLUTION



STRIDE



$$\frac{W - K + 2P}{S} + 1$$

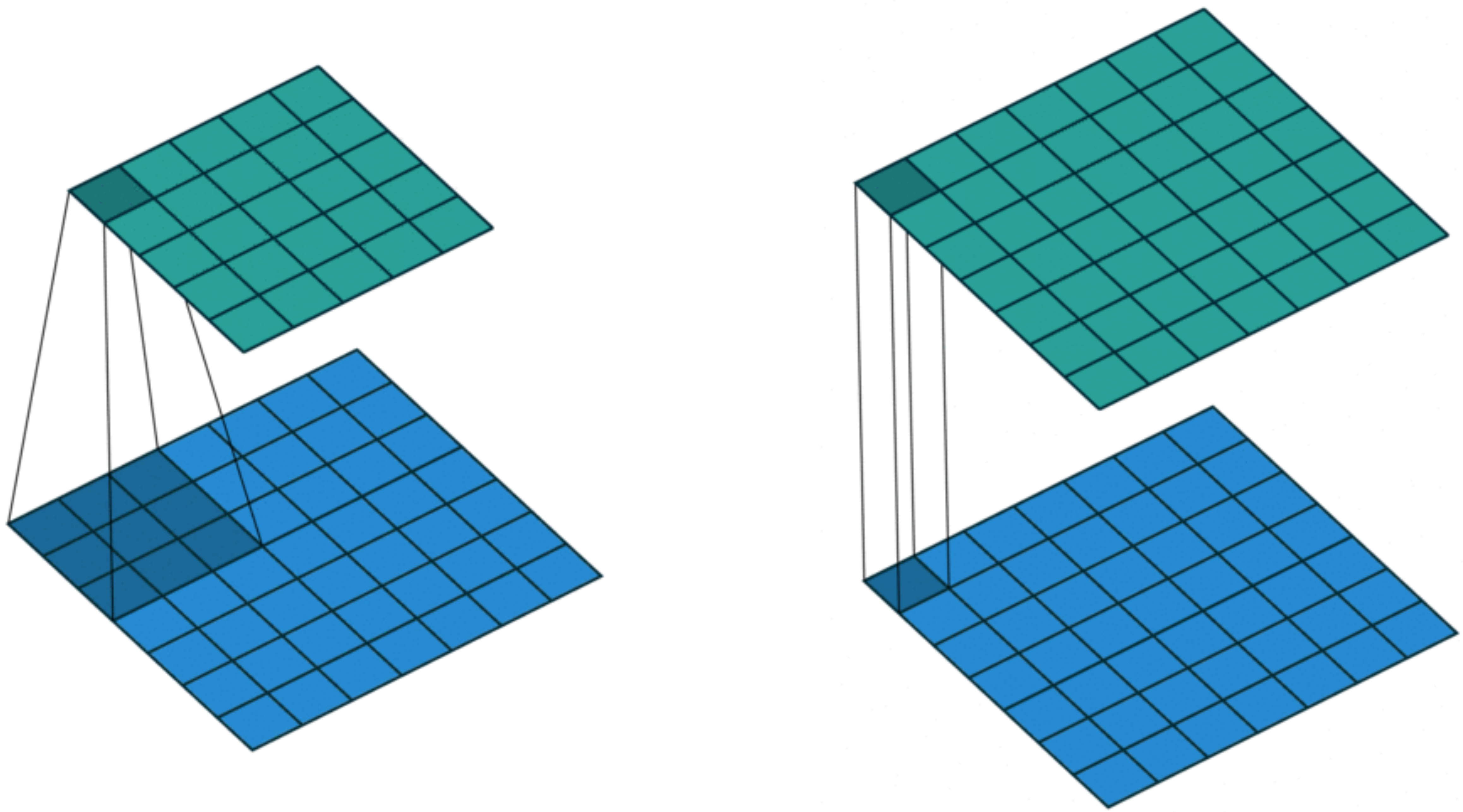
W: volume size

K: kernel size

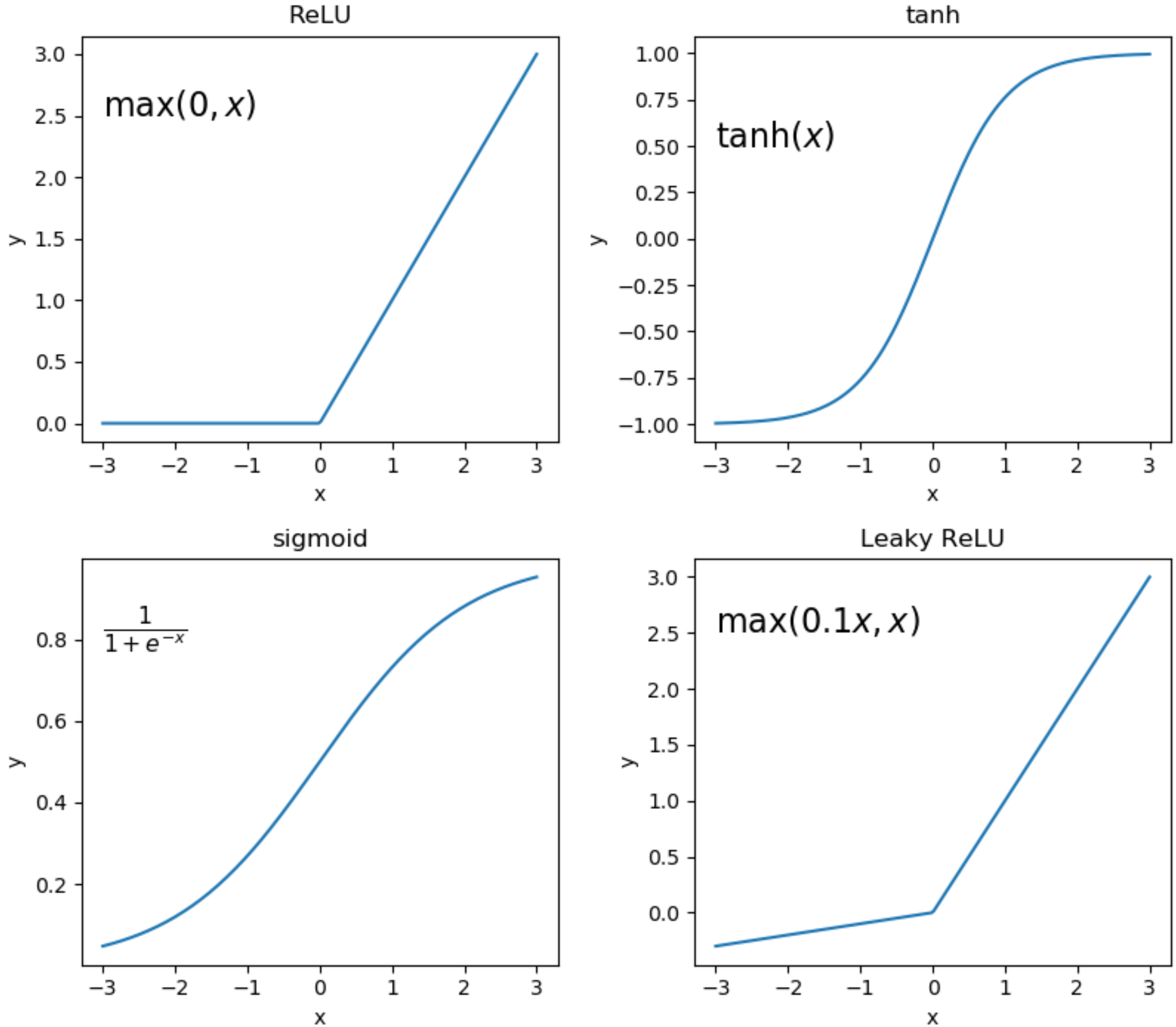
P: zero padding

S: stride

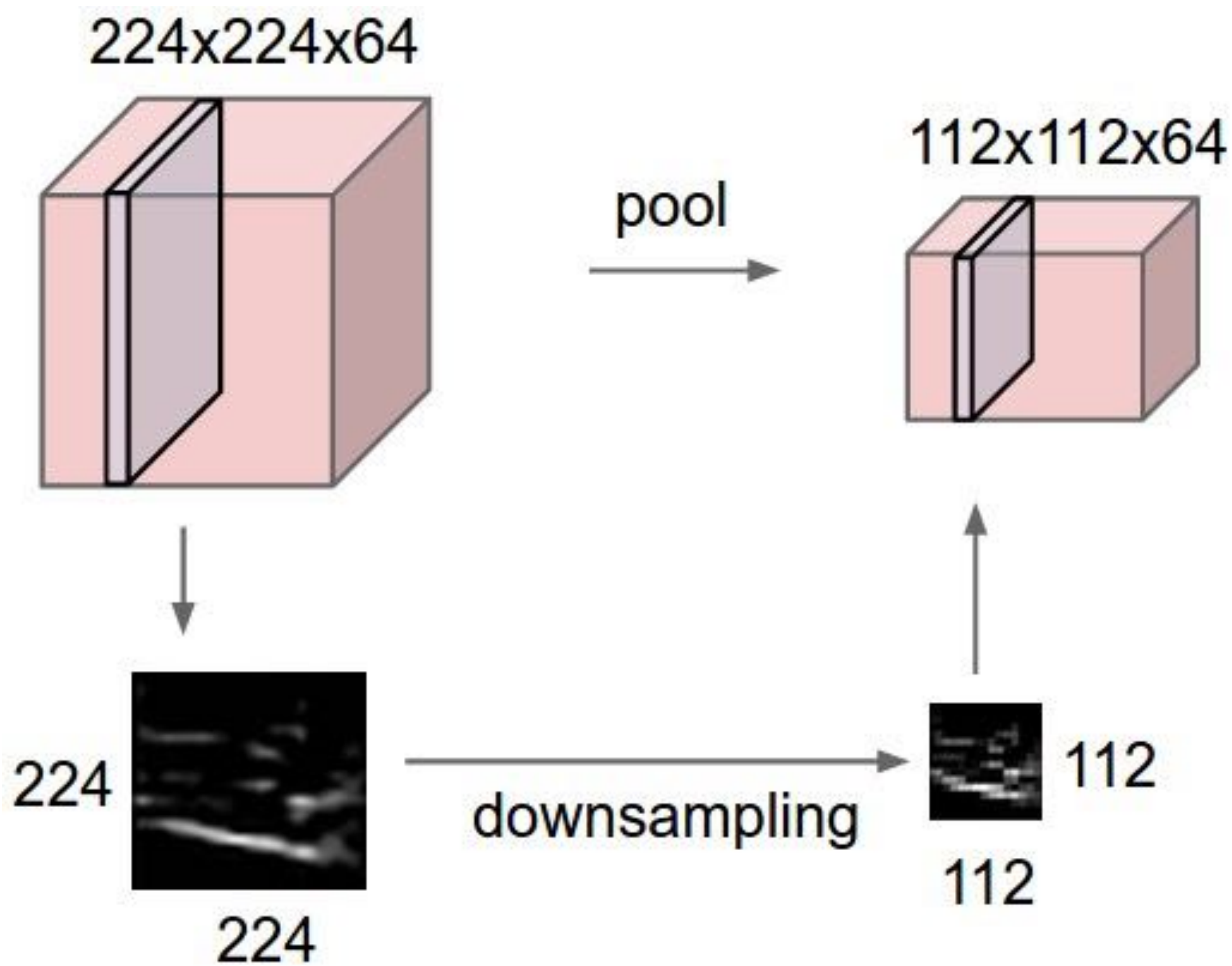
1X1 CONVOLUTION



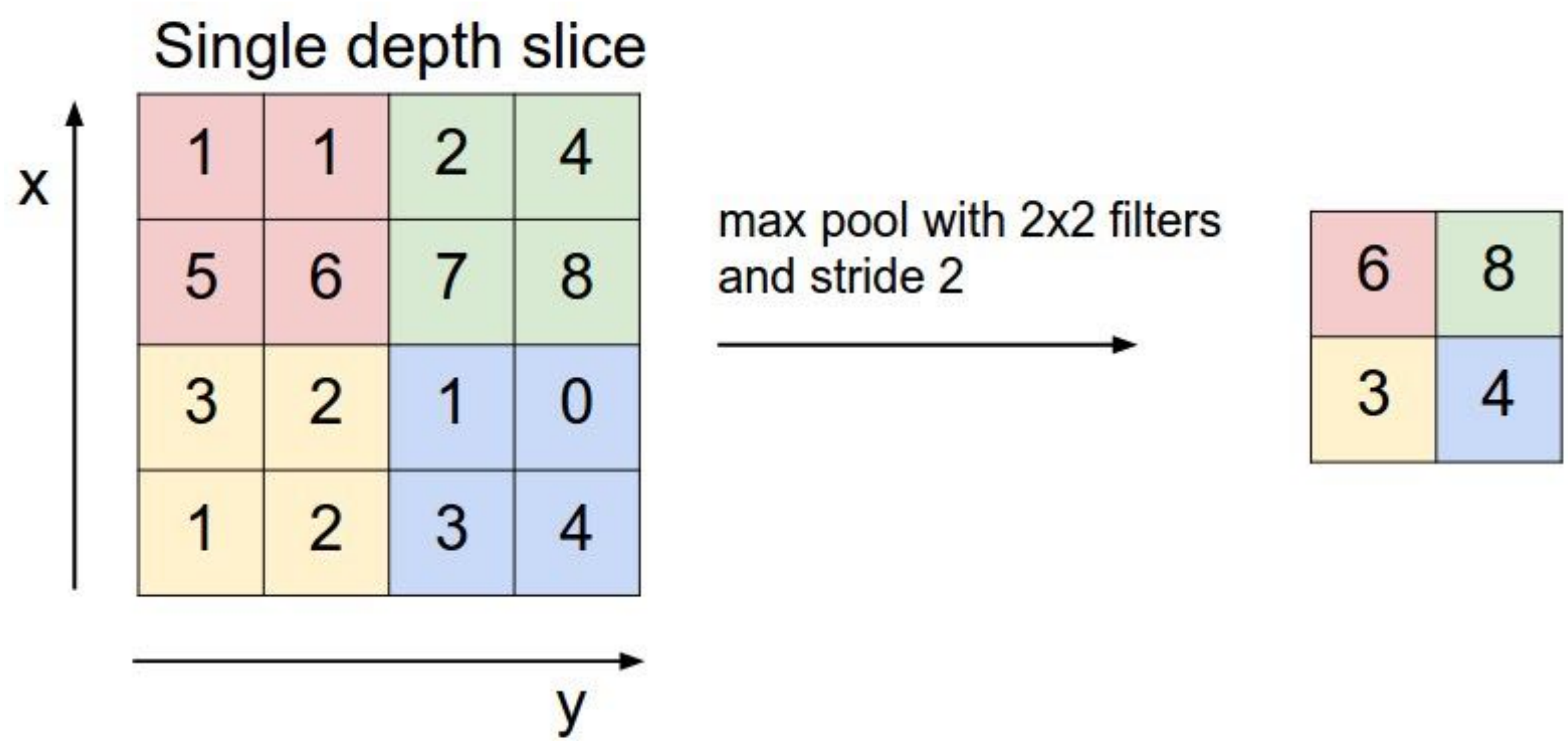
ACTIVATION FUNCTION



POOLING

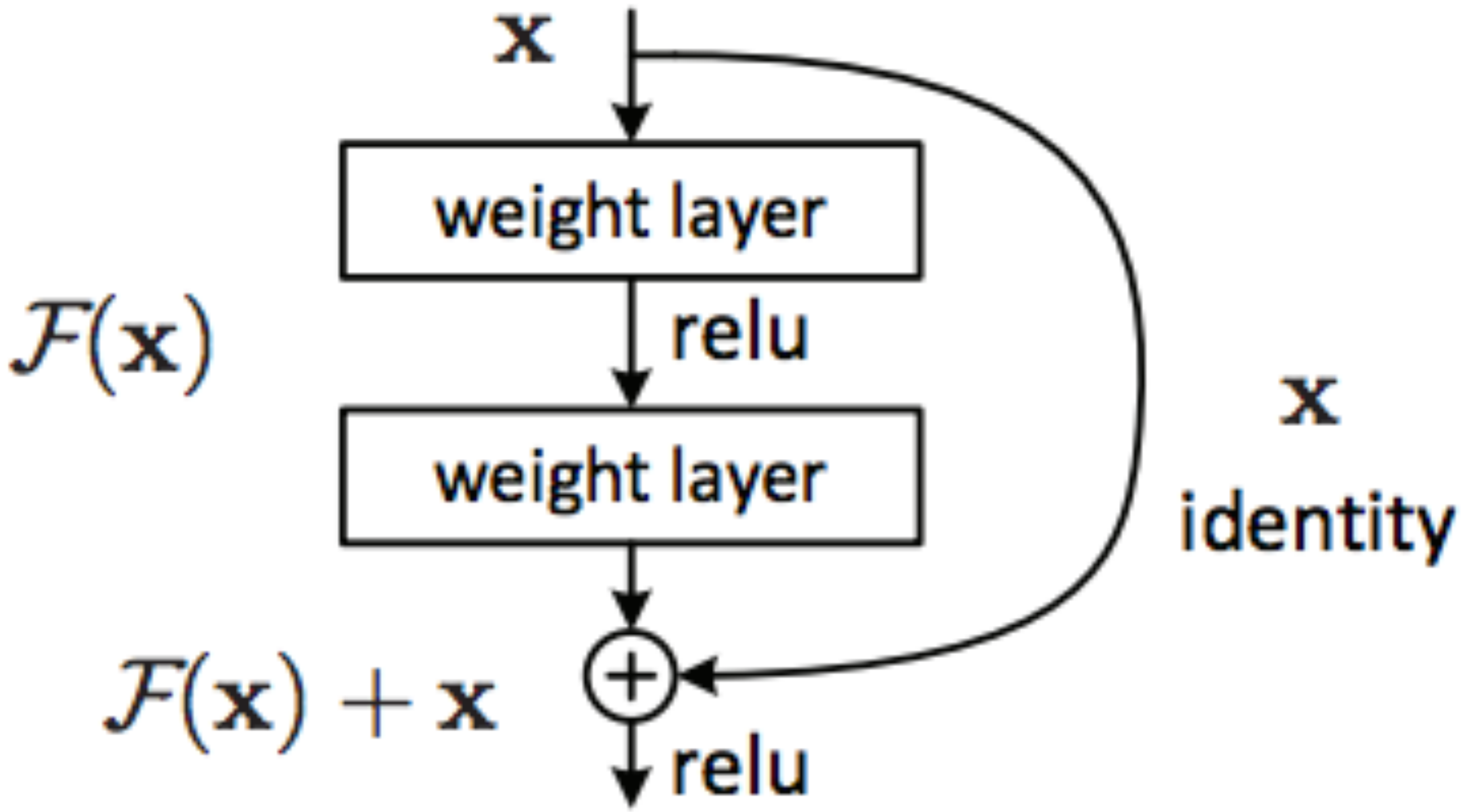


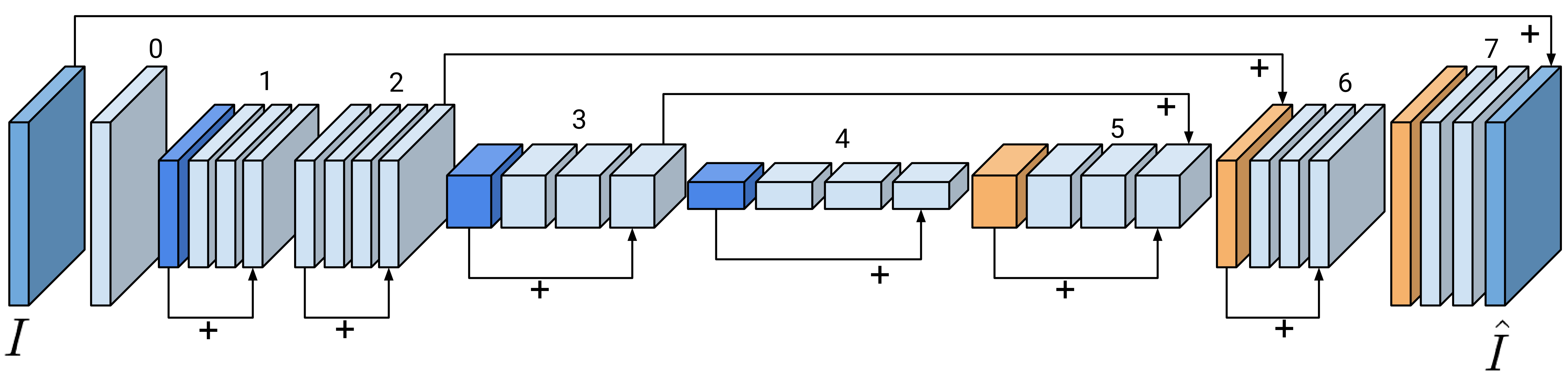
MAX-POOLING



Single depth slice
x
y
1 1 2 4
5 6 7 8
3 2 1 0
1 2 3 4
max pool with 2x2 filters
and stride 2
6 8
3 4

RESIDUAL CONNECTION





BATCH NORMALIZATION

Input: Values of x over a mini-batch: $\mathcal{B} = \{x_{1\ldots m}\}$;
Parameters to be learned: γ, β

Output: $\{y_i = \mathrm{BN}_{\gamma,\beta}(x_i)\}$

$$\mu_\mathcal{B} \leftarrow \frac{1}{m}\sum_{i=1}^{m} x_i \qquad \text{// mini-batch mean}$$

$$\sigma_\mathcal{B}^2 \leftarrow \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_\mathcal{B})^2 \qquad \text{// mini-batch variance}$$

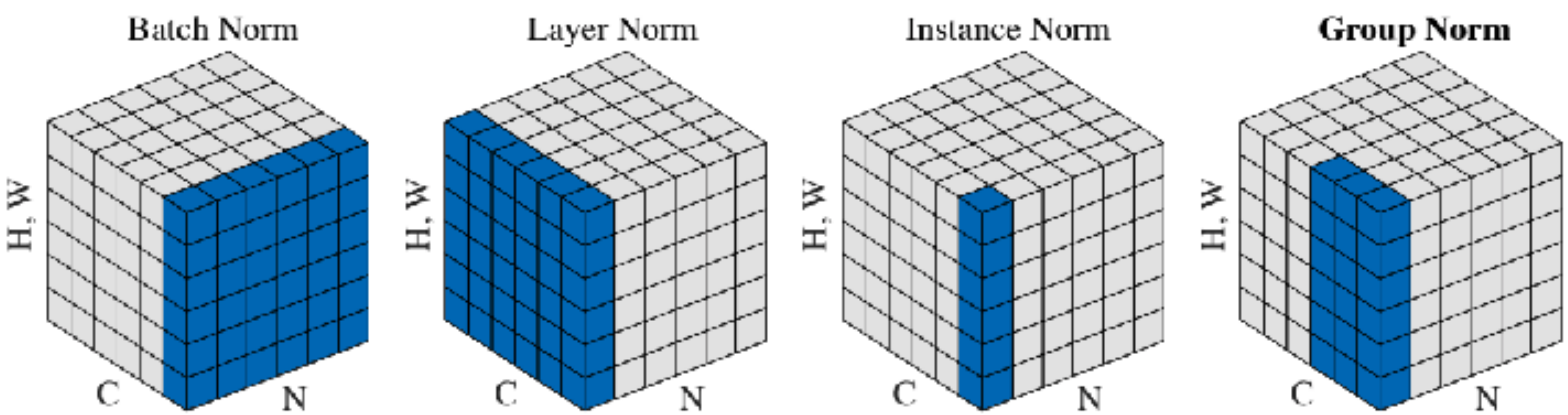
$$\widehat{x}_i \leftarrow \frac{x_i - \mu_\mathcal{B}}{\sqrt{\sigma_\mathcal{B}^2 + \epsilon}} \qquad \text{// normalize}$$

$$y_i \leftarrow \gamma\widehat{x}_i + \beta \equiv \mathrm{BN}_{\gamma,\beta}(x_i) \qquad \text{// scale and shift}$$

Algorithm 1: Batch Normalizing Transform, applied to activation x over a mini-batch.

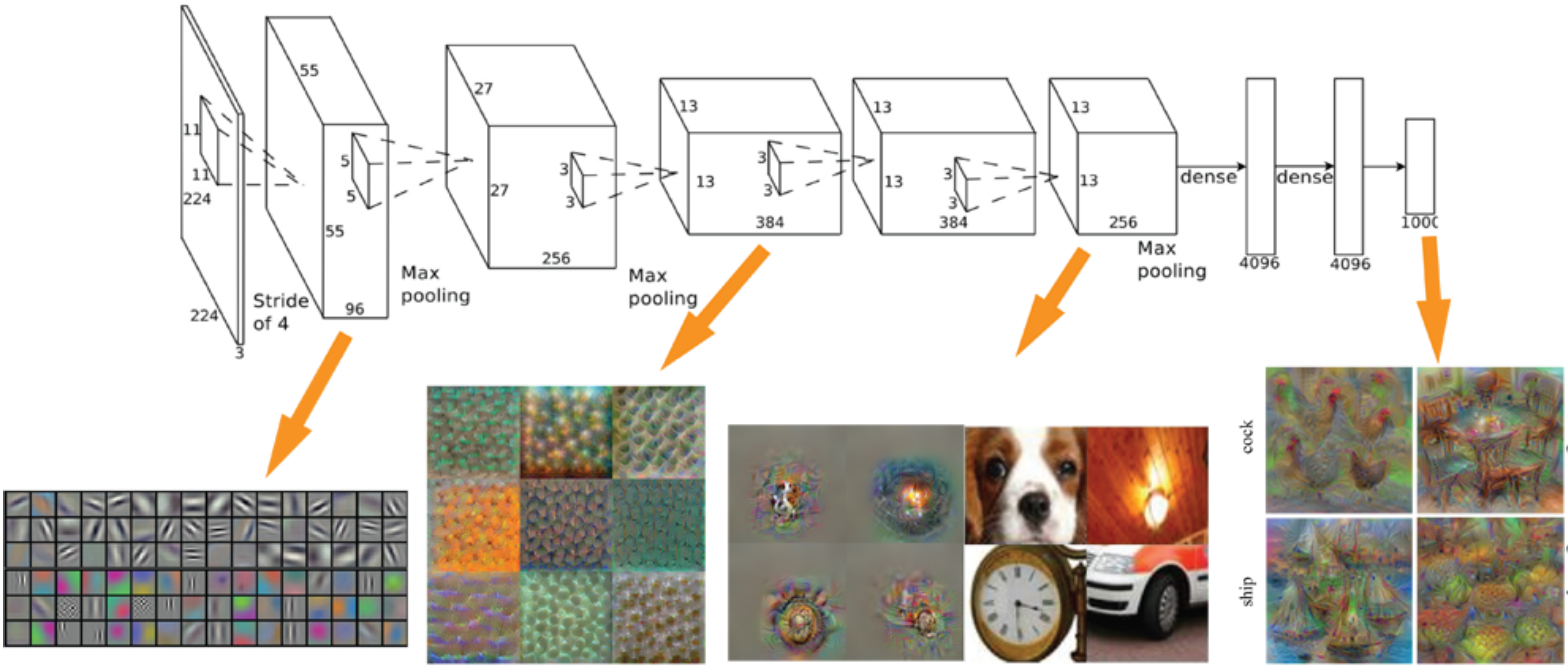
Ioffe & Szegedy (2015)

BATCH NORMALIZATION



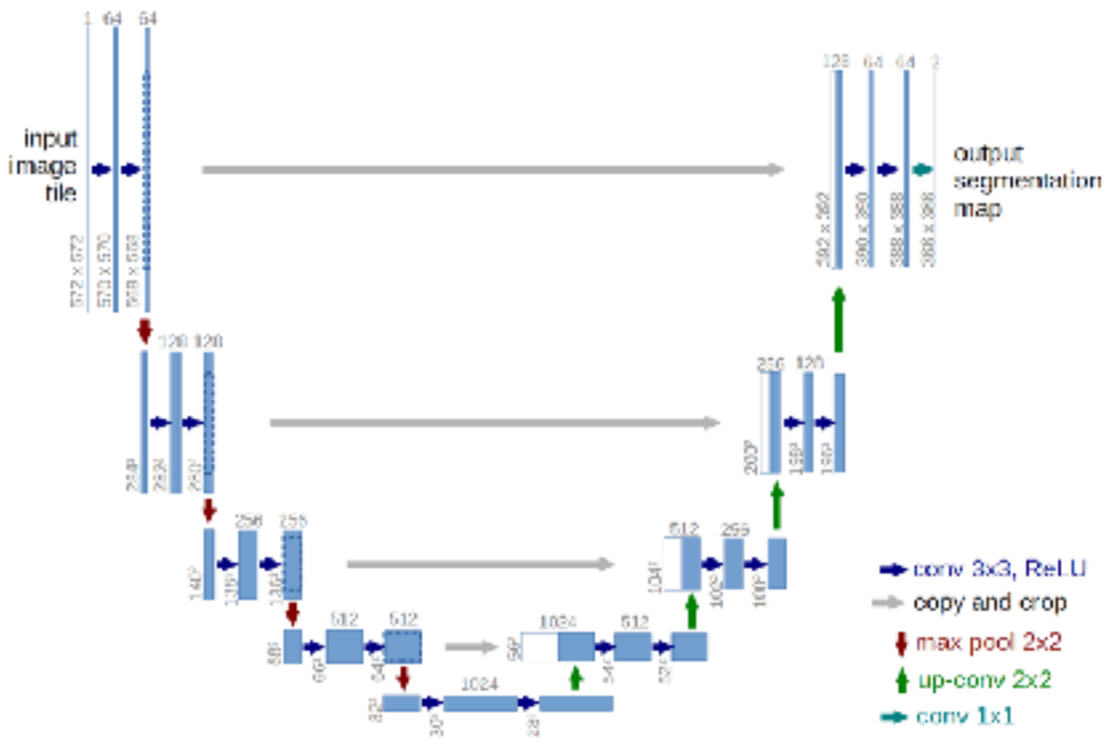
Wu & He (2018)

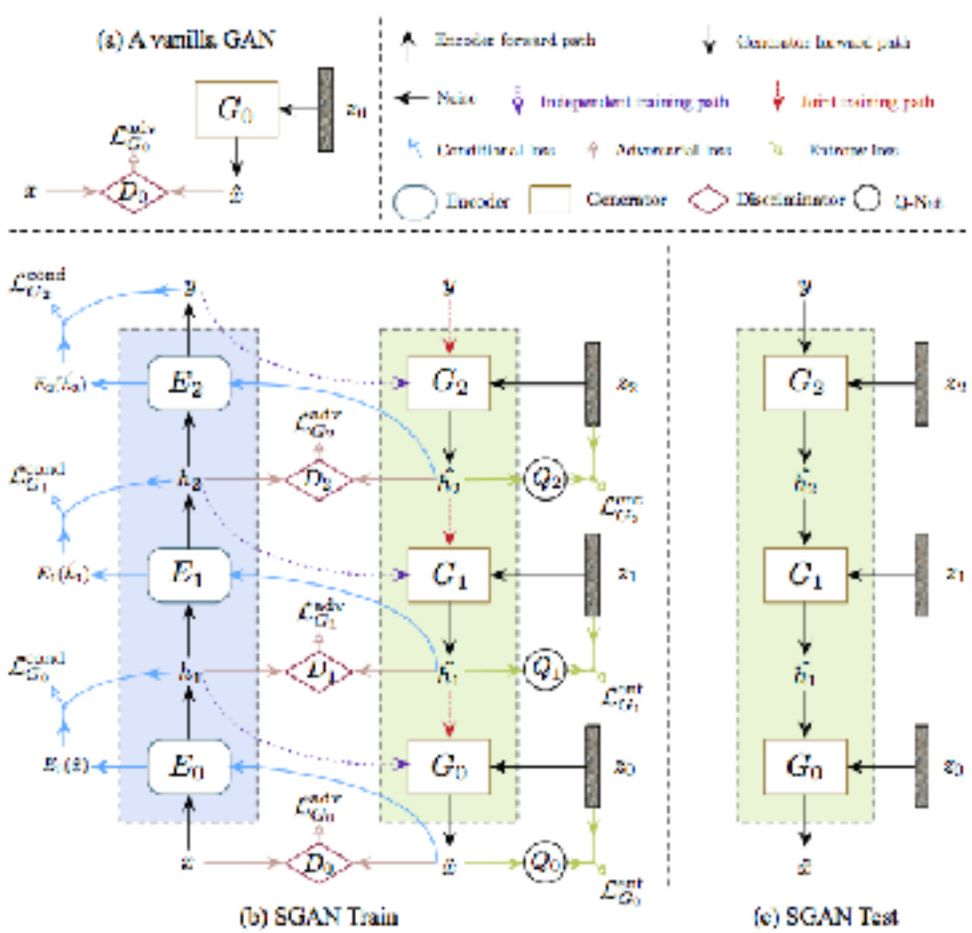
MULTISCALE ANALYSIS

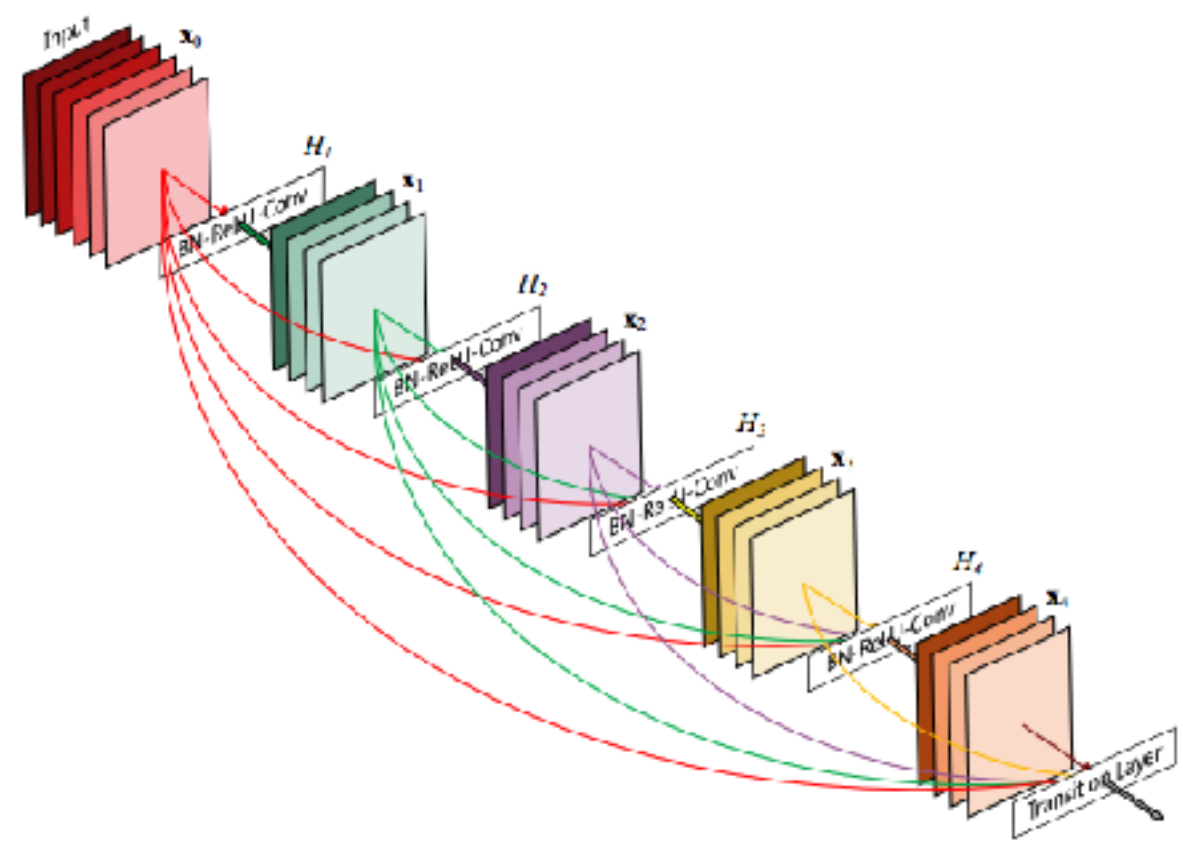


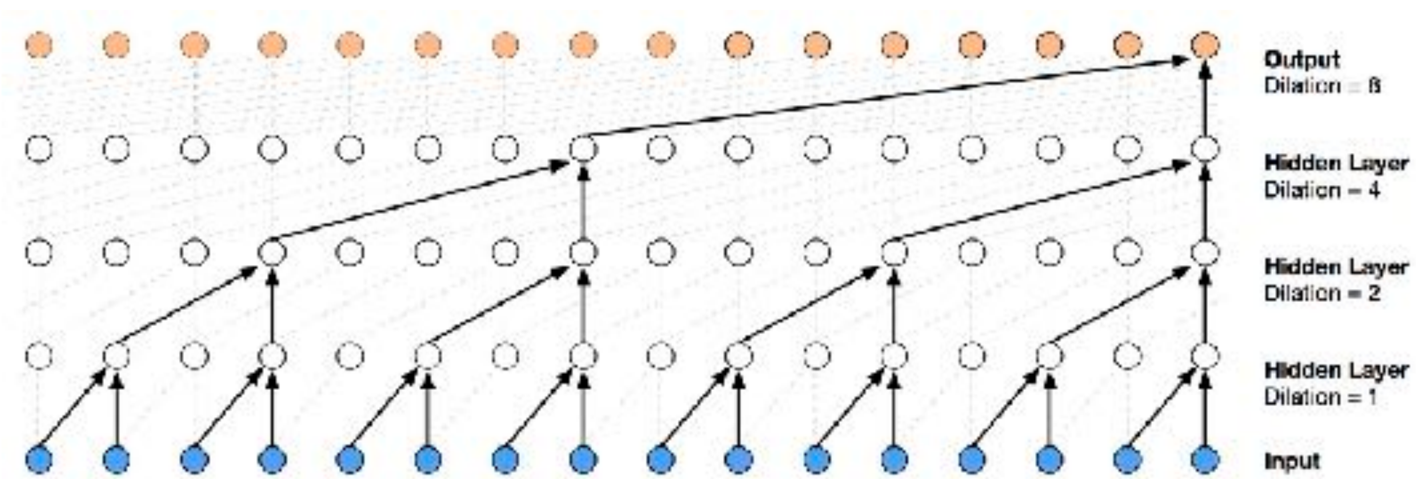
Conv 1: Edge+Blob
Conv 3: Texture
Conv 5: Object Parts
Fc8: Object Classes

ENORMOUS LANDSCAPE









TWO RULES TO DECIDE THE ARCHITECTURE

Read a lot!
Still not in books: arxiv!

Experiment a lot!

Training of a neural network

LOSS FUNCTIONS

Mean squared error

$$L = \frac{\sum_{i=1}^{n} \left(y_i - \hat{y}_i\right)^2}{n}$$

Mean absolute error

$$L = \frac{\sum_{i=1}^{n} \left|y_i - \hat{y}_i\right|}{n}$$

Cross-entropy

$$L = -\left(y_i \log\left(\hat{y}_i\right) + (1 - y_i)\log\left(1 - \hat{y}_i\right)\right)$$

TRAINING: USE THE SIMPLEST YOU CAN THINK OF

Gradient descent

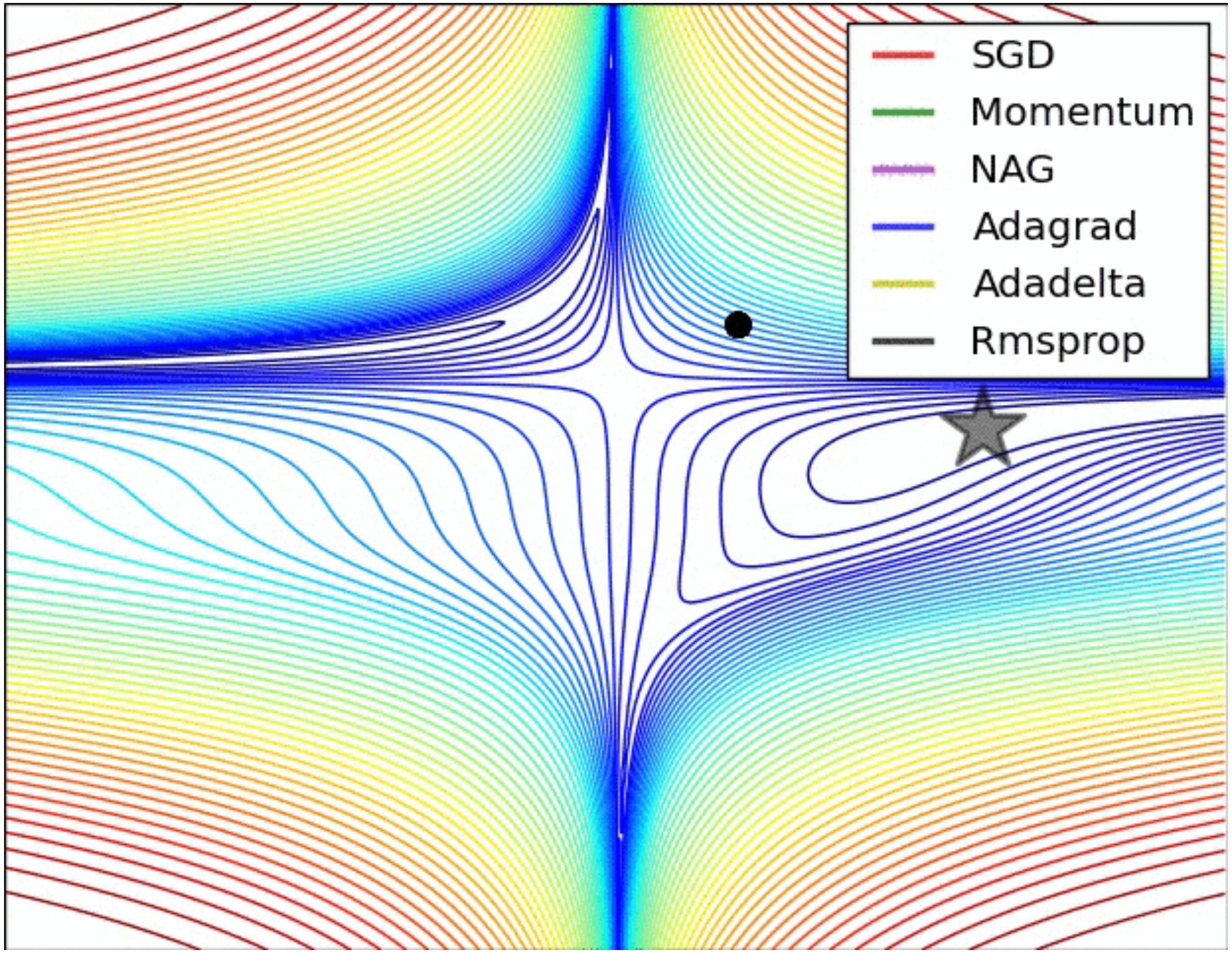
$$\theta_{i+1} = \theta_i - h\nabla_\theta f(\theta, \mathbf{T})$$

Stochastic gradient descent

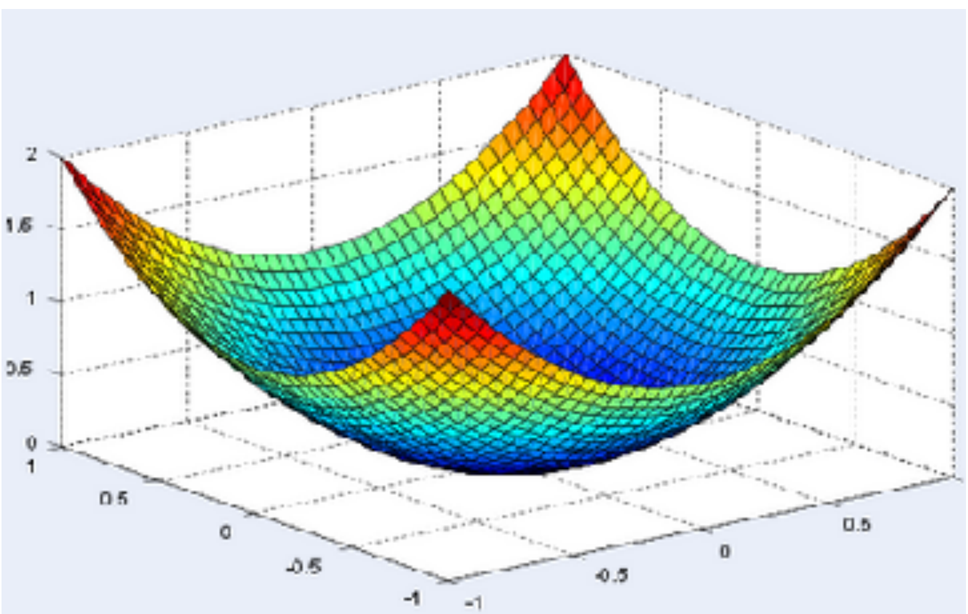
$$\theta_{i+1} = \theta_i - h\nabla_\theta f(\theta, \mathbf{T}_{\text{subset}})$$

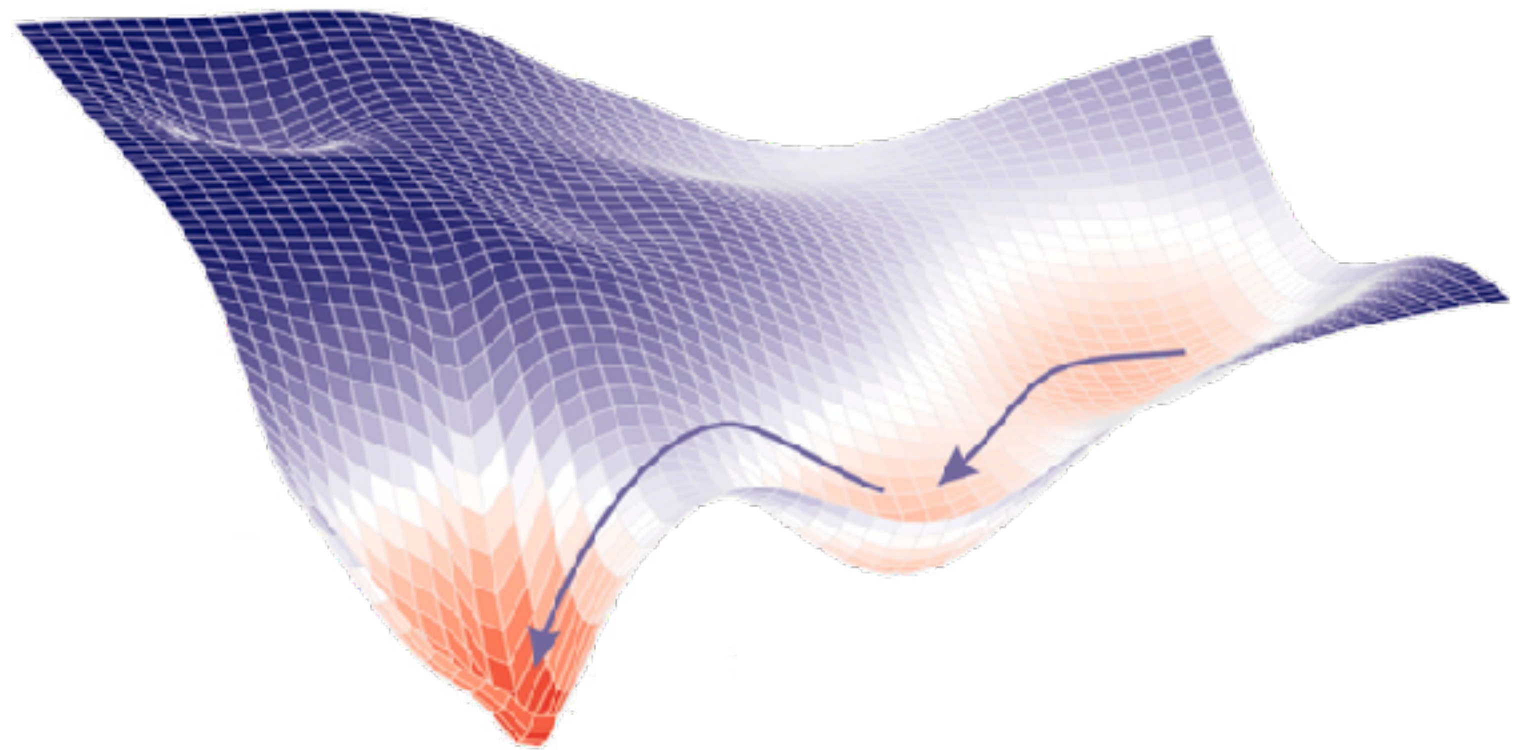
TRAINING

$$\theta_{i+1} = \theta_i - h\nabla_\theta f(\theta, \mathbf{T}_{\text{subset}})$$

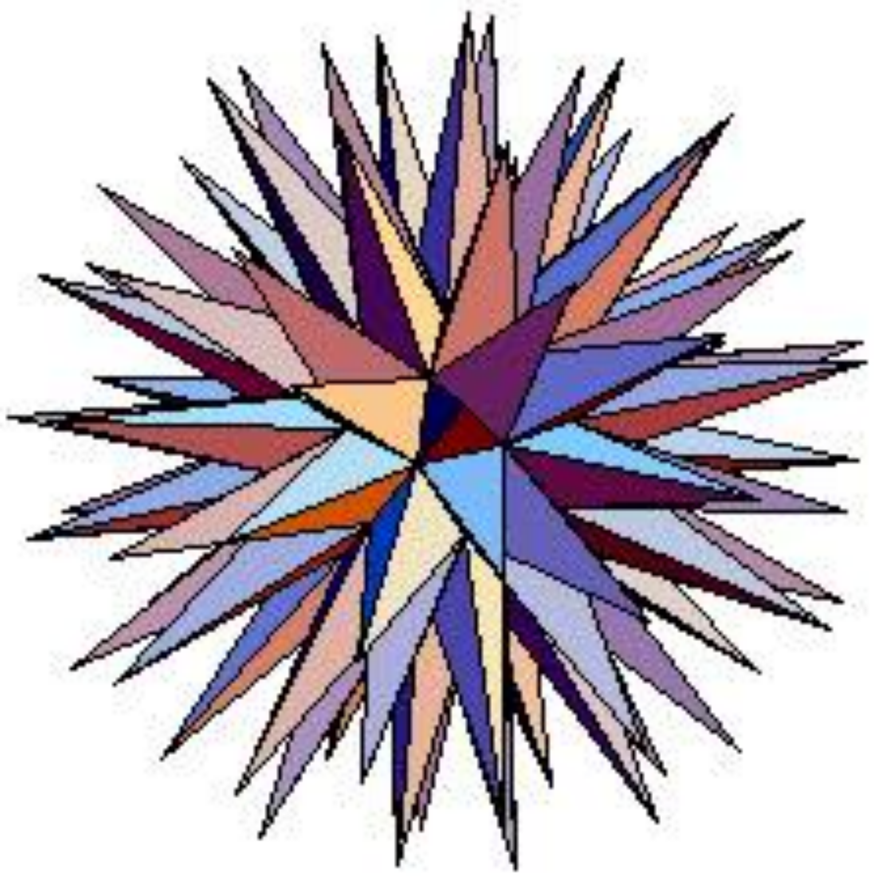


CONVEXITY VS. NON-CONVEXITY





CURSE OF DIMENSIONALITY



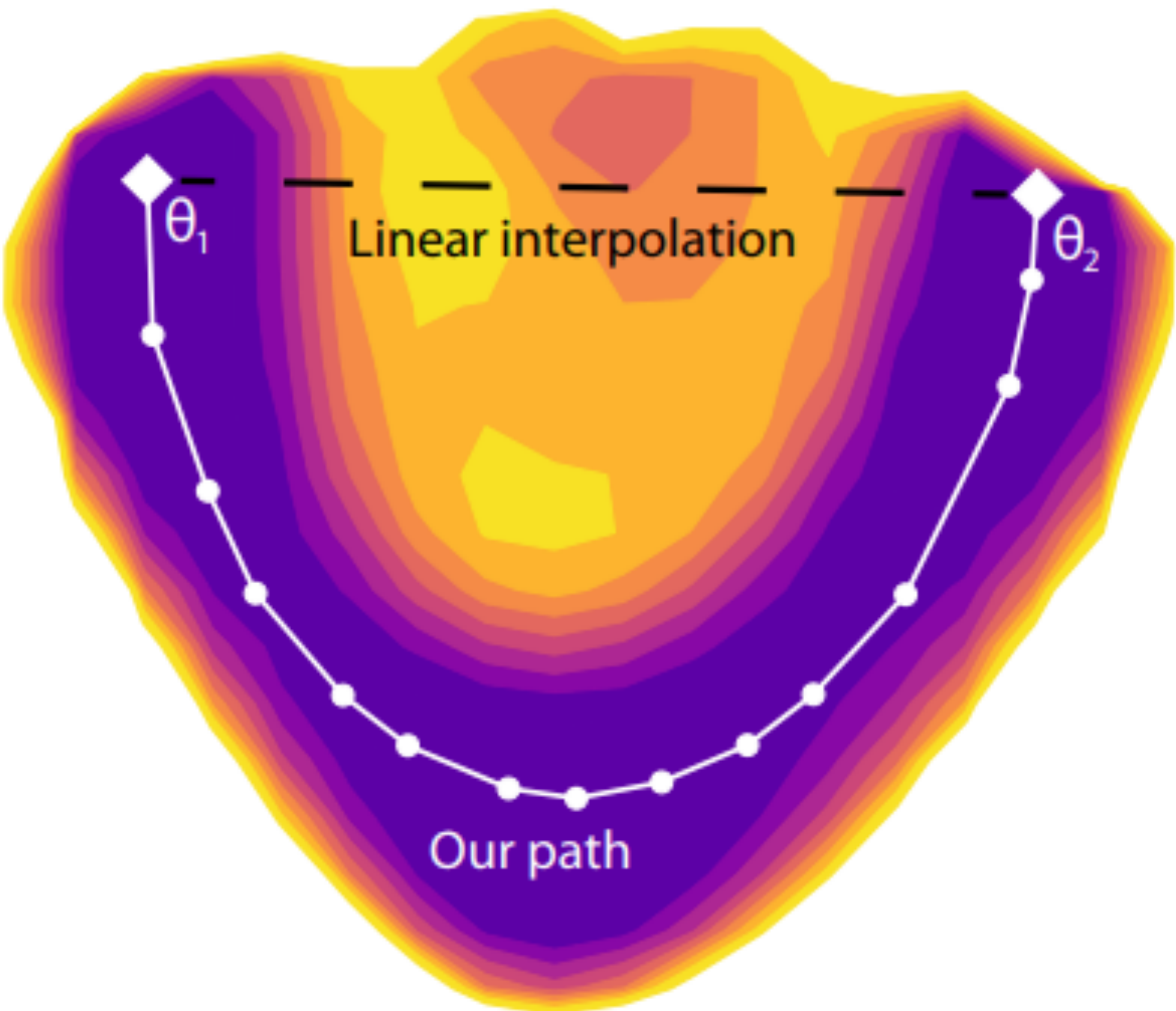
N. directions forming angles between 88 and 92 degrees

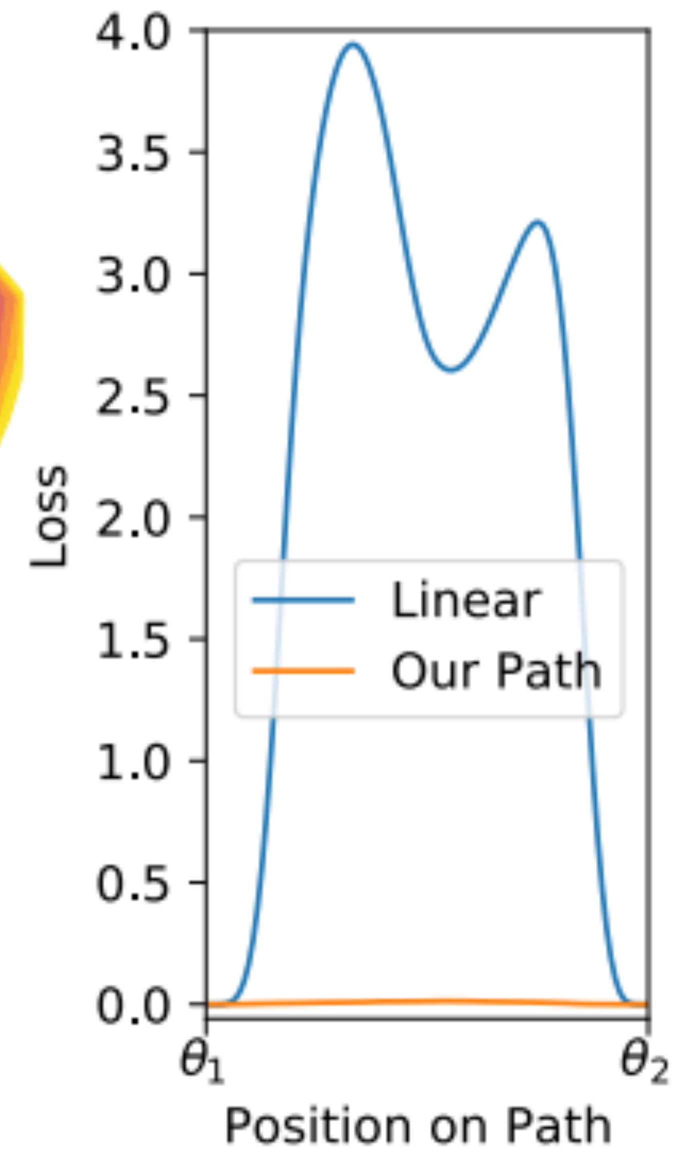
$$\mathbb{R}^2 \to 2$$

$$\mathbb{R}^3 \to 2$$

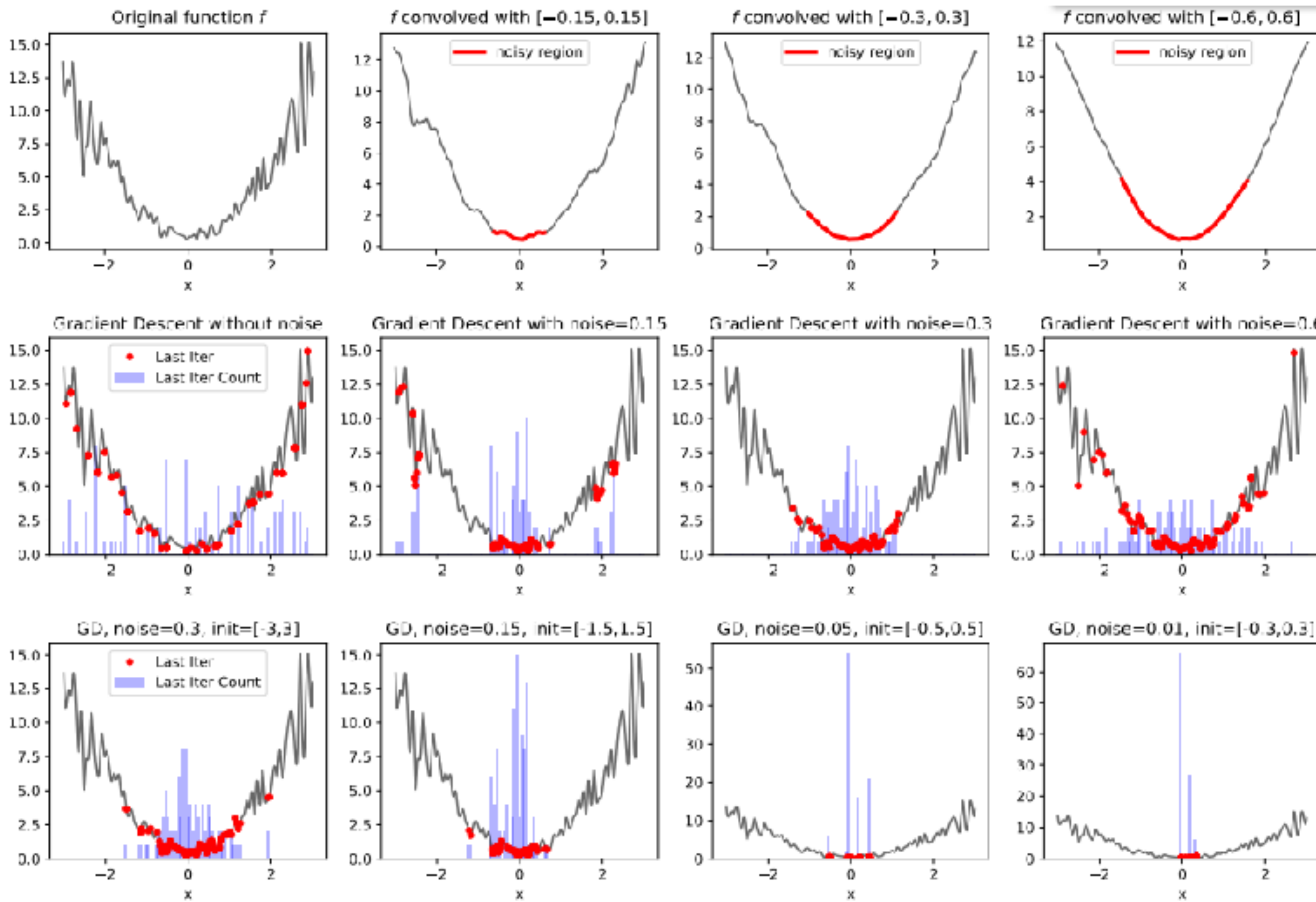
$$\mathbb{R}^d \to \exp(cd)$$

ALL MINIMA ARE EQUIVALENT





SGD MODIFIES THE LOSS FUNCTION



Backpropagation

HOW TO EFFICIENTLY COMPUTE THE GRADIENT

$$L = g(\mathbf{y})$$

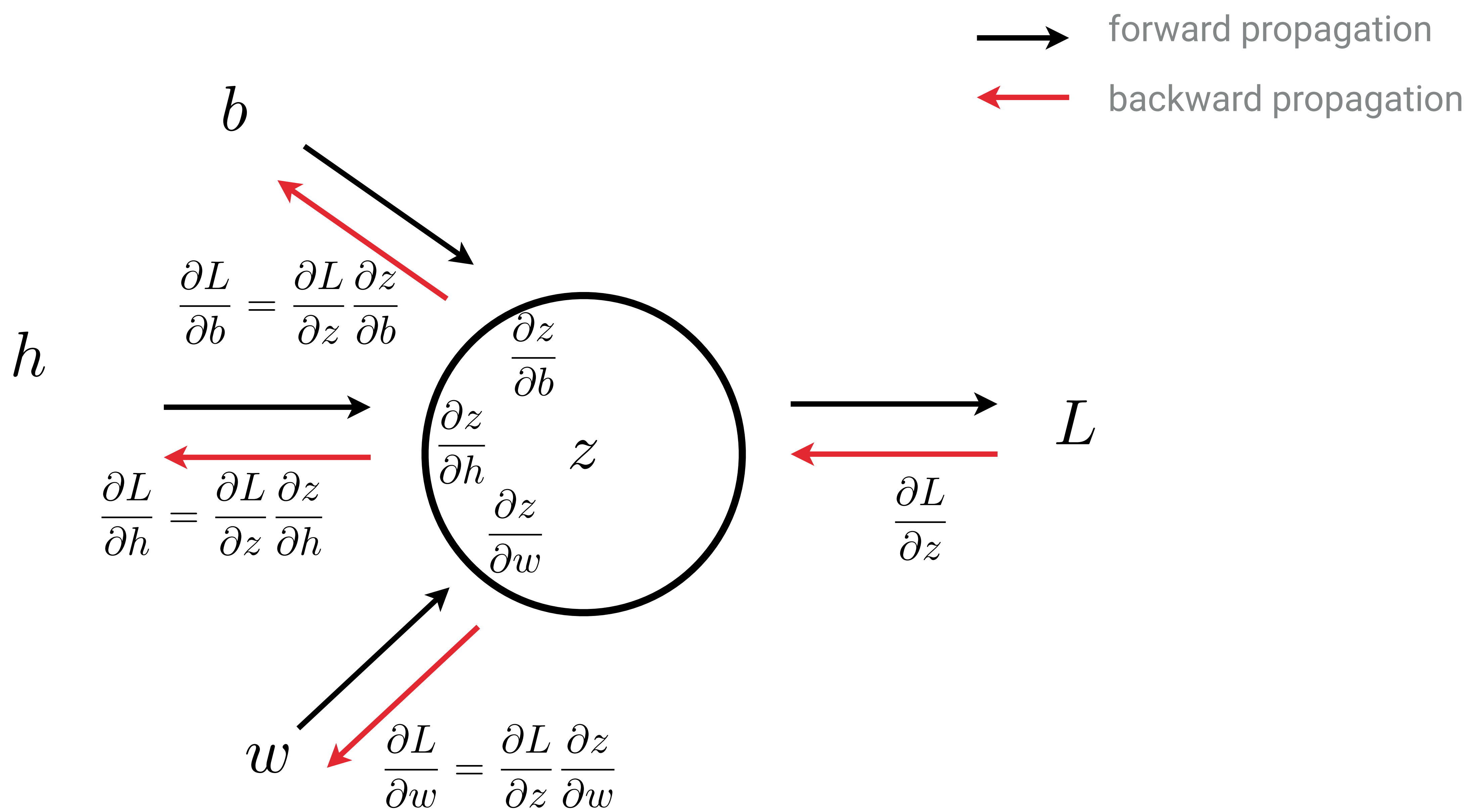
$$\mathbf{y} = f(\mathbf{x})$$

$$L = g(f(\mathbf{x}))$$

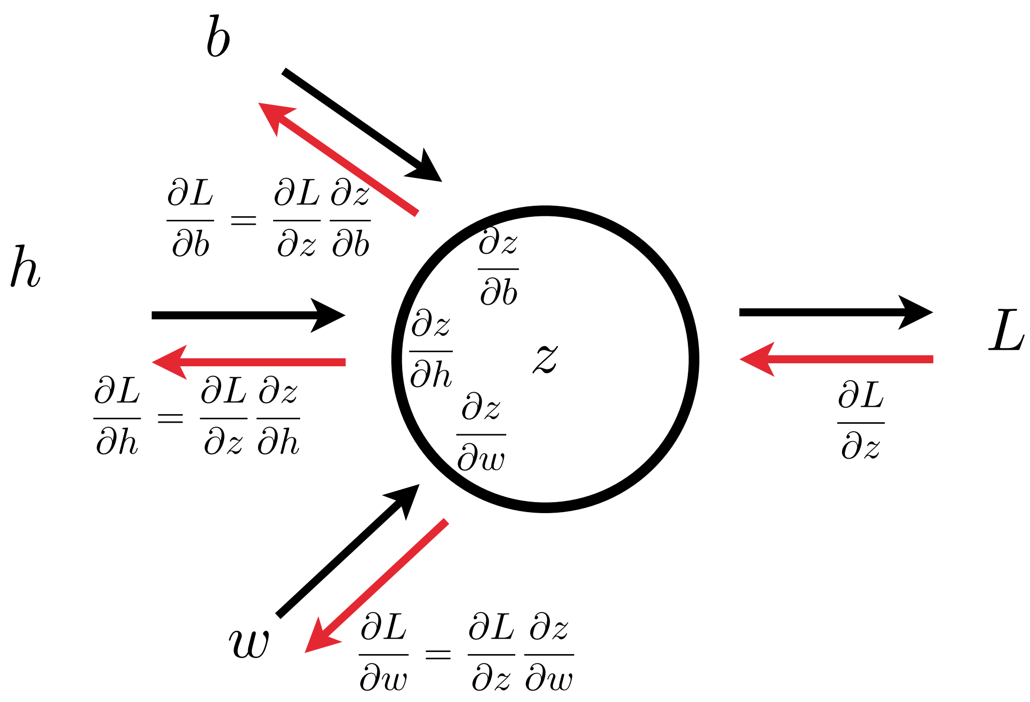
$$\frac{\partial L}{\partial \mathbf{x}} = J^T \frac{\partial L}{\partial \mathbf{y}} = \frac{\partial \mathbf{y}}{\partial \mathbf{x}} \frac{\partial L}{\partial \mathbf{y}}$$

$$J^T = \begin{pmatrix} \frac{\partial y_1}{\partial x_1} & \cdots & \frac{\partial y_m}{\partial x_1} \\ \vdots & \ddots & \vdots \\ \frac{\partial y_1}{\partial x_n} & \cdots & \frac{\partial y_m}{\partial x_n} \end{pmatrix}$$

HOW TO EFFICIENTLY COMPUTE THE GRADIENT



HOW TO EFFICIENTLY COMPUTE THE GRADIENT




```
class node(object):
 def forward(z):
  output = f(z)
  return output

 def backward(z, dLdz):
  J = jacobian(z)
  return J.dot(dLdz)
```

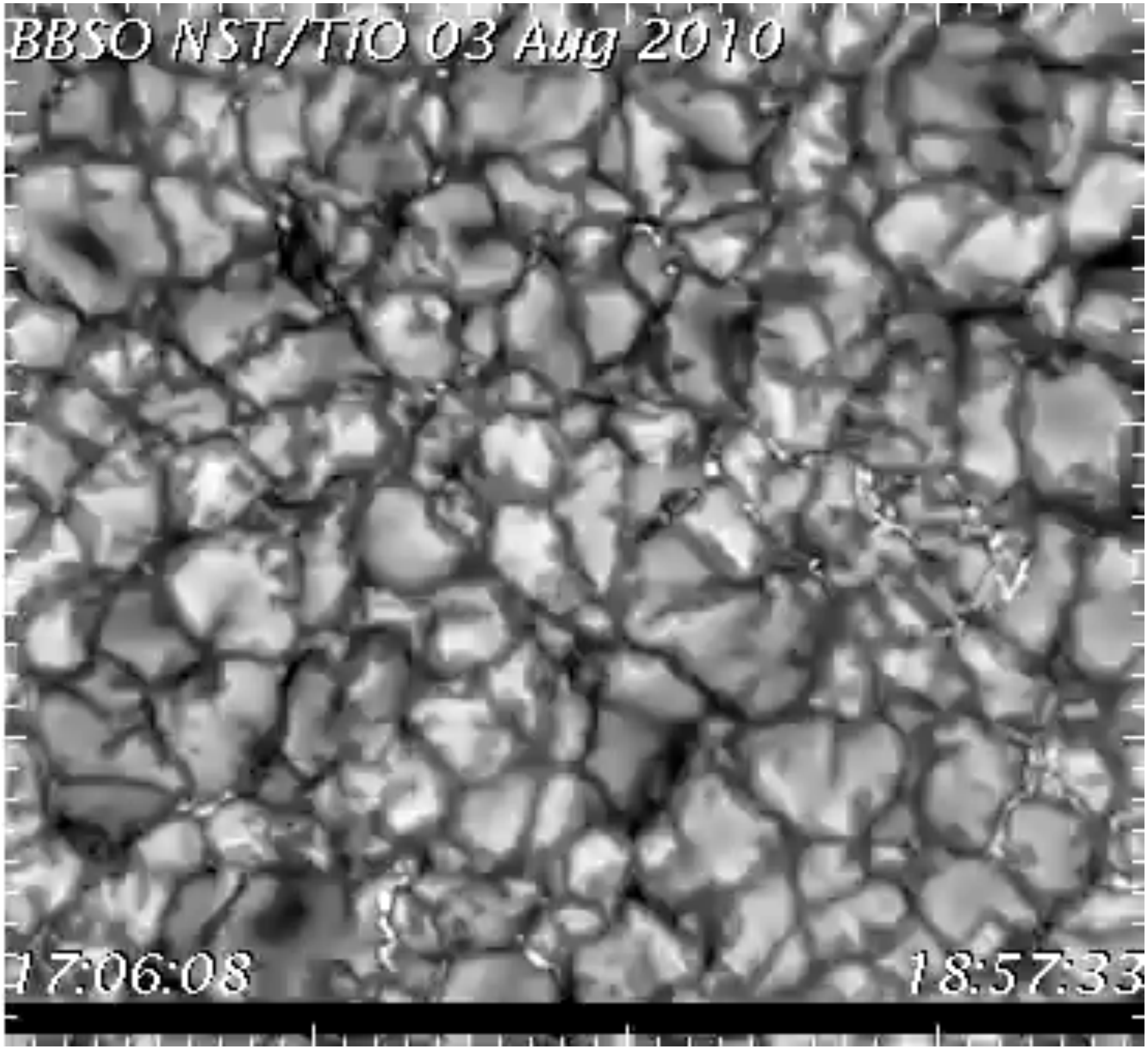
Applications in Solar Physics

PROBLEMS TACKLED SO FAR

- Measuring velocities
- Enhancing HMI images
- Multiframe blind deconvolution
- Fast inversion of Stokes profiles
- Farside imaging
- Classification of solar structures
- Physical conditions in flares

measuring velocities

MEASURING VELOCITIES



MEASURING VELOCITIES

Longitudinal component

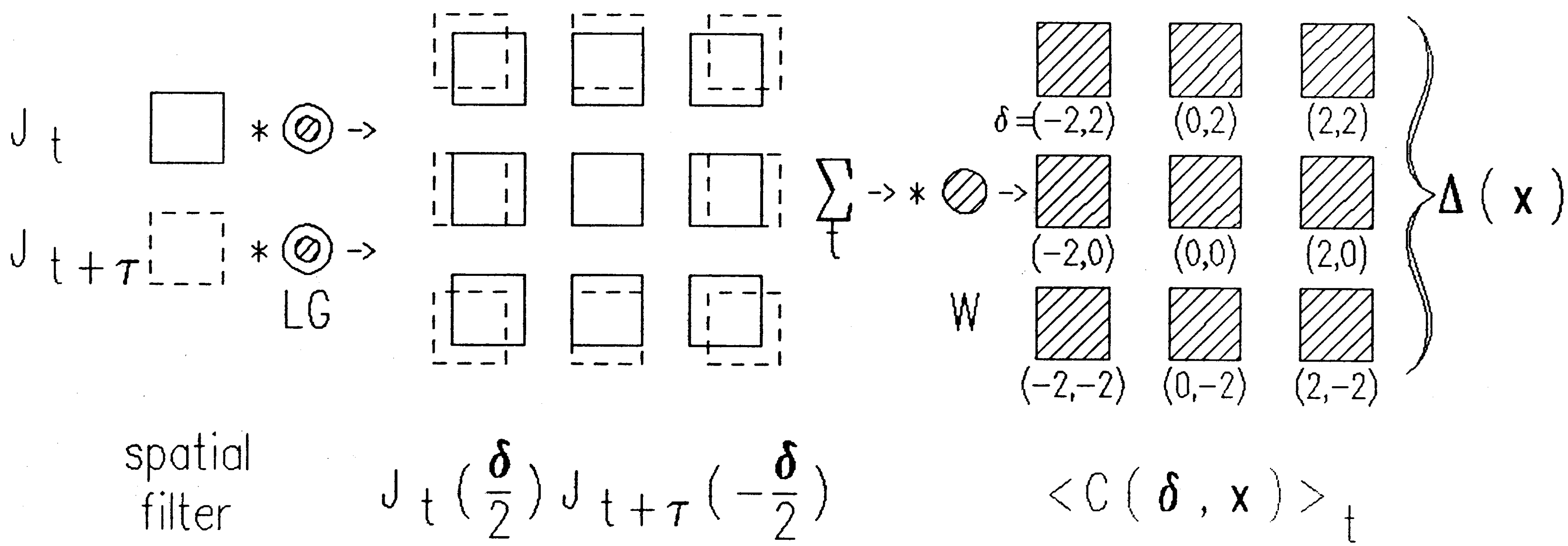
- Can be measured with Doppler effect using spectroscopy
- Physical meaning

Transverse component

- Cannot be spectroscopically measured
- Not obvious physical meaning
- Different depending on selection of “corks”

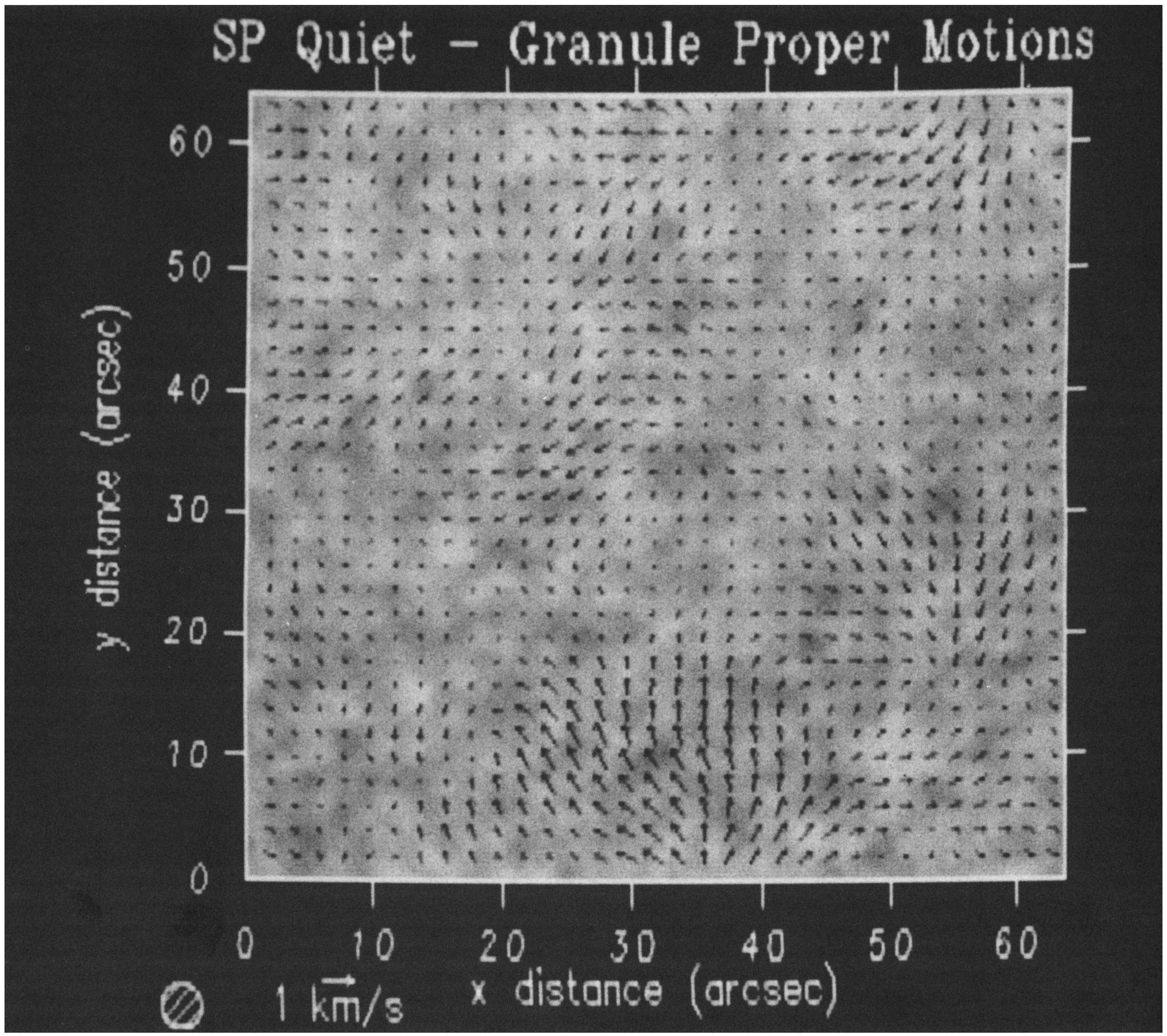
MEASURING VELOCITIES IN THE PLANE OF THE SKY

November & Simon (1988) - Local correlation tracking



MEASURING VELOCITIES IN THE PLANE OF THE SKY

November & Simon (1988) - Local correlation tracking



- Spatial correlation window
- Temporal correlation window
- Noise sensitive

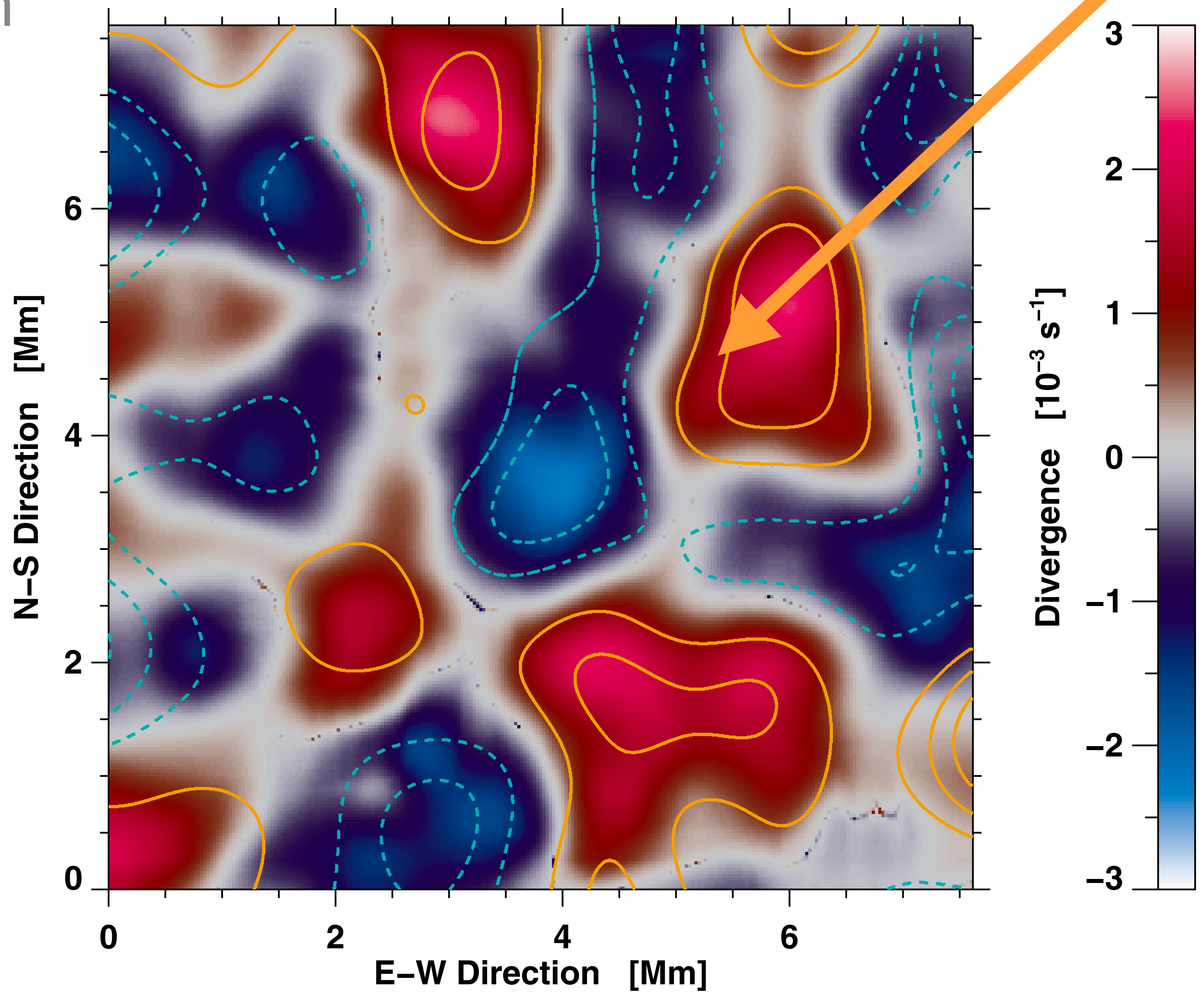
LCT VS. SIMULATIONS

Average time 1 h

FWHM = 1200 km

Divergence map

1-hour average from simulations

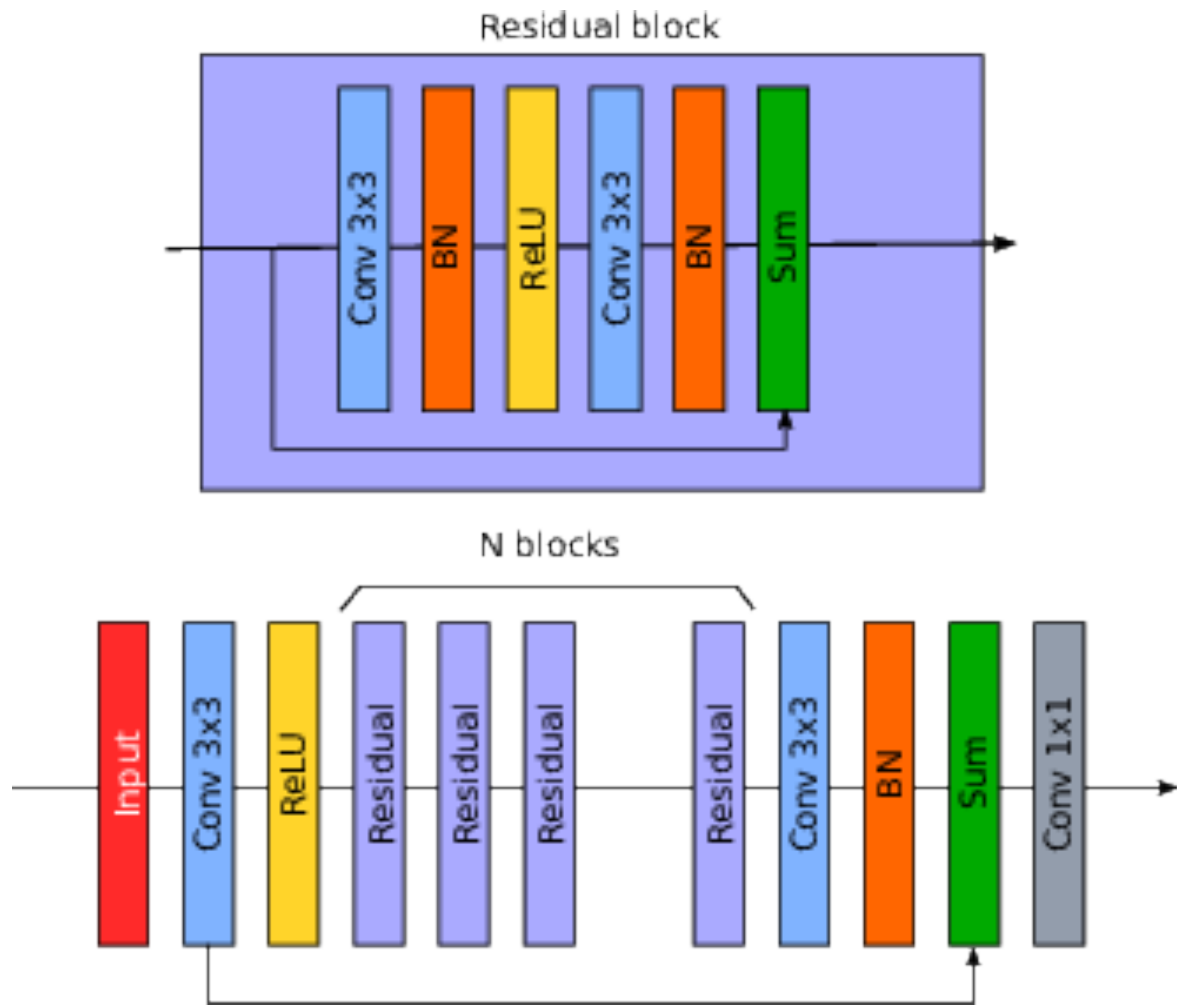


Verma & Denker (2013)

LIST OF DESIRES: DEEPVEL

- End-to-end approach
- Scale to any image size
- Be fast
- Easy to train

DEEPVEL: ARCHITECTURE

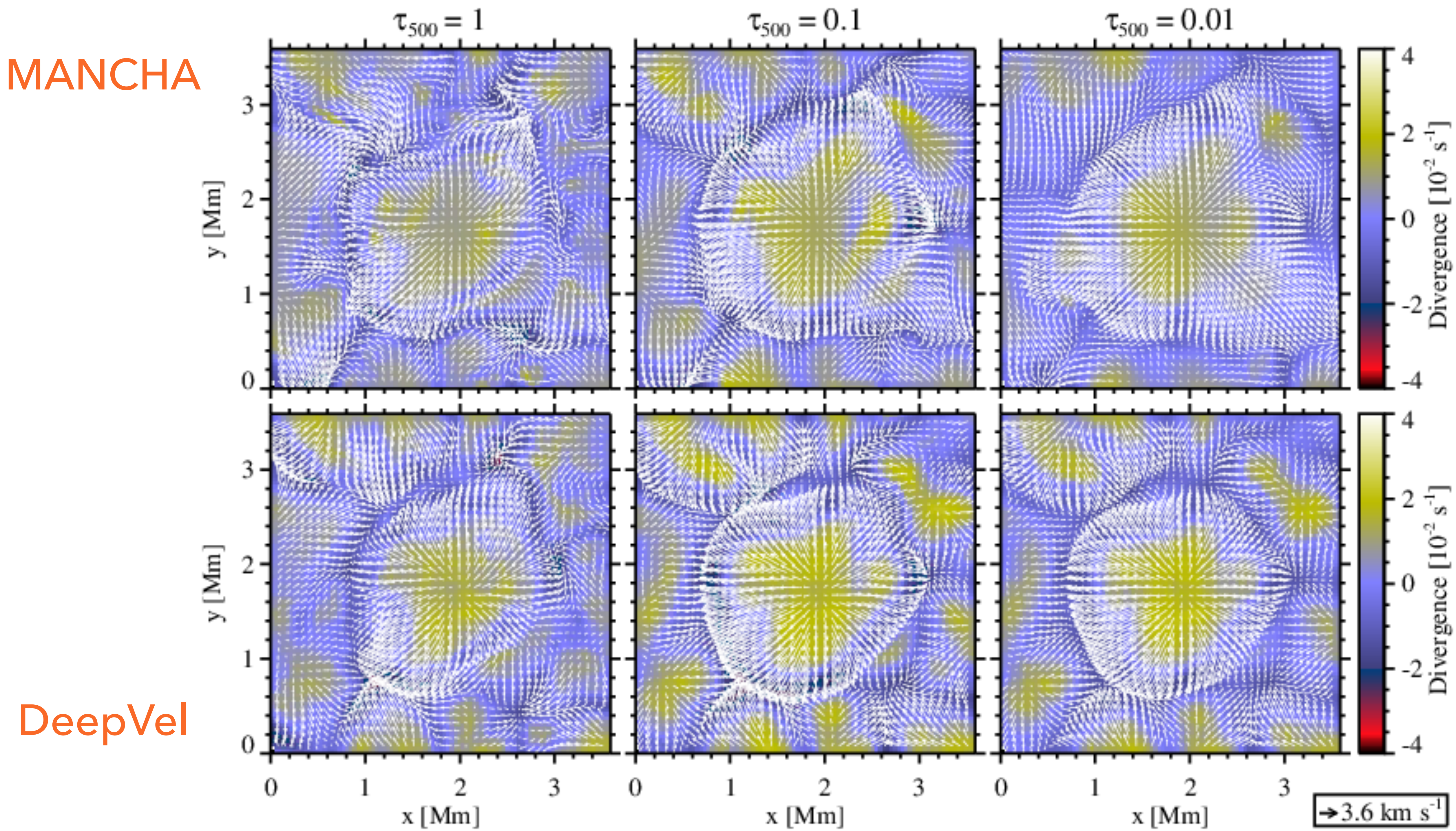


Asensio Ramos, Requerey & Vitas (2017)

DEEPVEL: TRAINING WITH SIMULATIONS

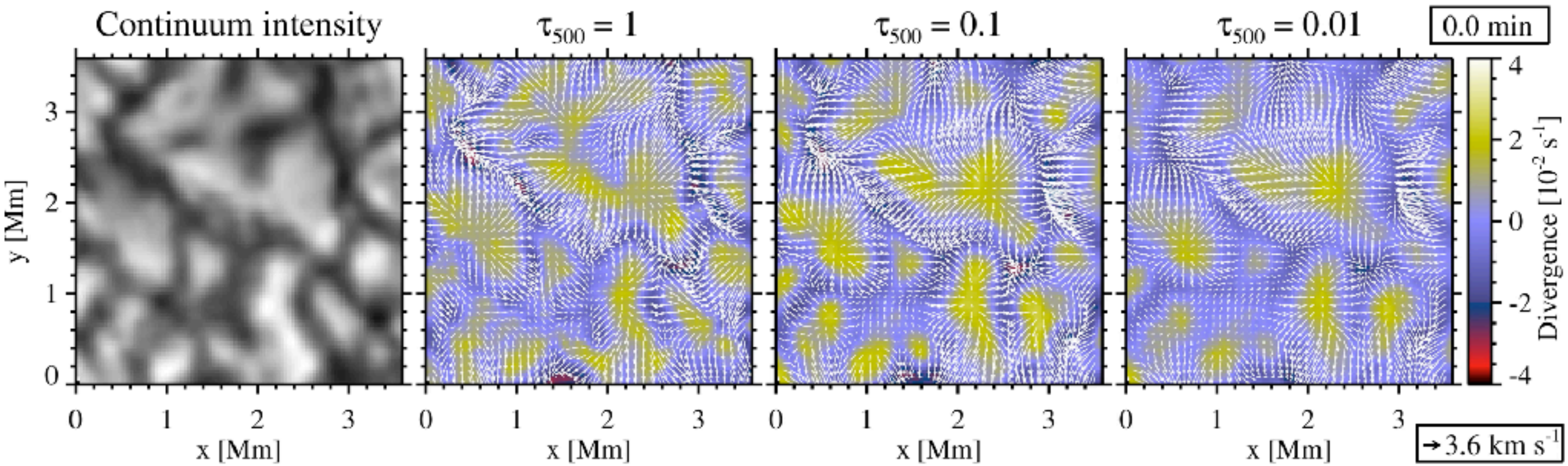
- Synthetic images from Stein & Nordlund (2012) + degradation
- We extract 30000 pairs of patches of 50x50 pixels separated by 30 s
- The outputs are maps of v_x and v_y at τ=1,0.1,0.01
- Loss function : ℓ_2 -norm between predicted and simulated velocities
- Trained with ADAM optimizer with $\beta=10^{-4}$ for 900k steps

VALIDATION



MANCHA
DeepVel
$\tau_{500} = 1$
$\tau_{500} = 0.1$
$\tau_{500} = 0.01$
y [Mm]
x [Mm]
Divergence [10^{-2} s^{-1}]
→3.6 km s^{-1}

DEEPVEL https://github.com/aasensio/deepvel

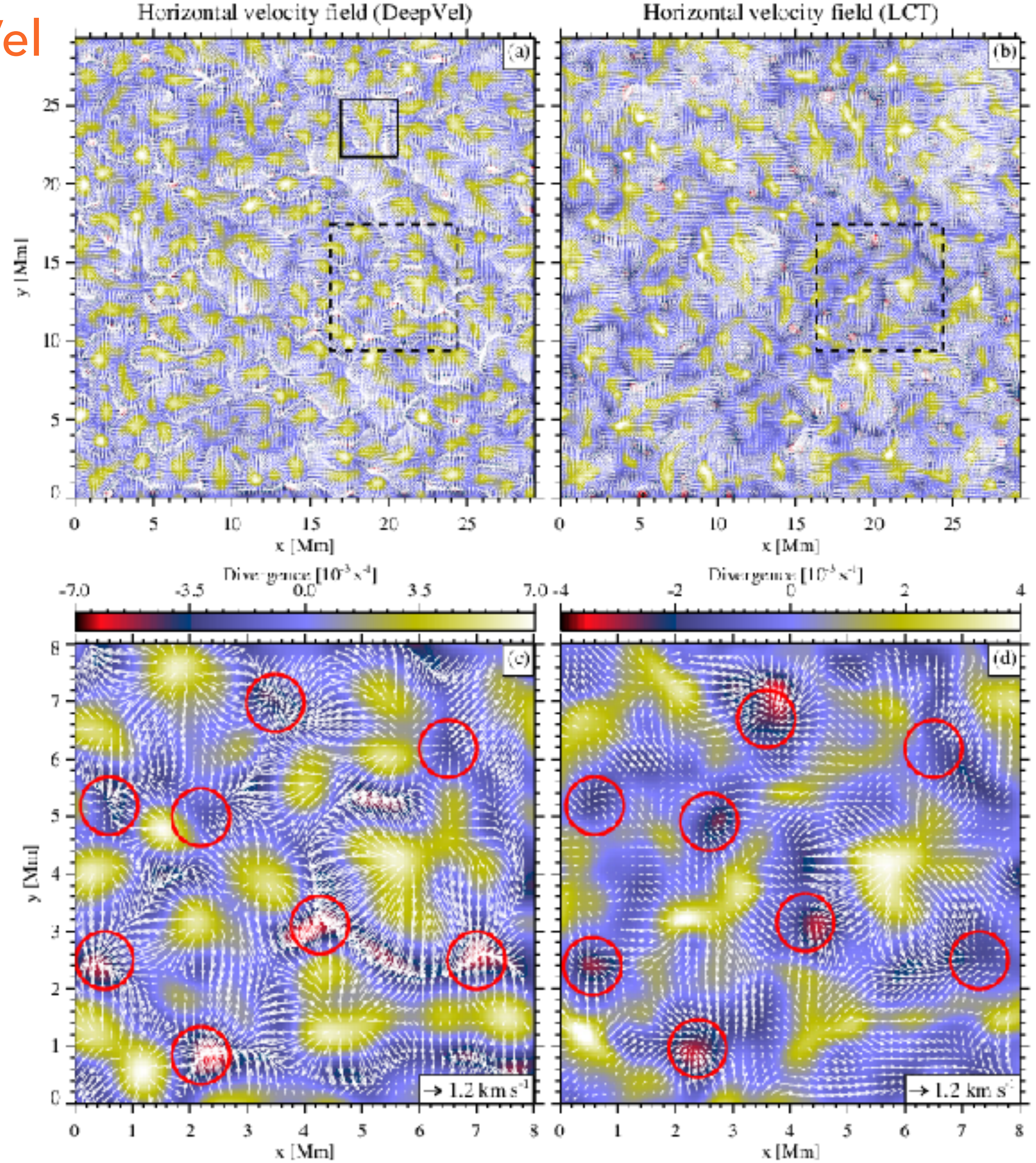


Asensio Ramos, Requerey & Vitas (2017)

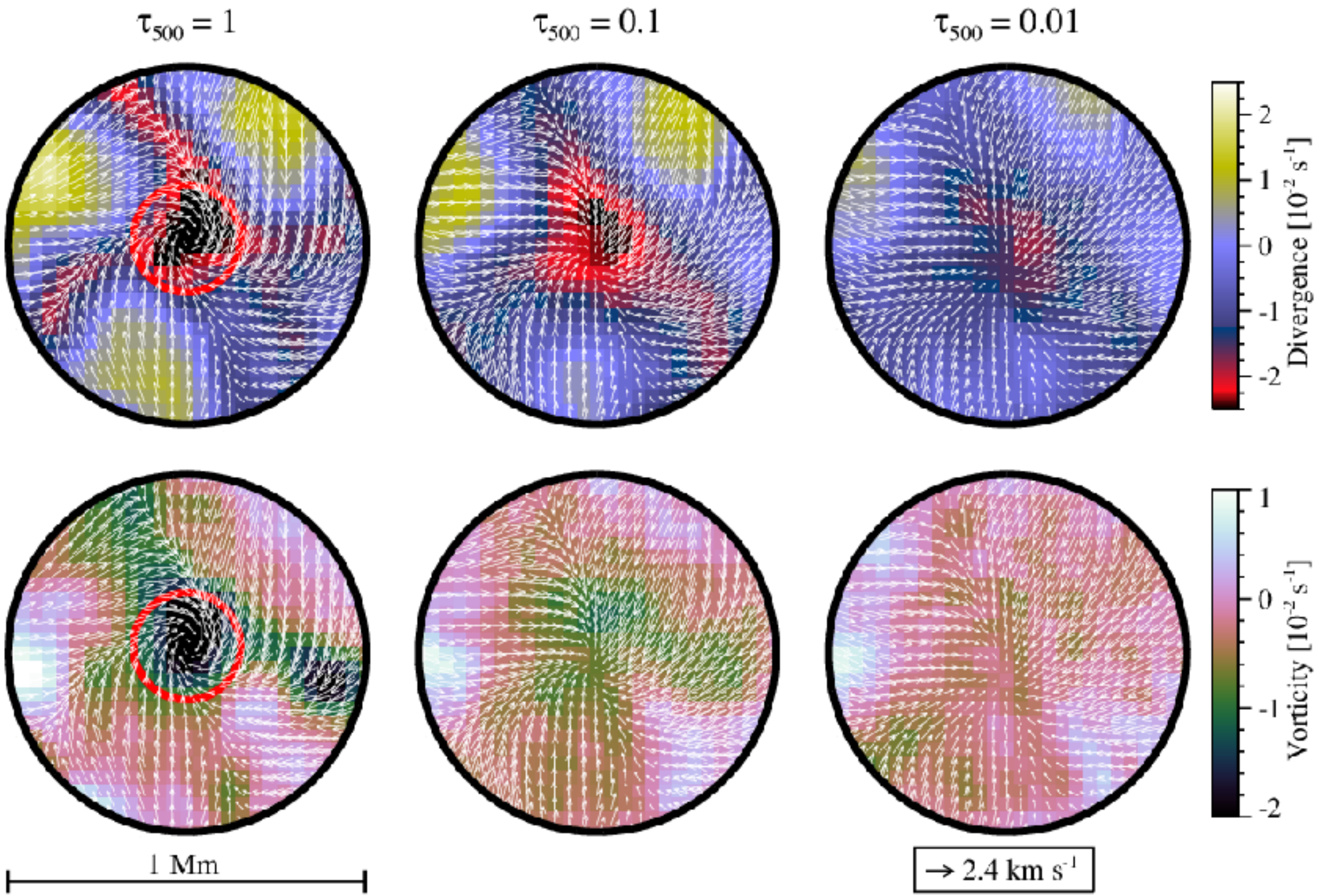
AVERAGE PROPERTIES

DeepVel

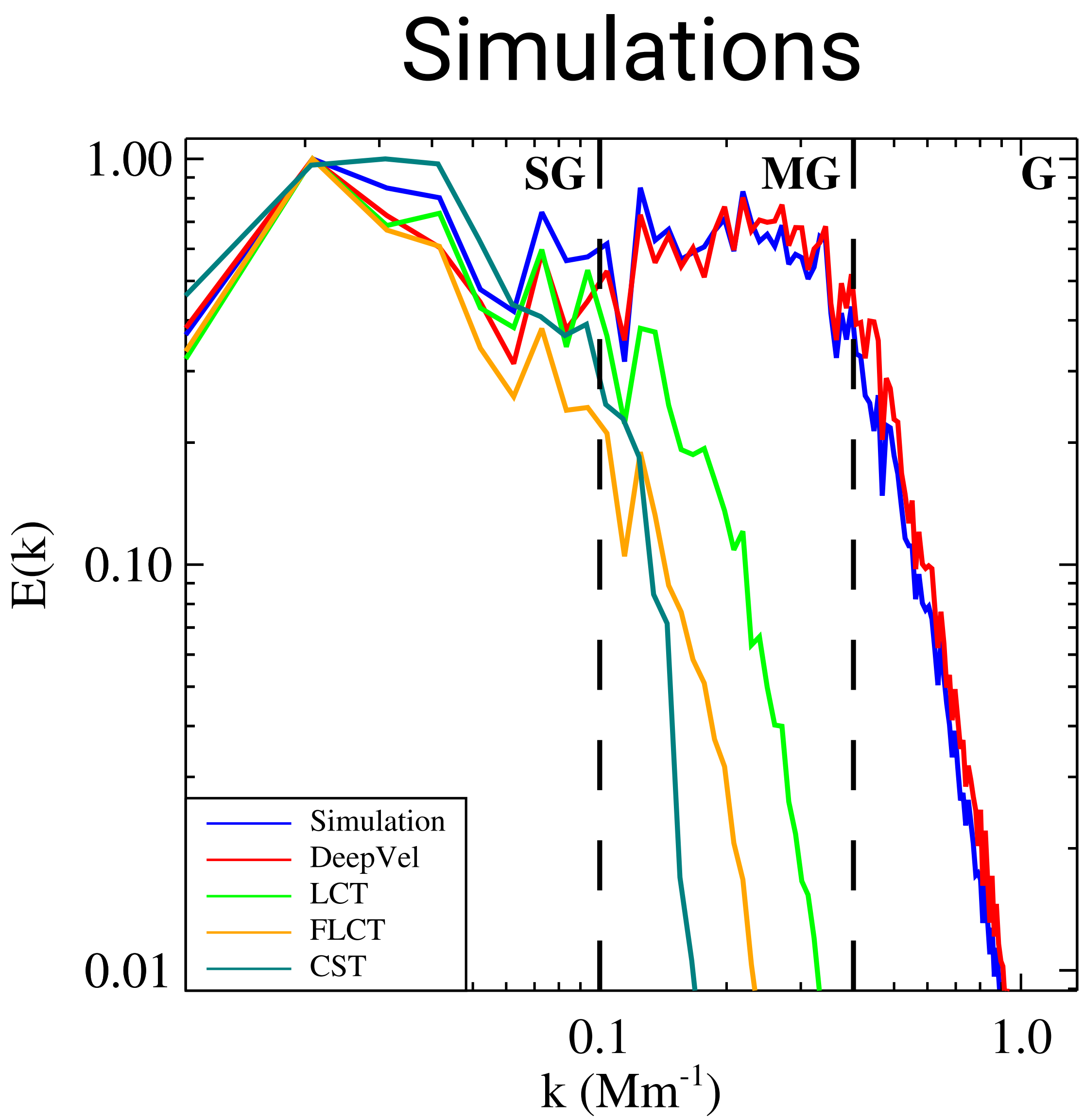
LCT

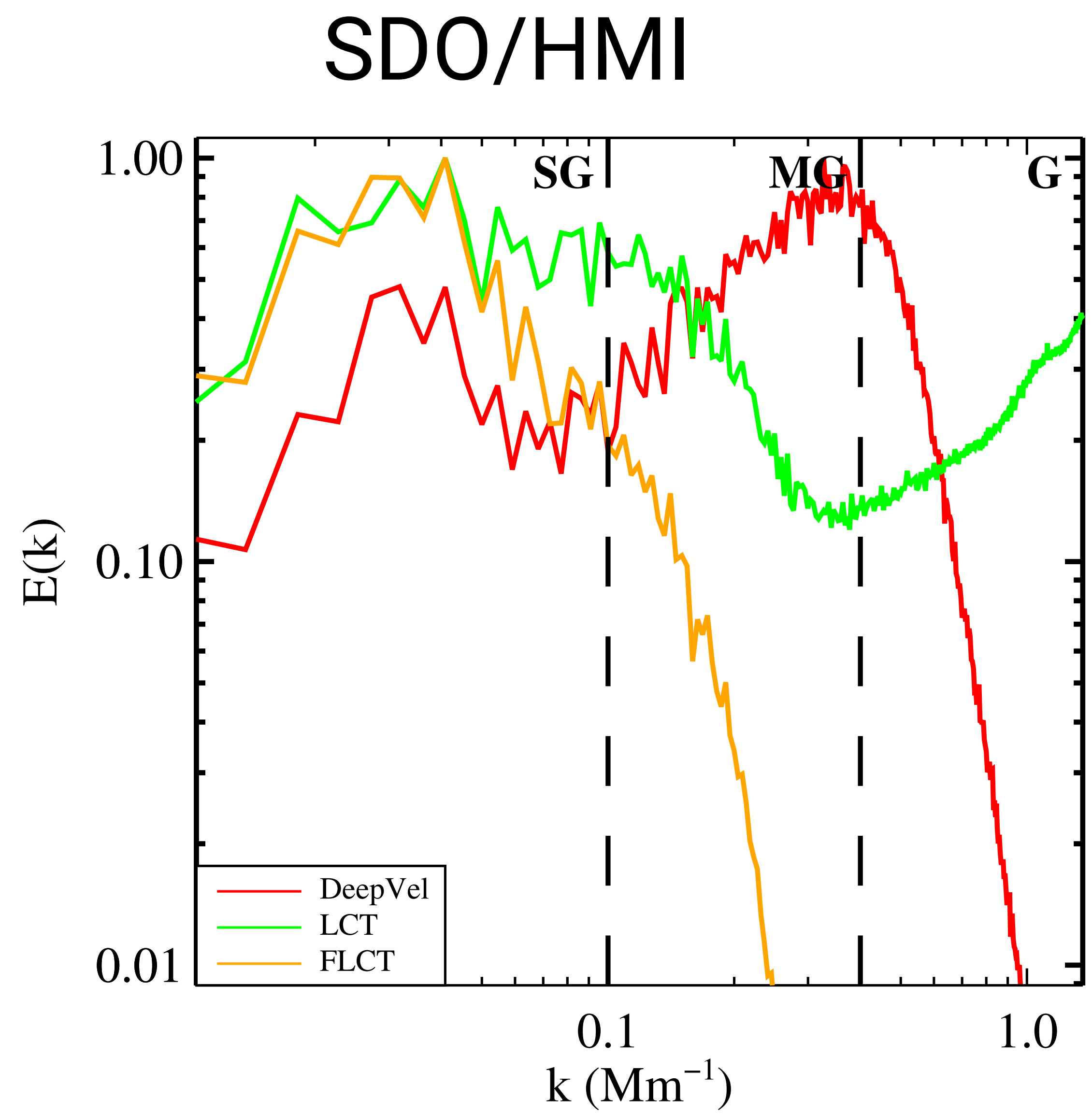


SMALL SCALE VORTEX FLOWS



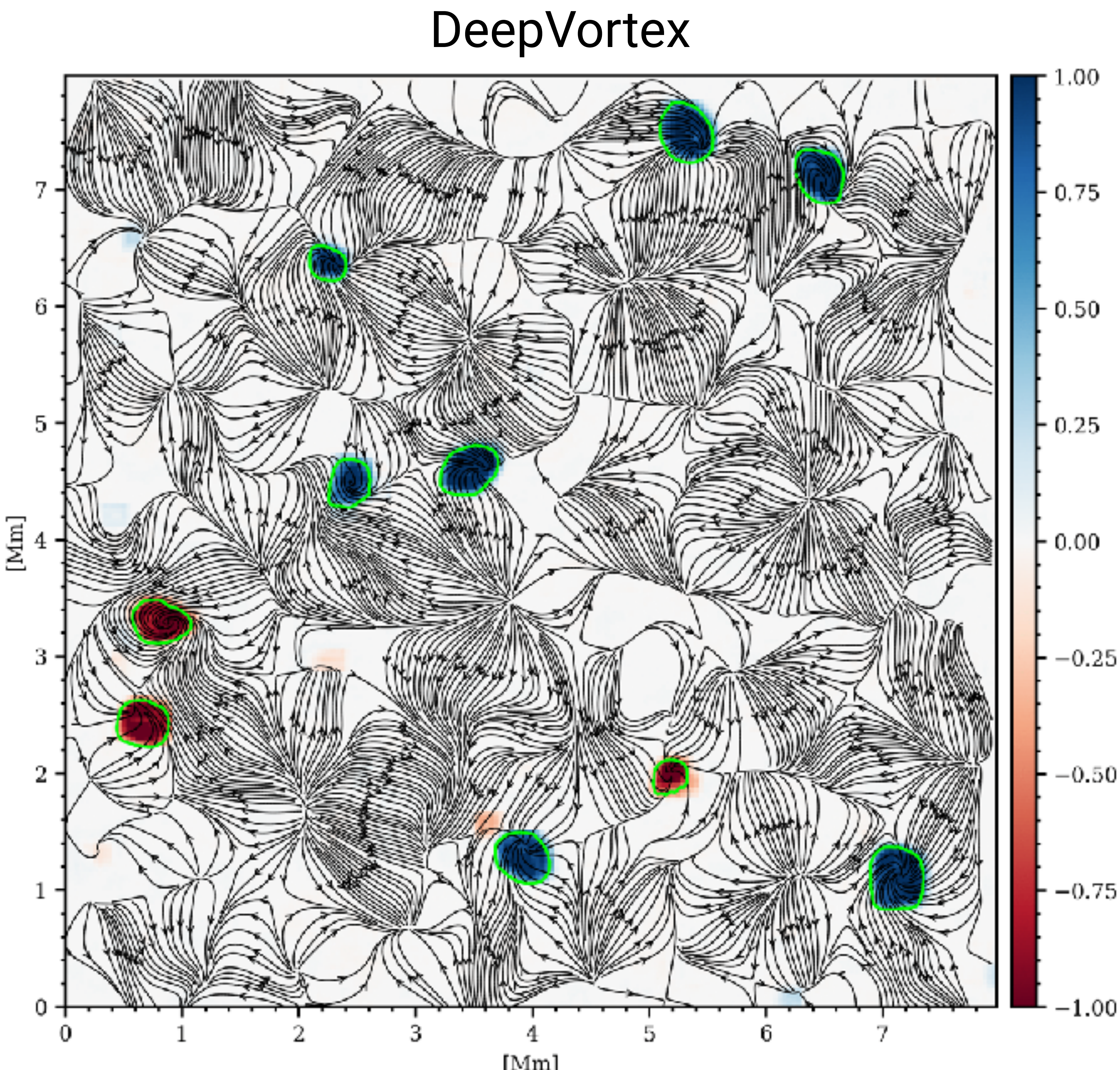
KINETIC ENERGY SPECTRUM





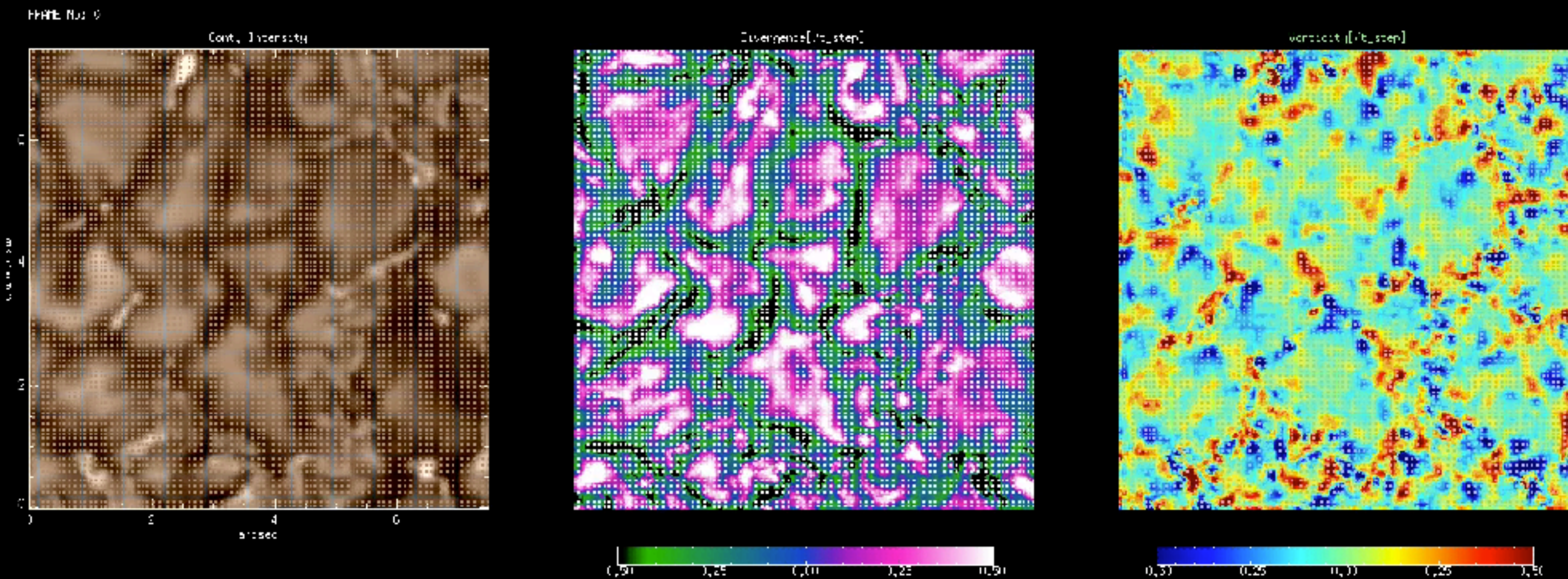
Tremblay et al. (2018)

VORTEX DETECTION



Requerey et al. (in prep)

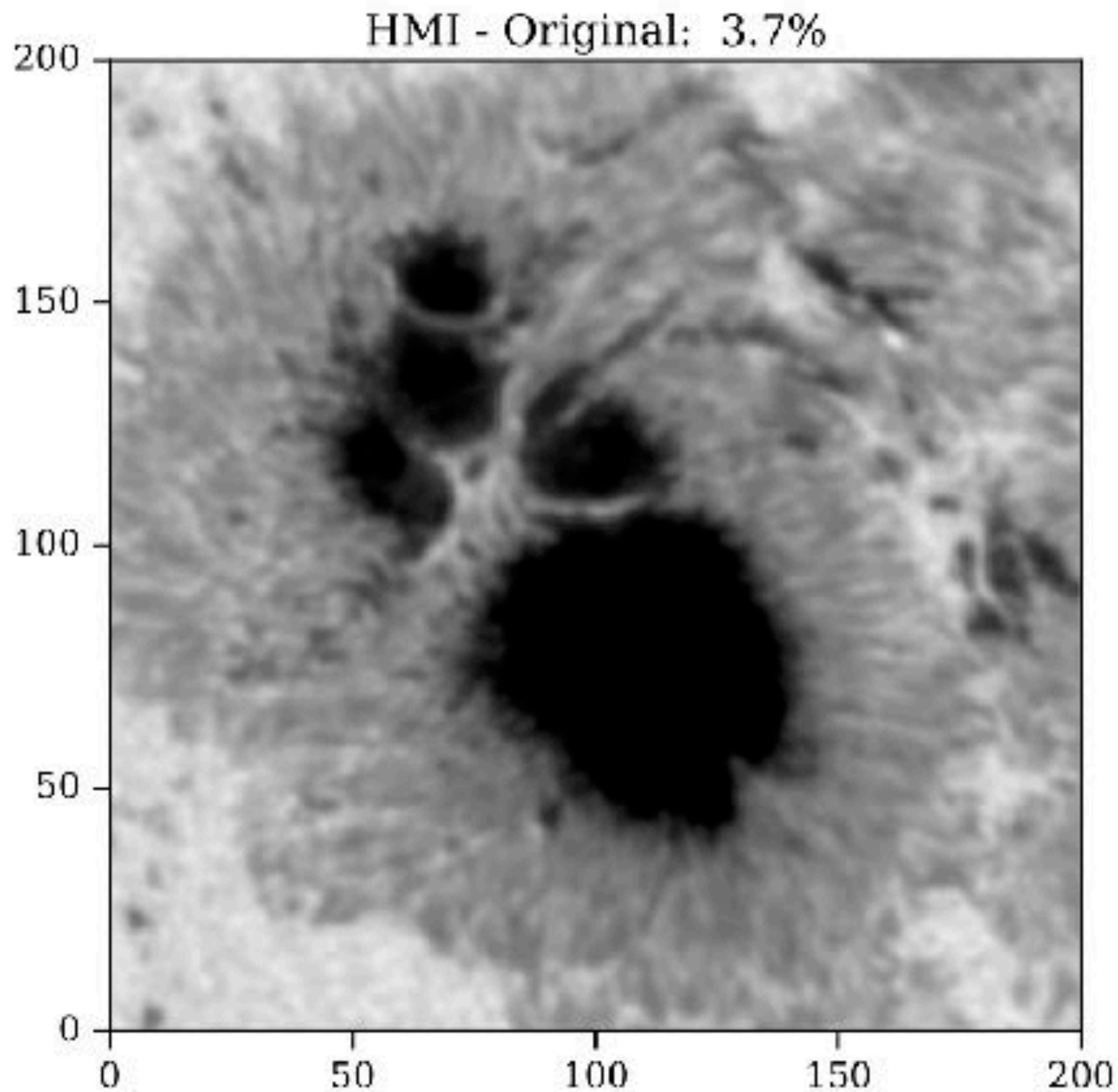
CORKS EVOLUTION

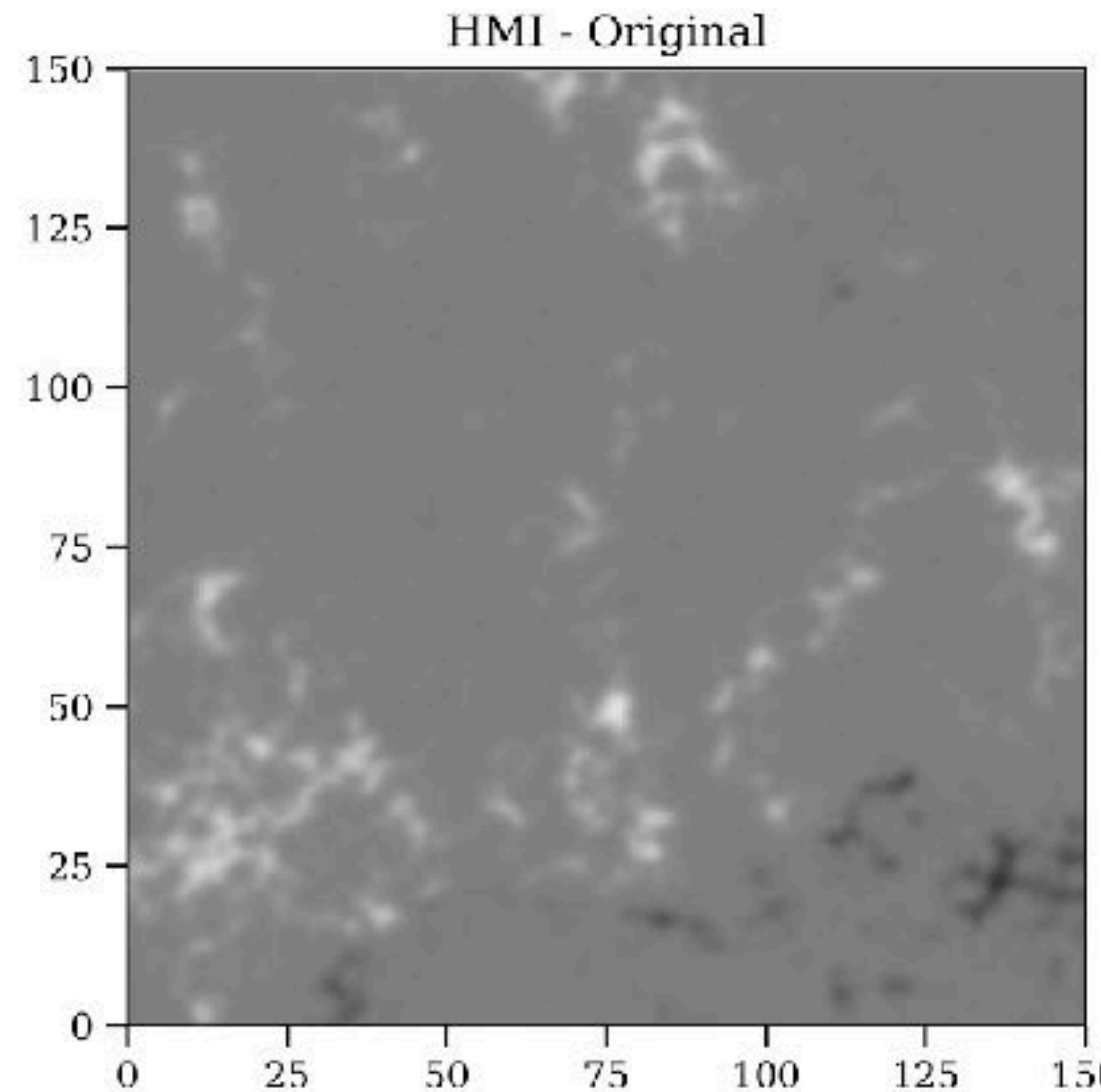


Rouppe van der Voort (private comm)

enhancing HMI images

HMI: 24/7 BUT NOT ENOUGH SPATIAL RESOLUTION

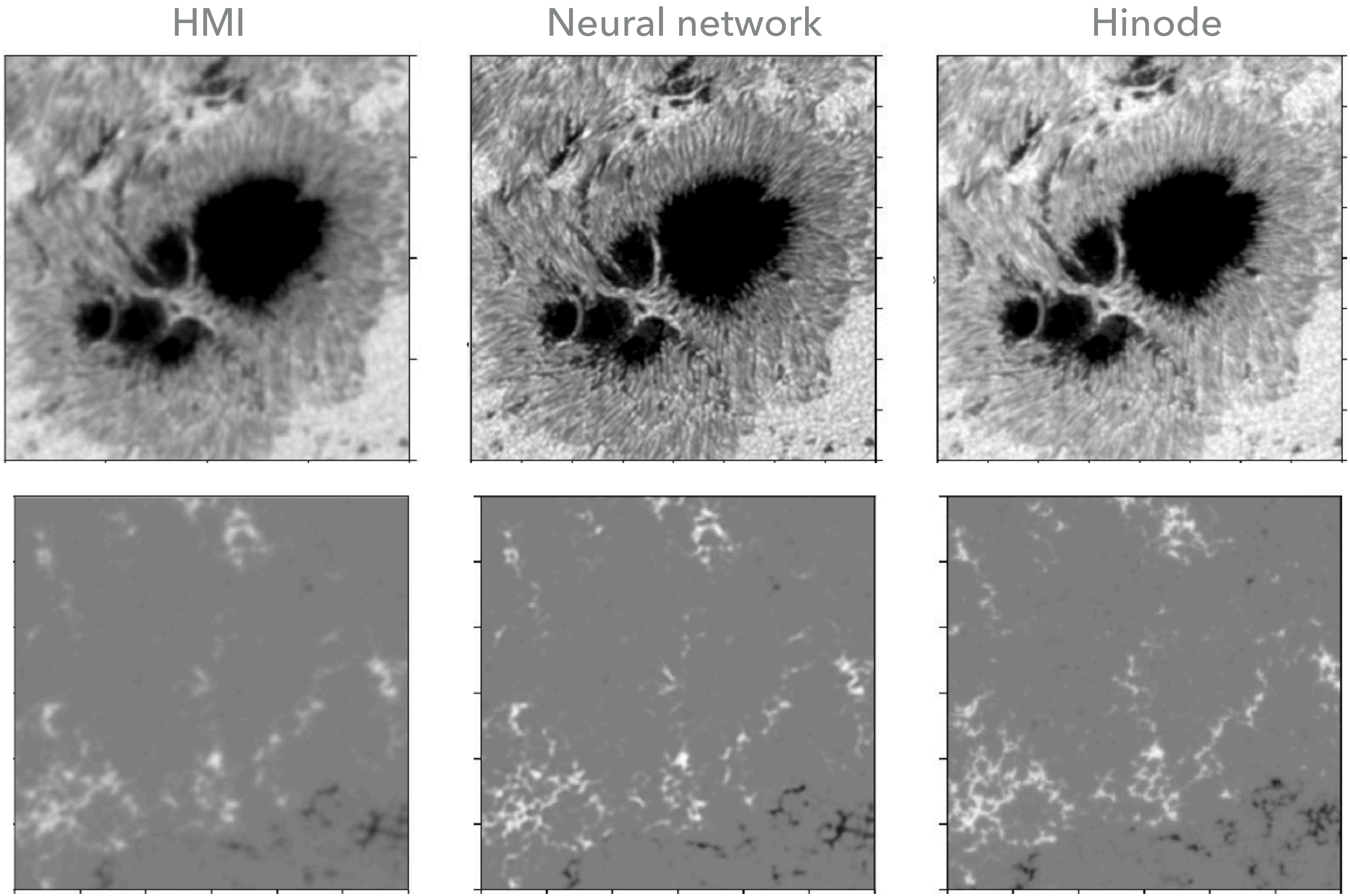




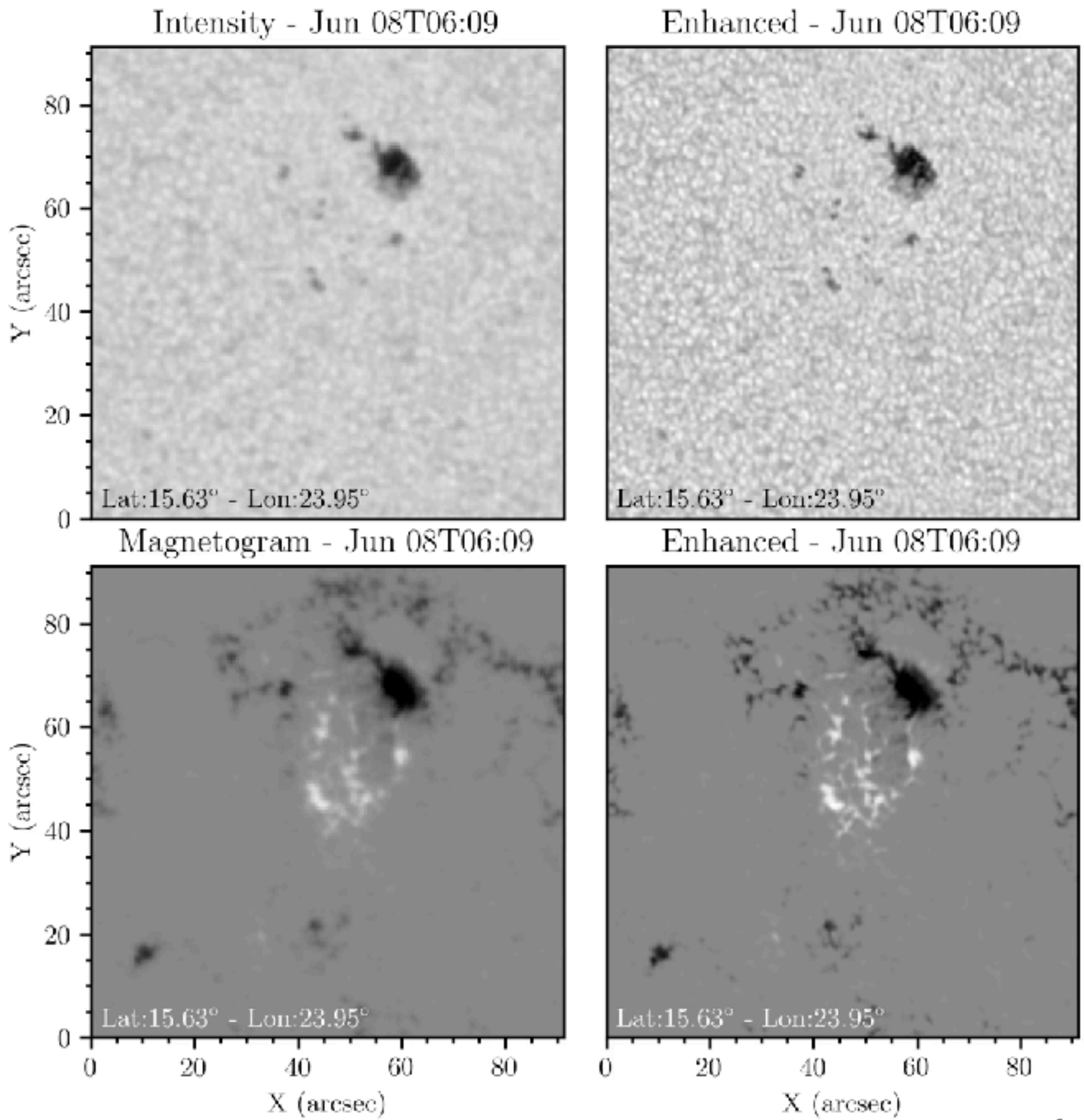
ENHANCE:

- Trained on simulations (courtesy of M. Cheung)
- End-to-end deep neural network
- Continuum + magnetograms

ENHANCE: SINGLE IMAGE SUPERRESOLUTION



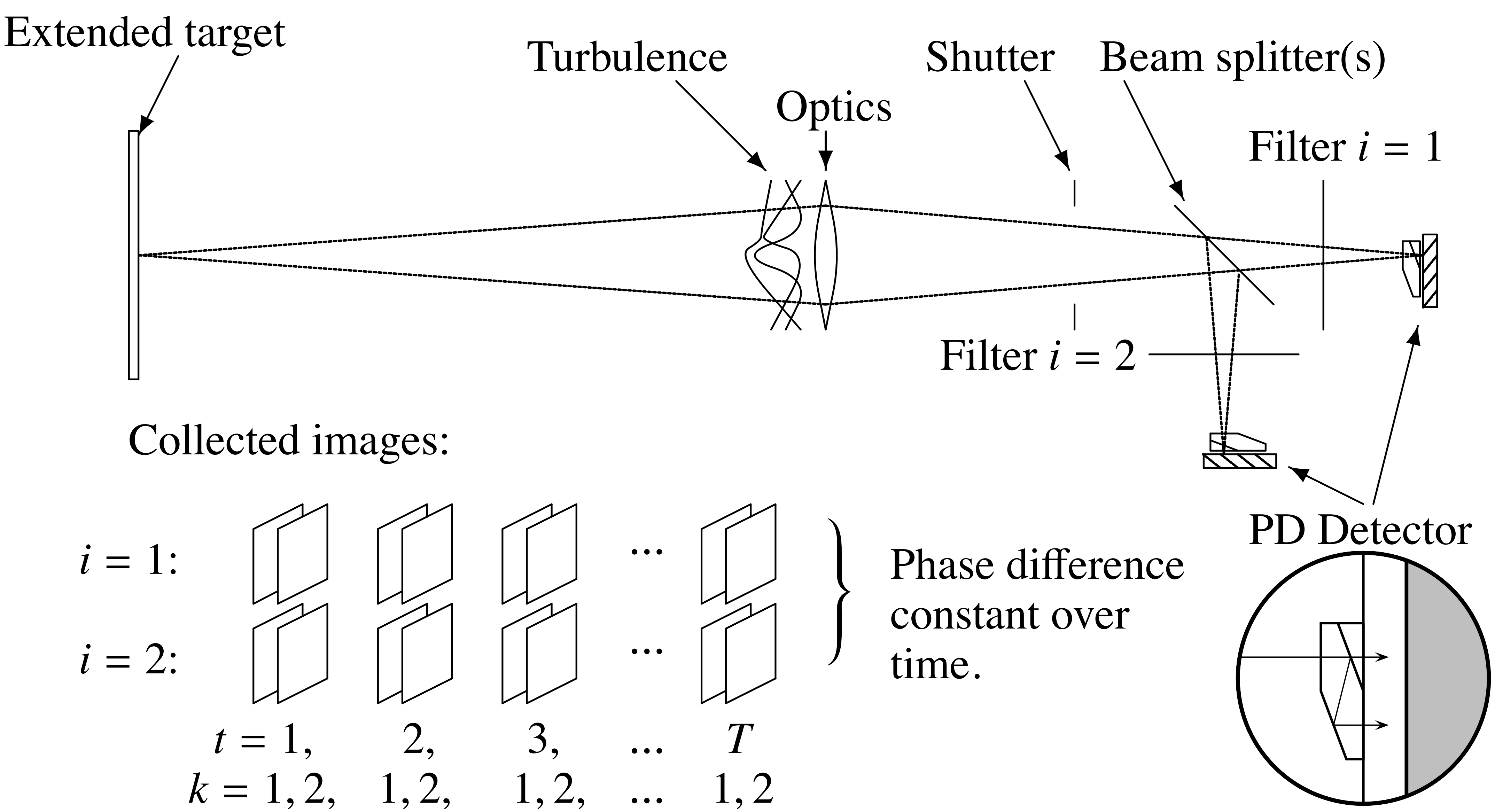
ENHANCE https://github.com/cdiazbas/enhance



courtesy of S. Castellanos Durán

real-time multiframe deconvolution

MULTIFRAME BLIND DECONVOLUTION



van Noort et al. (2005)

MULTIFRAME BLIND DECONV : MAX-LIKELIHOOD

Observed frames

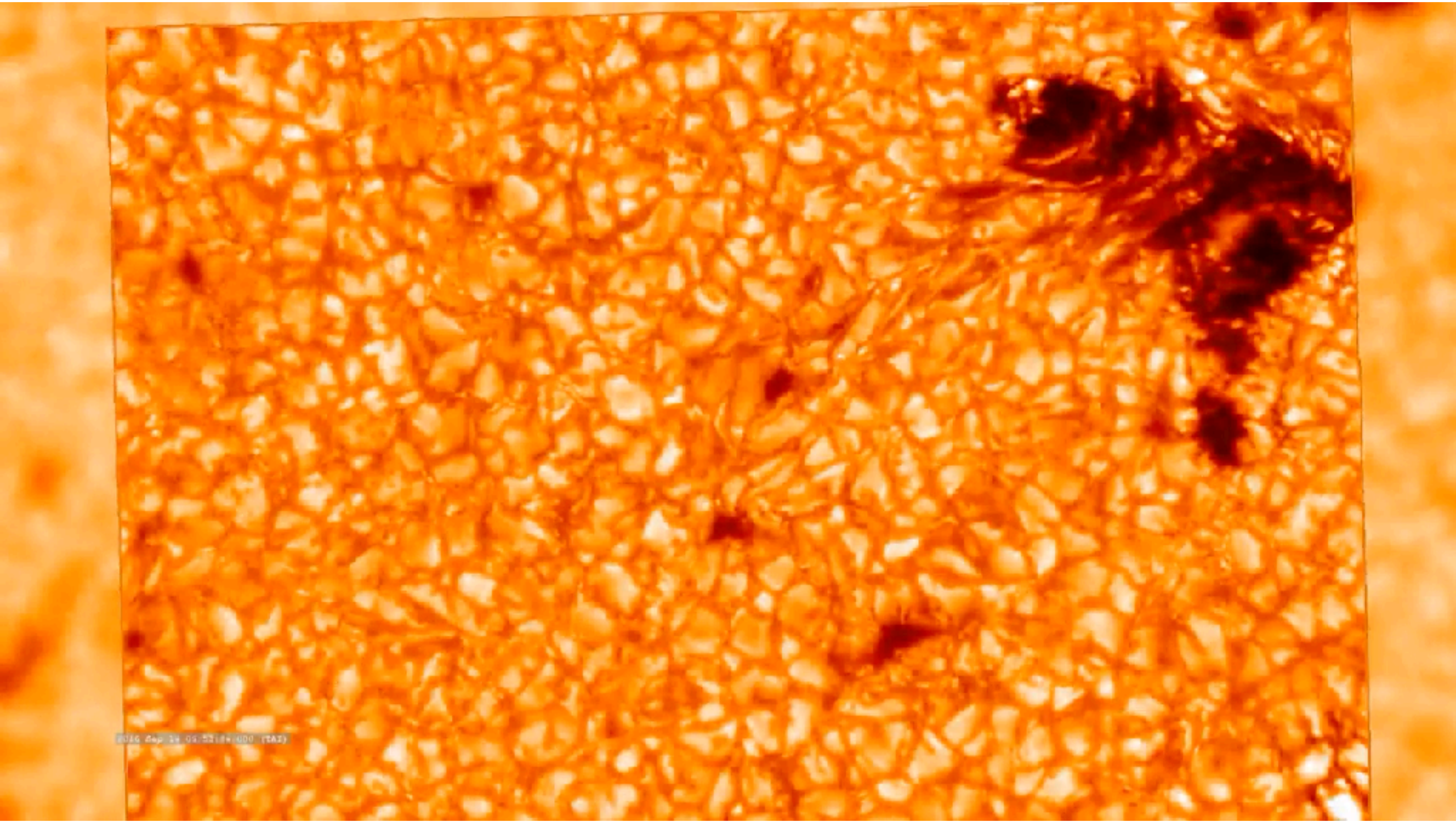
Optical transfer function

$$L_i(\boldsymbol{\alpha}_i) = \sum_u \left[\sum_j^J |D_{ij}|^2 - \frac{\left| \sum_j^J D_{ij}^* \hat{S}_{ij} \right|^2}{\sum_j^J \left| \hat{S}_{ij} \right|^2 + \gamma_i} \right]$$

$$P_{ij} = A_{ij} \exp\{\mathrm{i}\phi_{ij}\}$$

van Noort et al. (2005)

MULTIFRAME BLIND DECONVOLUTION



MULTIFRAME BLIND DECONVOLUTION

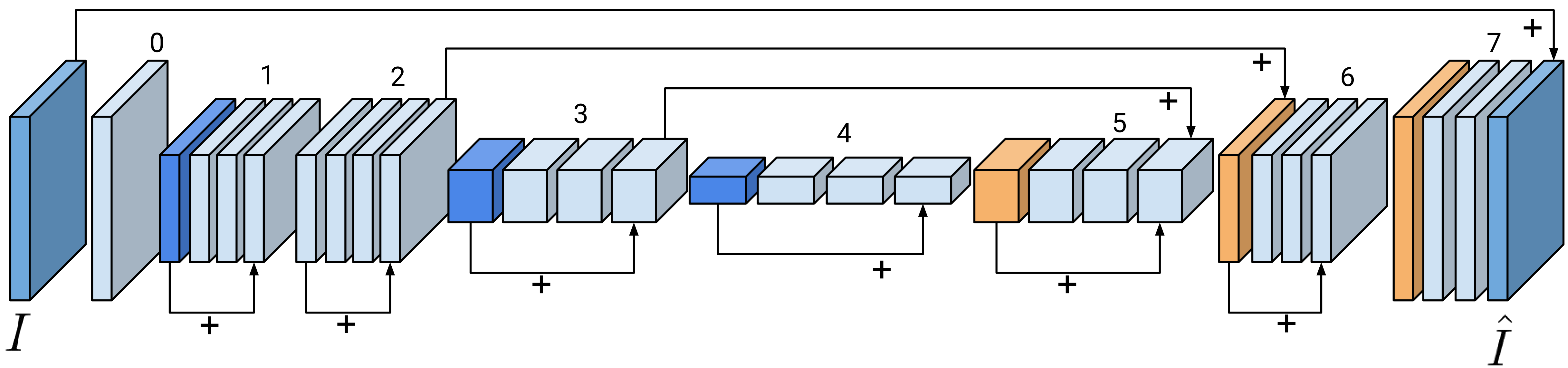
Short-exposure burst

Deconvolved image

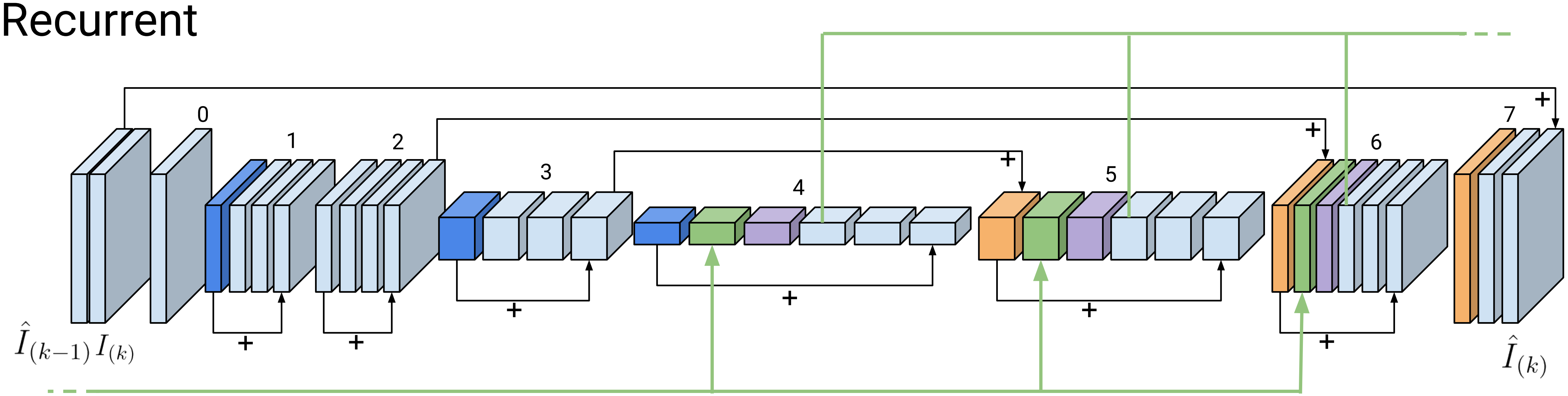
- Trained on CRISP@SST Fe I 630 nm and Ca II 854 nm deconvolved data
- End-to-end deep neural network
- Asensio Ramos et al. (A&A, arXiv:1806.07150)
- 1k x 1k images at ~100 Hz
- https://github.com/aasensio/learned_mfbd

MULTIFRAME BLIND DECONVOLUTION

Encoder-decoder

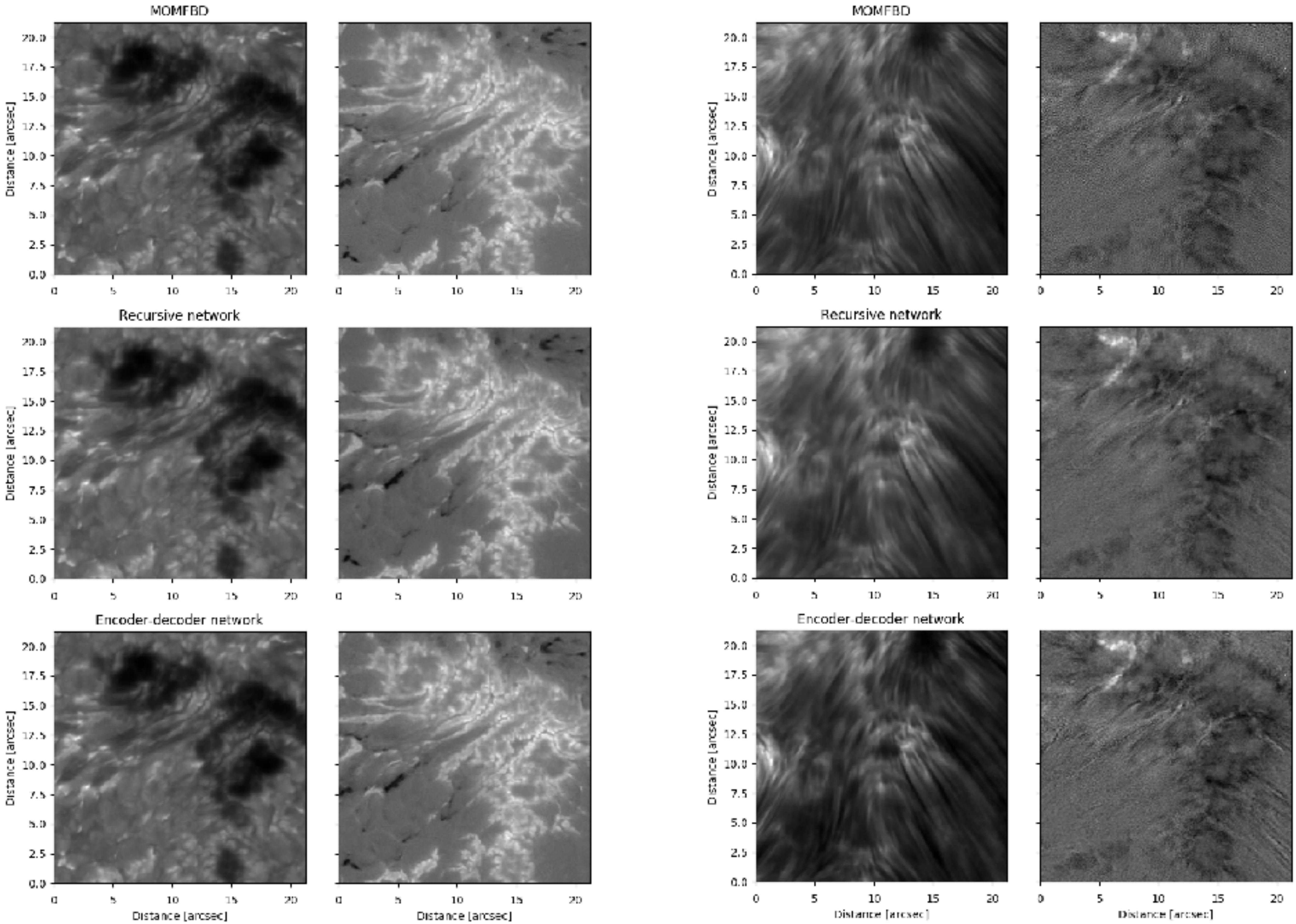


Recurrent

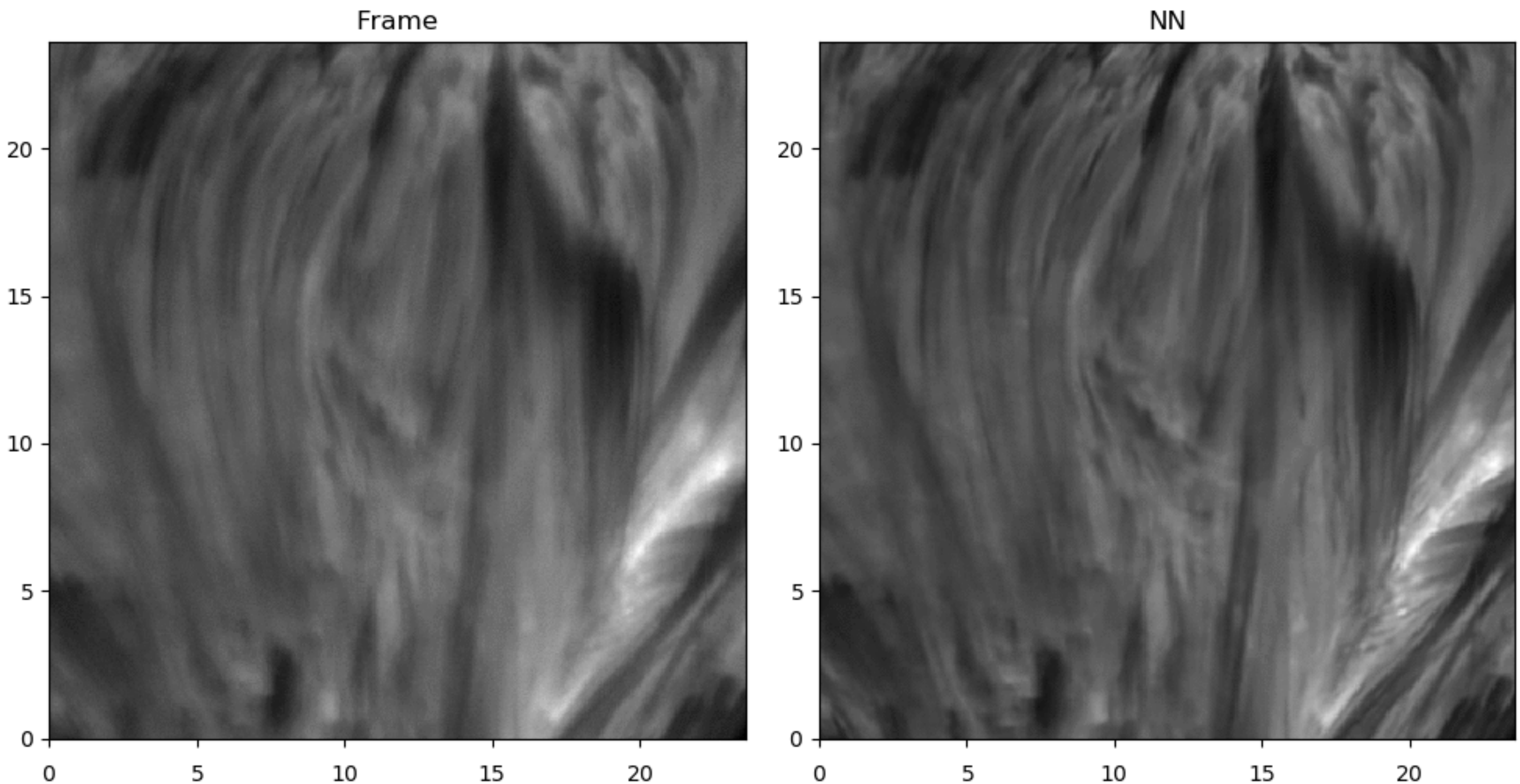


https://github.com/aasensio/learned_mfbd

POLARIMETRY

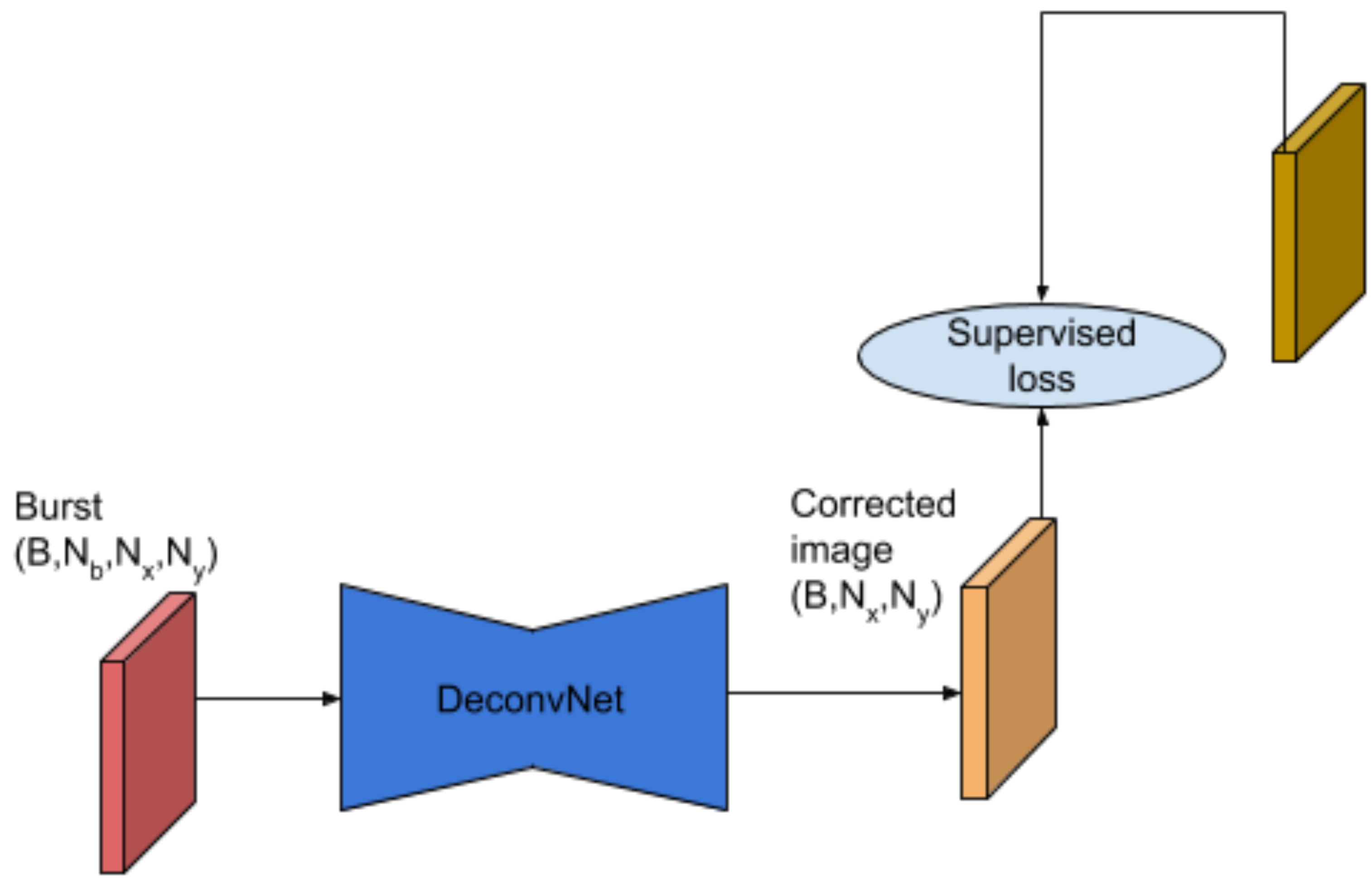


GENERALIZATION TO UNSEEN DATA

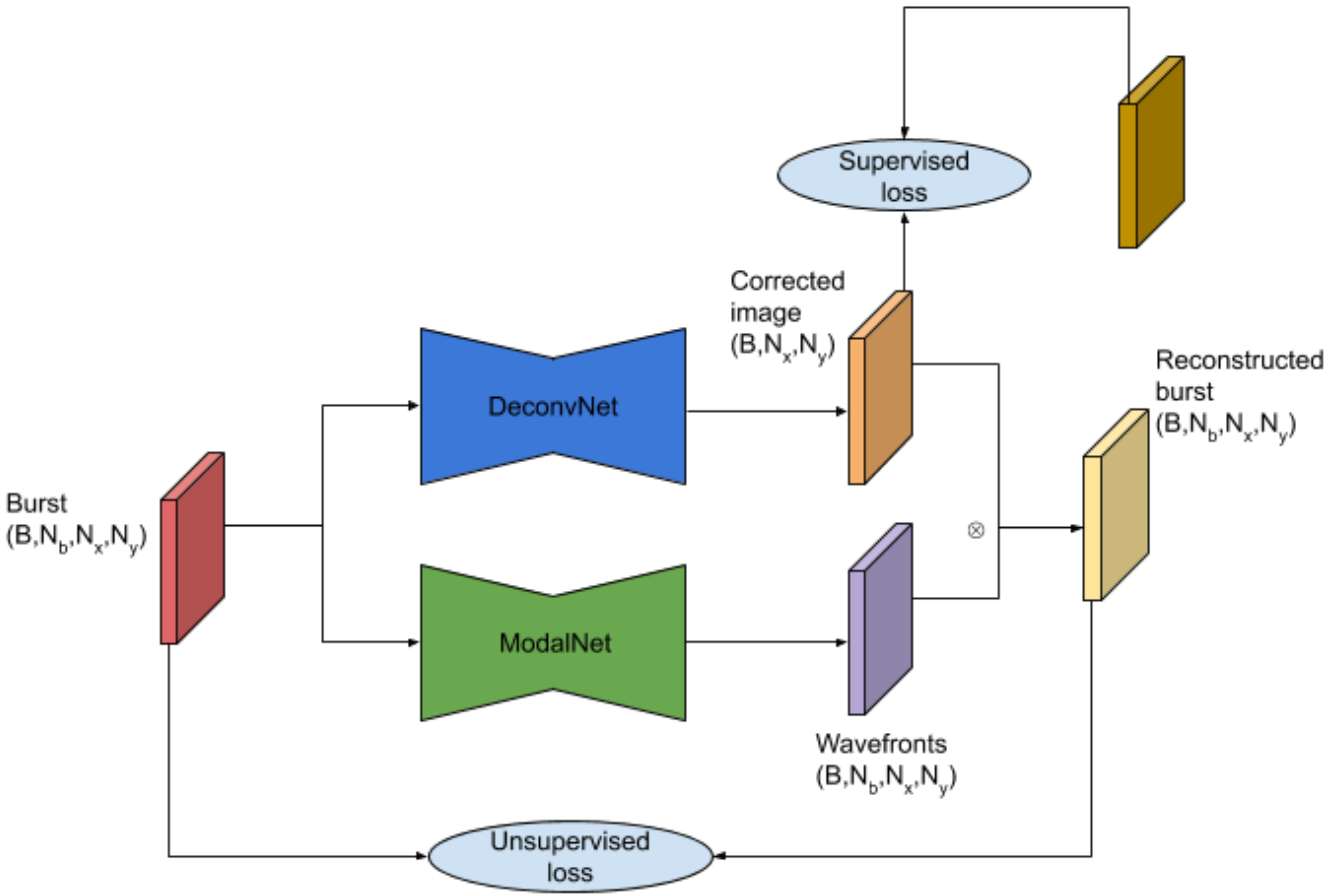


100 images/s

WIP : UNSUPERVISED TRAINING



WIP : UNSUPERVISED TRAINING



fast inversion of Stokes profiles

CLASSICAL INVERSION OF STOKES PROFILES

T, B, v, ...

$$\mathbf{S}(\lambda) = f(\mathbf{p}, \lambda)$$

Stokes vector

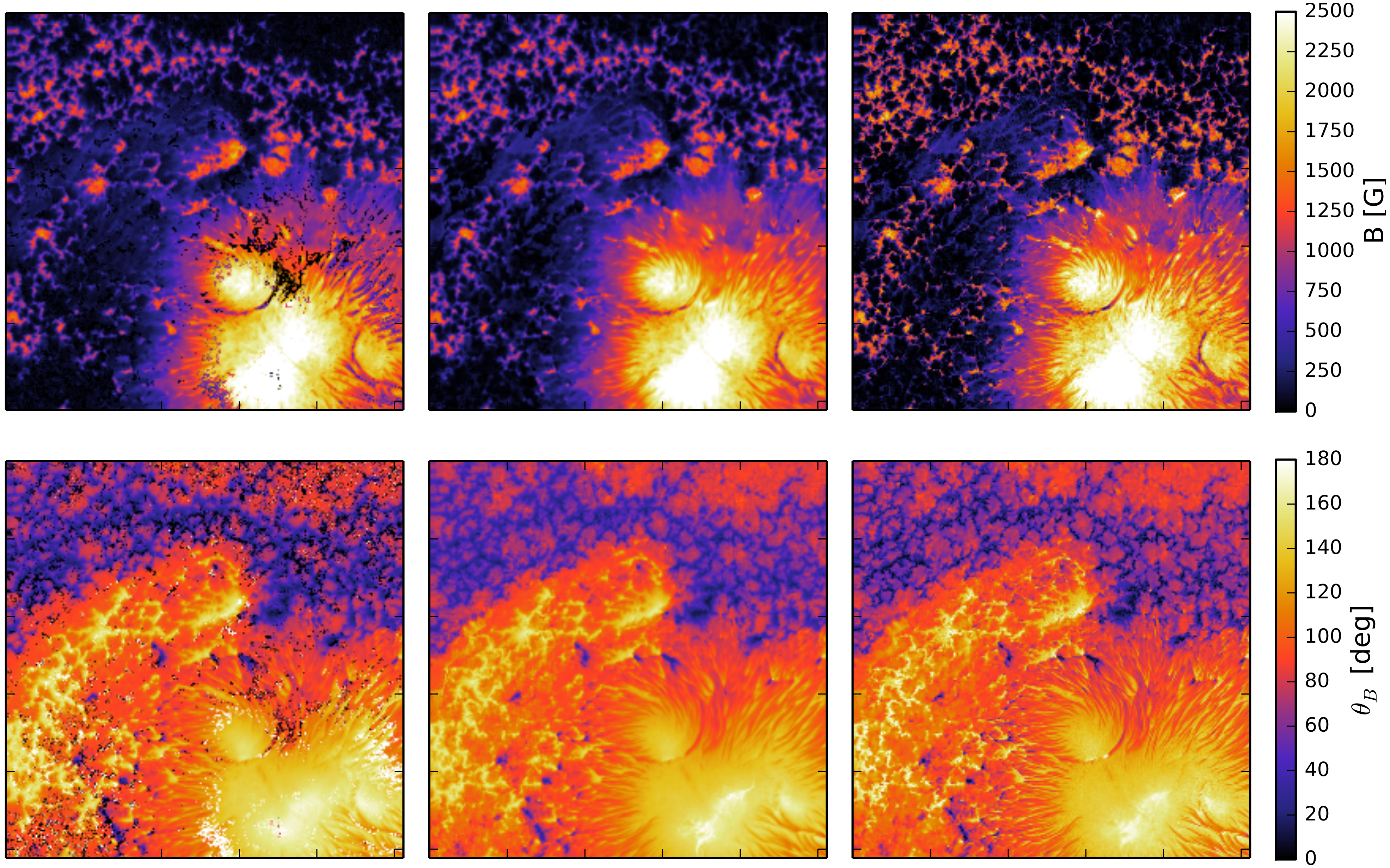
Radiative transfer+QM

$$L = \sum_{ij} \left[S_i(\lambda_j) - f_i(\mathbf{p}, \lambda_j)\right]^2$$

- Optimized with Levenberg-Marquardt
- Gradients are difficult to compute (non-linear + non-local forward)

SPARSITY CONSTRAINTS

$$L = \sum_{ij} \left[S_i(\lambda_j) - f_i(\mathbf{p}, \lambda_j) \right]^2 + \lambda \| \mathbf{W}^T \mathbf{p} \|_0$$



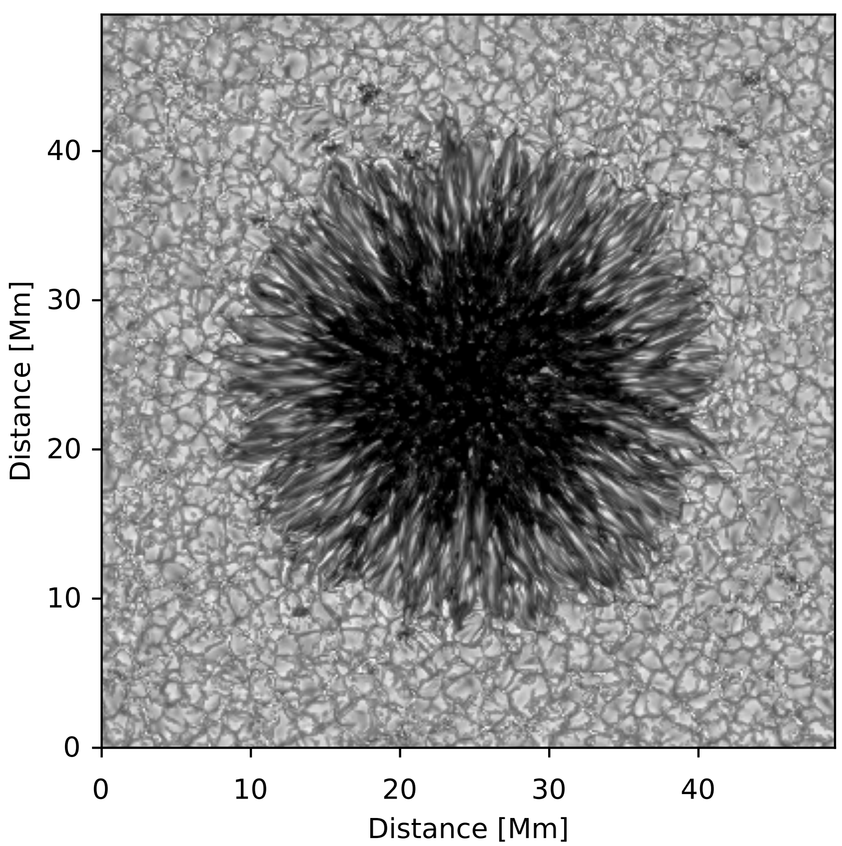
Pixel-by-pixel | Sparse | Sparse+deconv

Asensio Ramos & de la Cruz Rodríguez (2013)

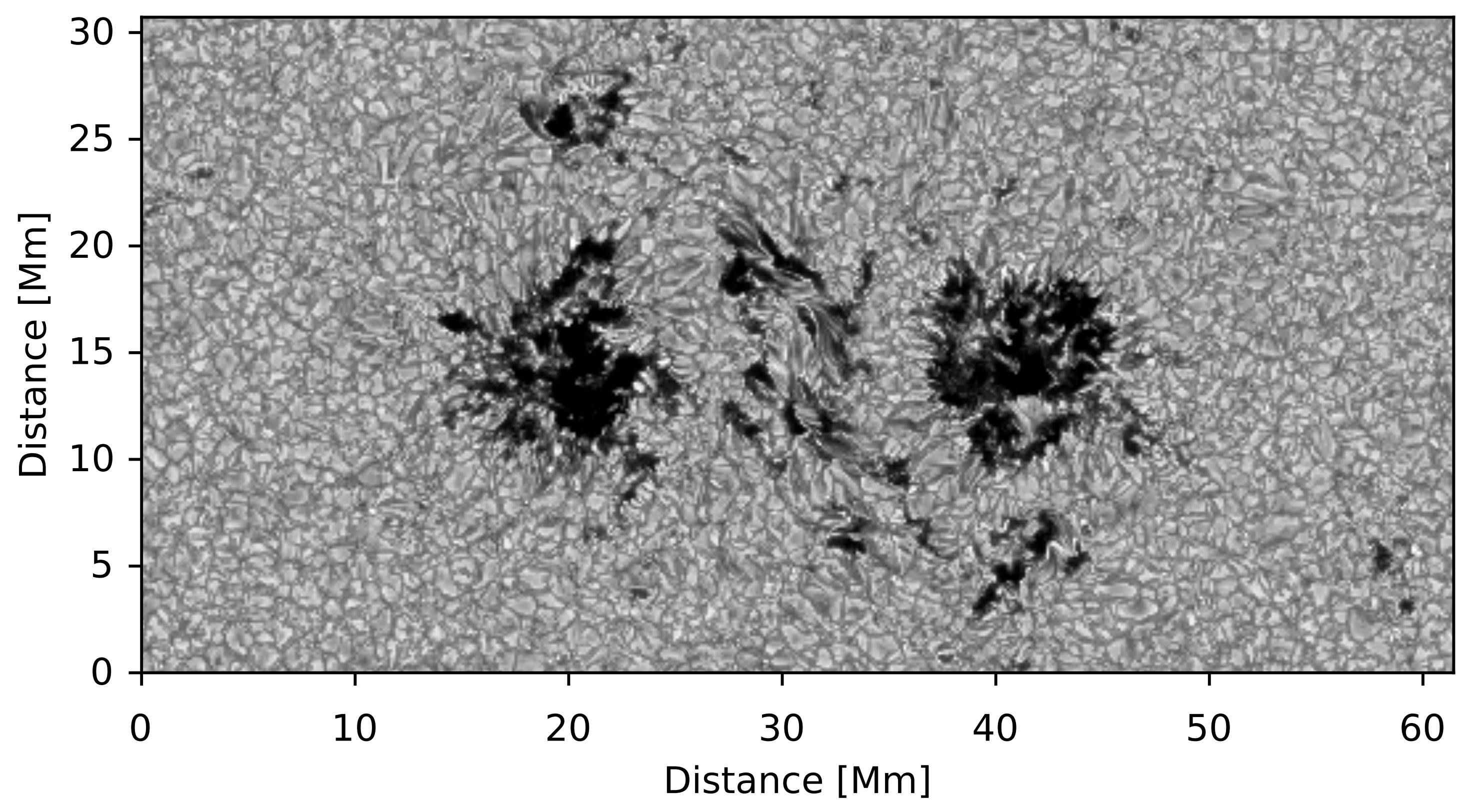
CAN WE TRAIN END-TO-END?

TRAINING SETS

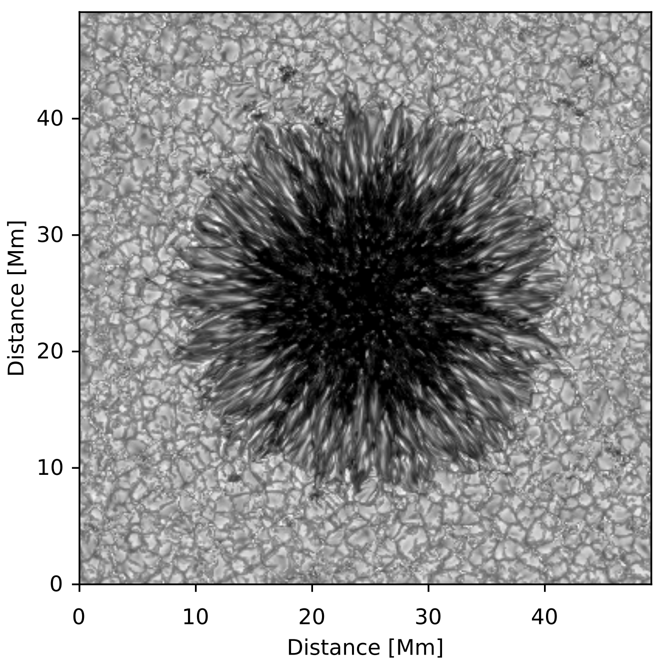
Rempel et al. (2012)



Cheung et al. (2010)



DEGRADING TRAINING SETS



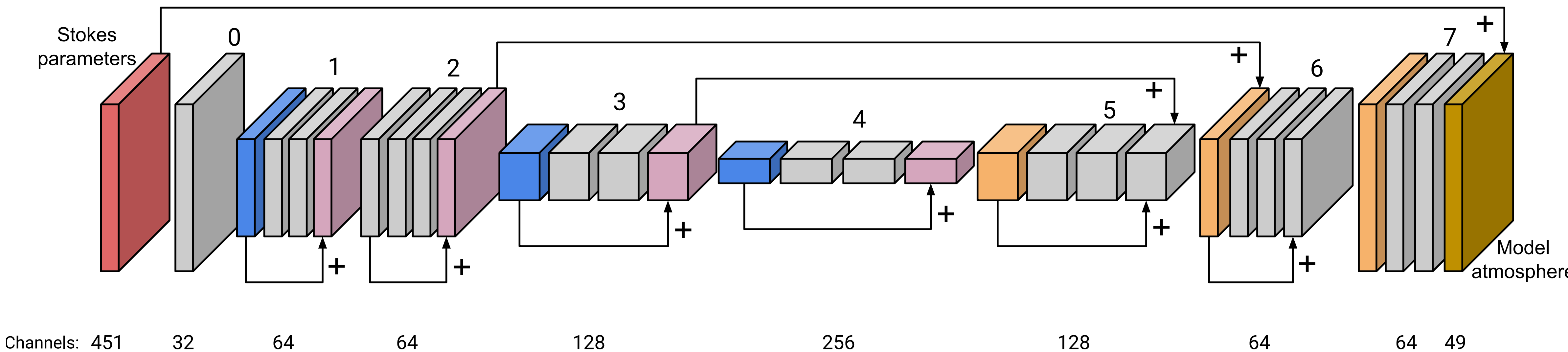
Stokes profiles

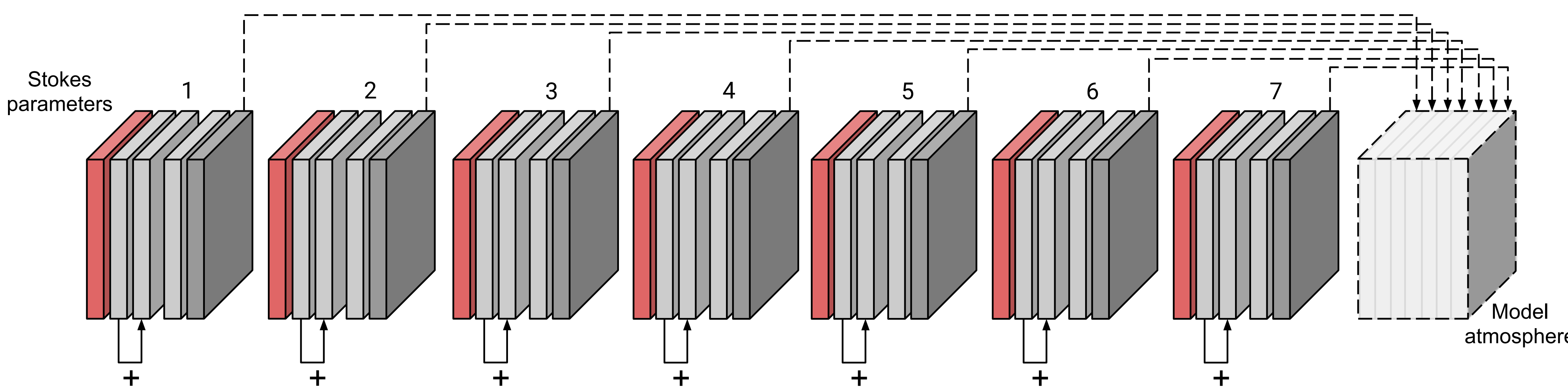
LTE SPECTRAL SYNTHESIS → SPATIAL SMEARING → SPECTRAL SMEARING → SPATIAL BINNING

OPTICAL DEPTH SURFACES → SPATIAL BINNING

Physical models

ARCHITECTURES

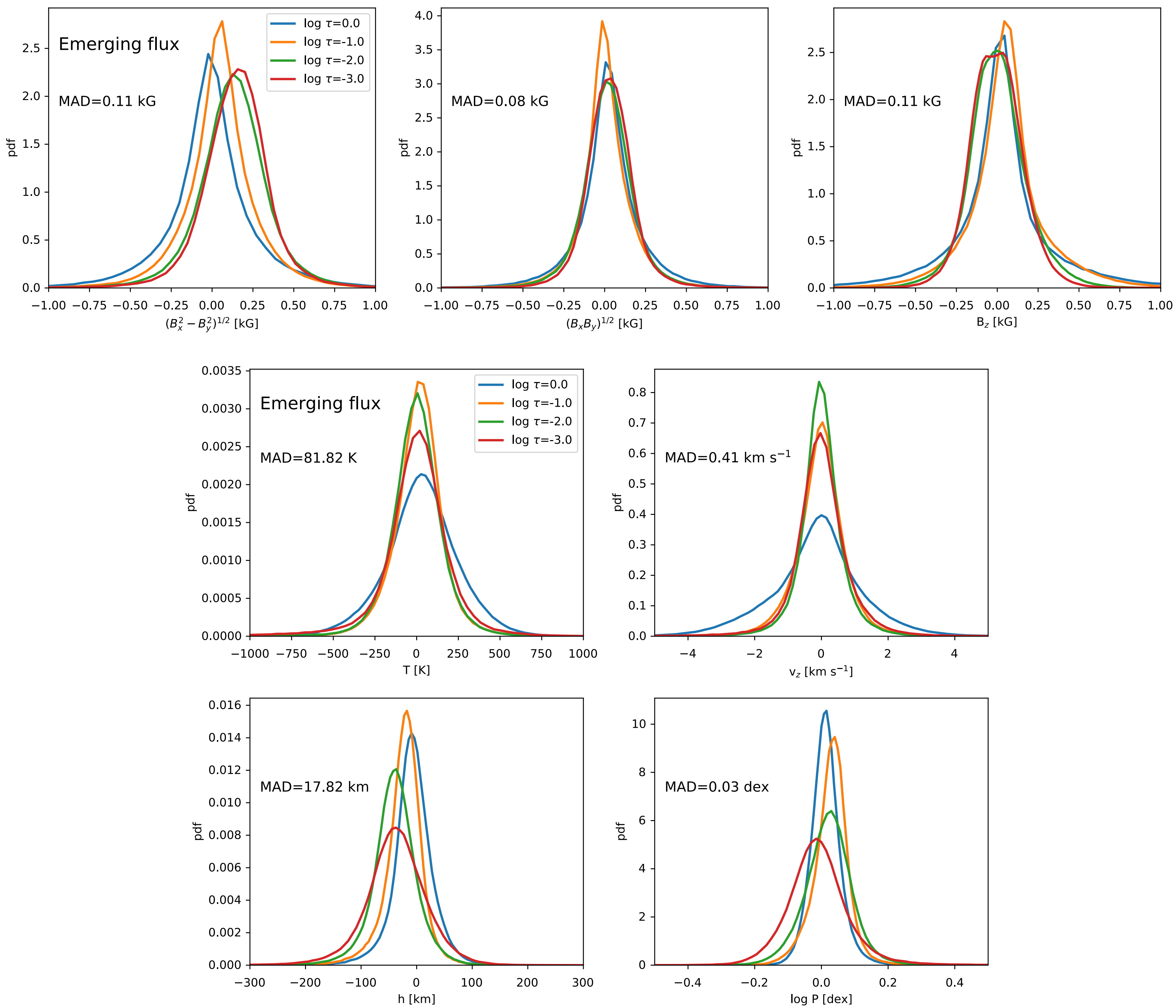




180 ms per inversion on 512x512

30 minutes for all Hinode observations

VALIDATION



Emerging flux
MAD=0.11 kG
log τ=0.0
log τ=-1.0
log τ=-2.0
log τ=-3.0
pdf
$(B_x^2 - B_y^2)^{1/2}$ [kG]
MAD=0.08 kG
$(B_x B_y)^{1/2}$ [kG]
MAD=0.11 kG
B$_z$ [kG]
Emerging flux
MAD=81.82 K
T [K]
MAD=0.41 km s^{-1}
v$_z$ [km s^{-1}]
MAD=17.82 km
h [km]
MAD=0.03 dex
log P [dex]

AR10933 : CONTINUUM

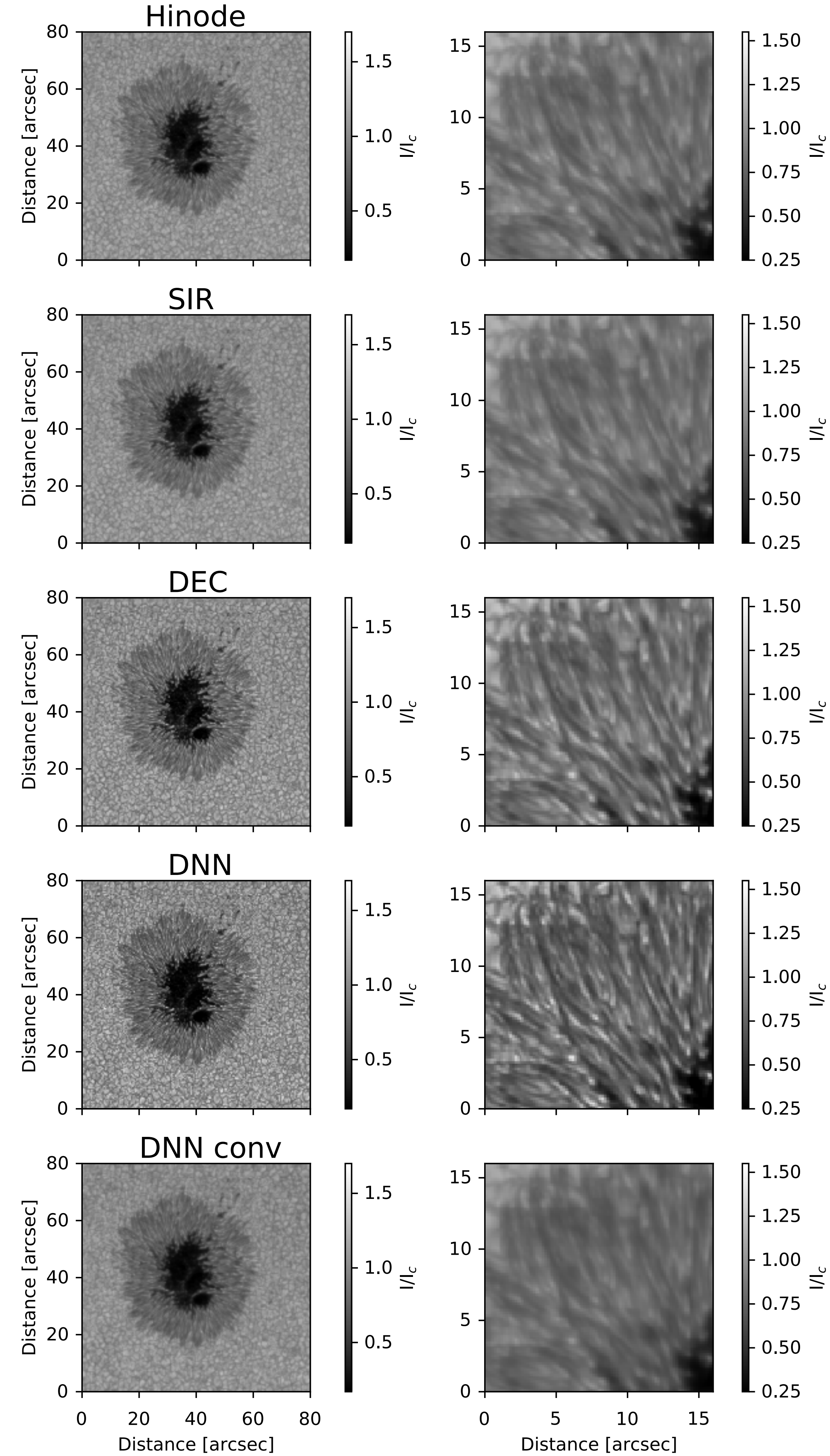
Original

SIR inversions

SIR inversions+deconvolution

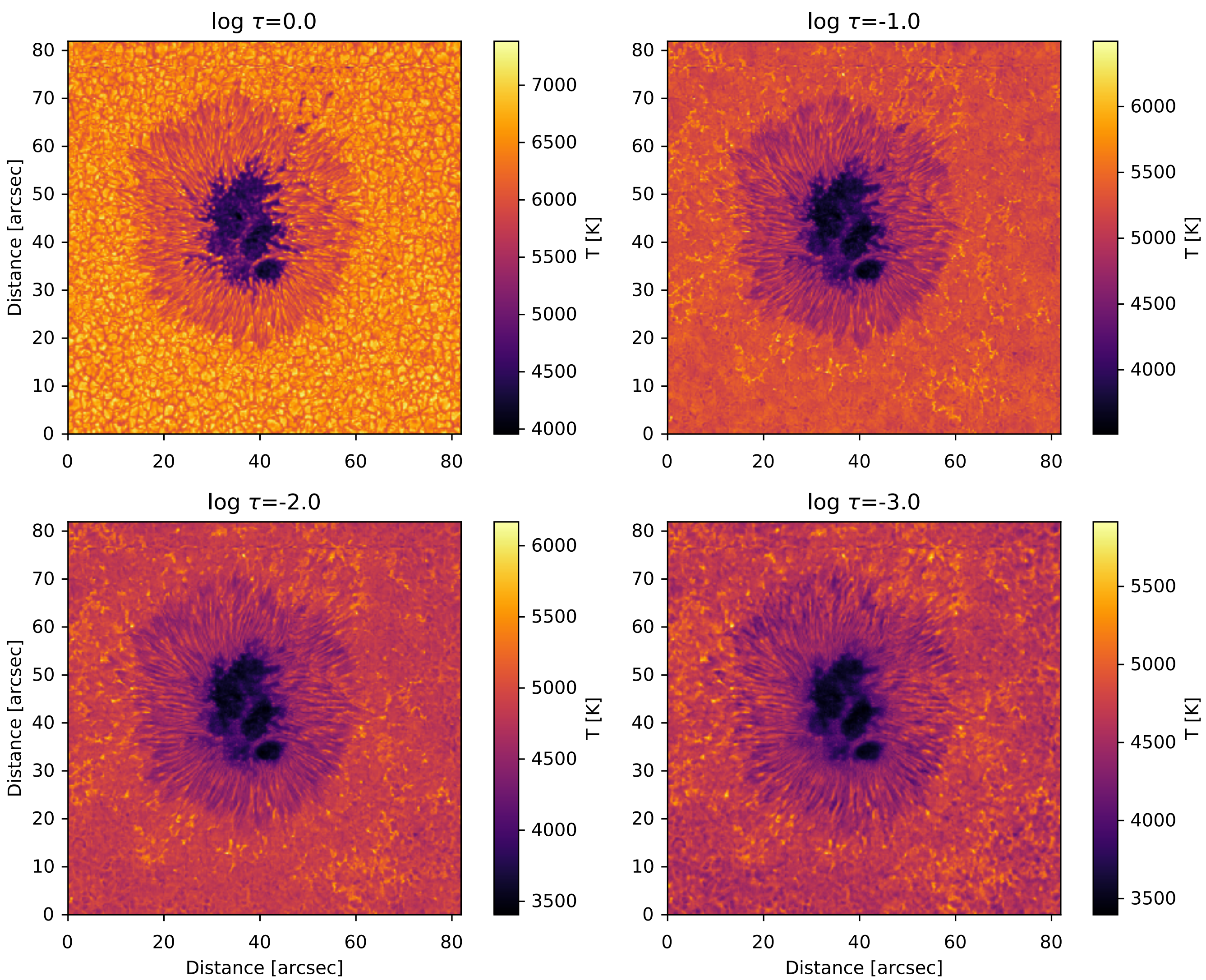
Deep neural network

Deep neural network+convolution

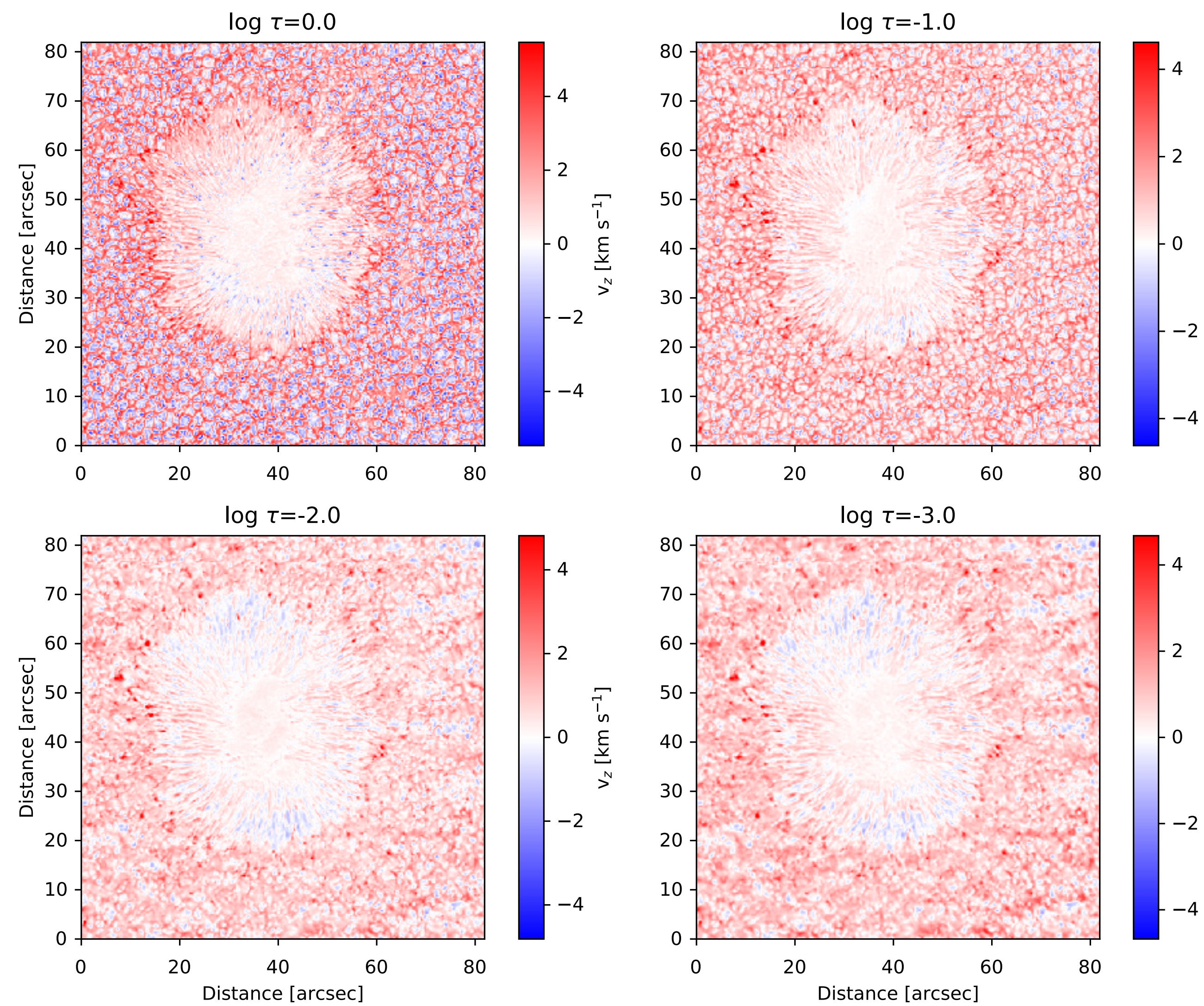


AR10933 : INFERENCE

Temperature



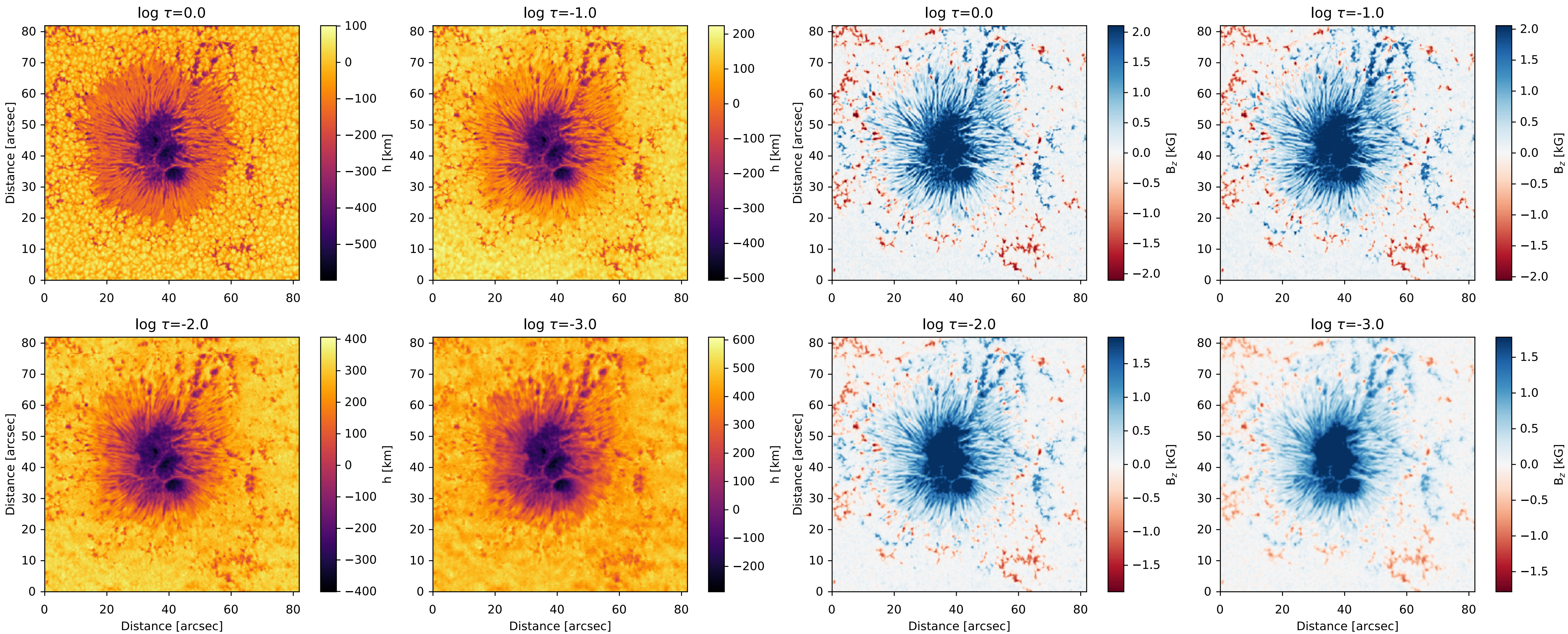
Doppler velocity



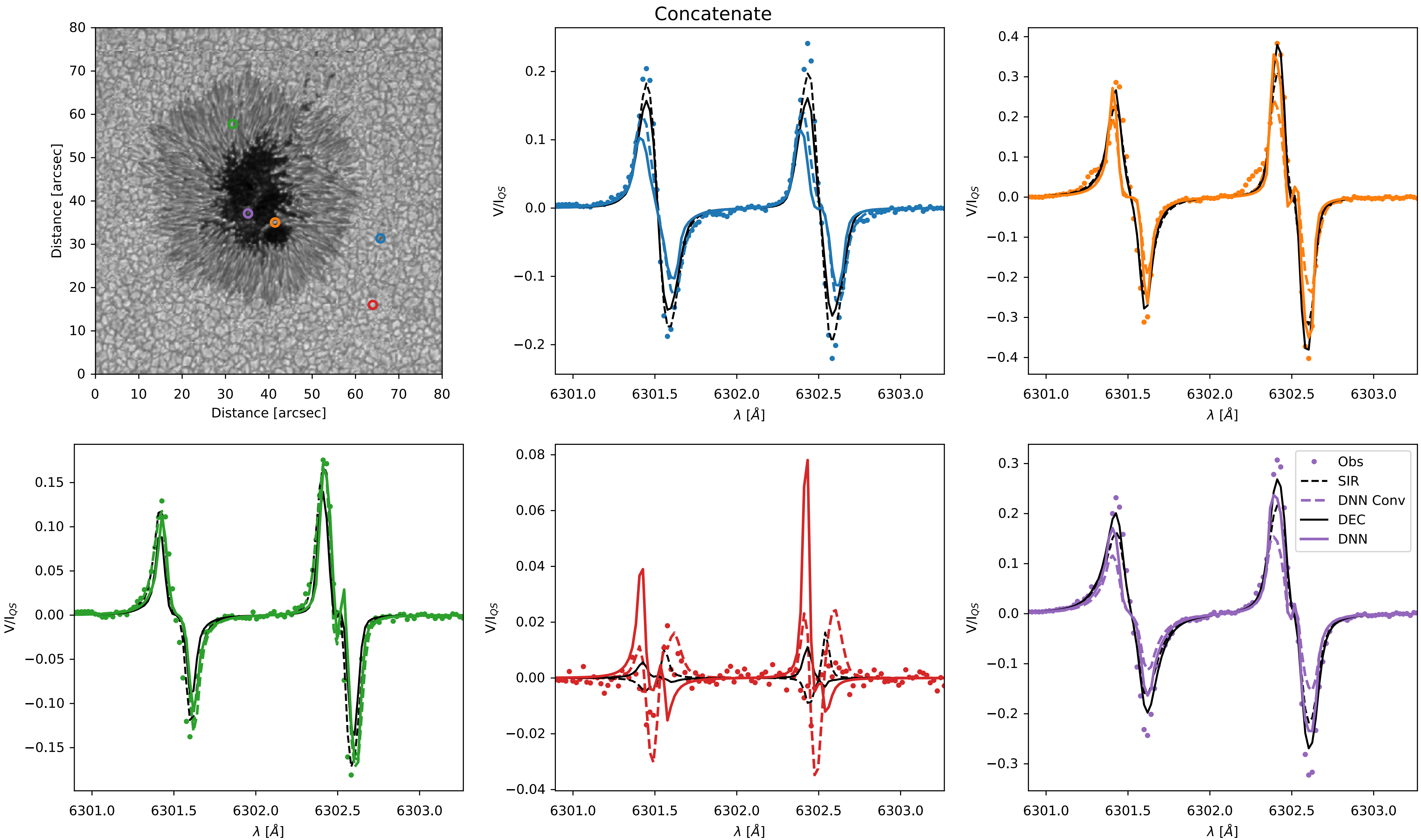
AR10933 : INFERENCE

τ surfaces

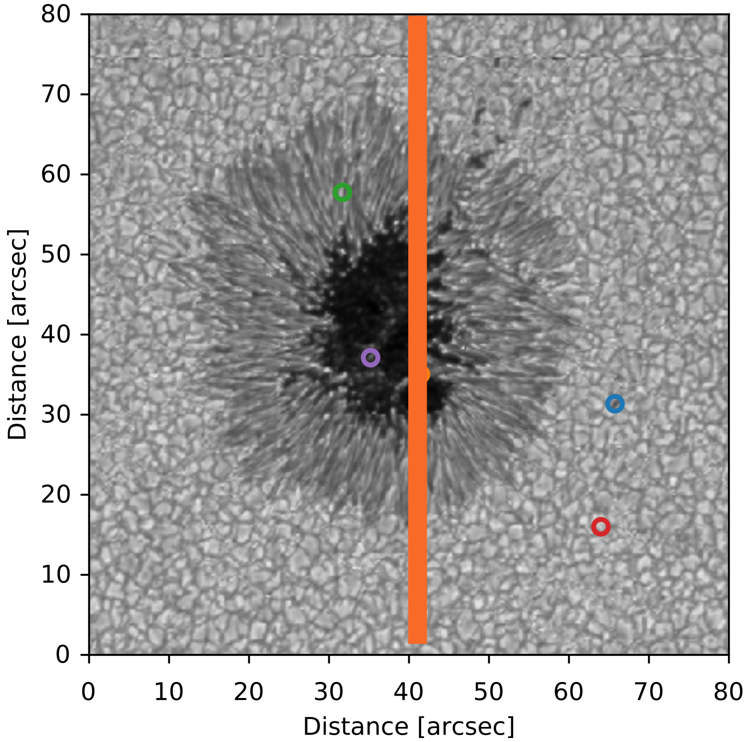
Bz

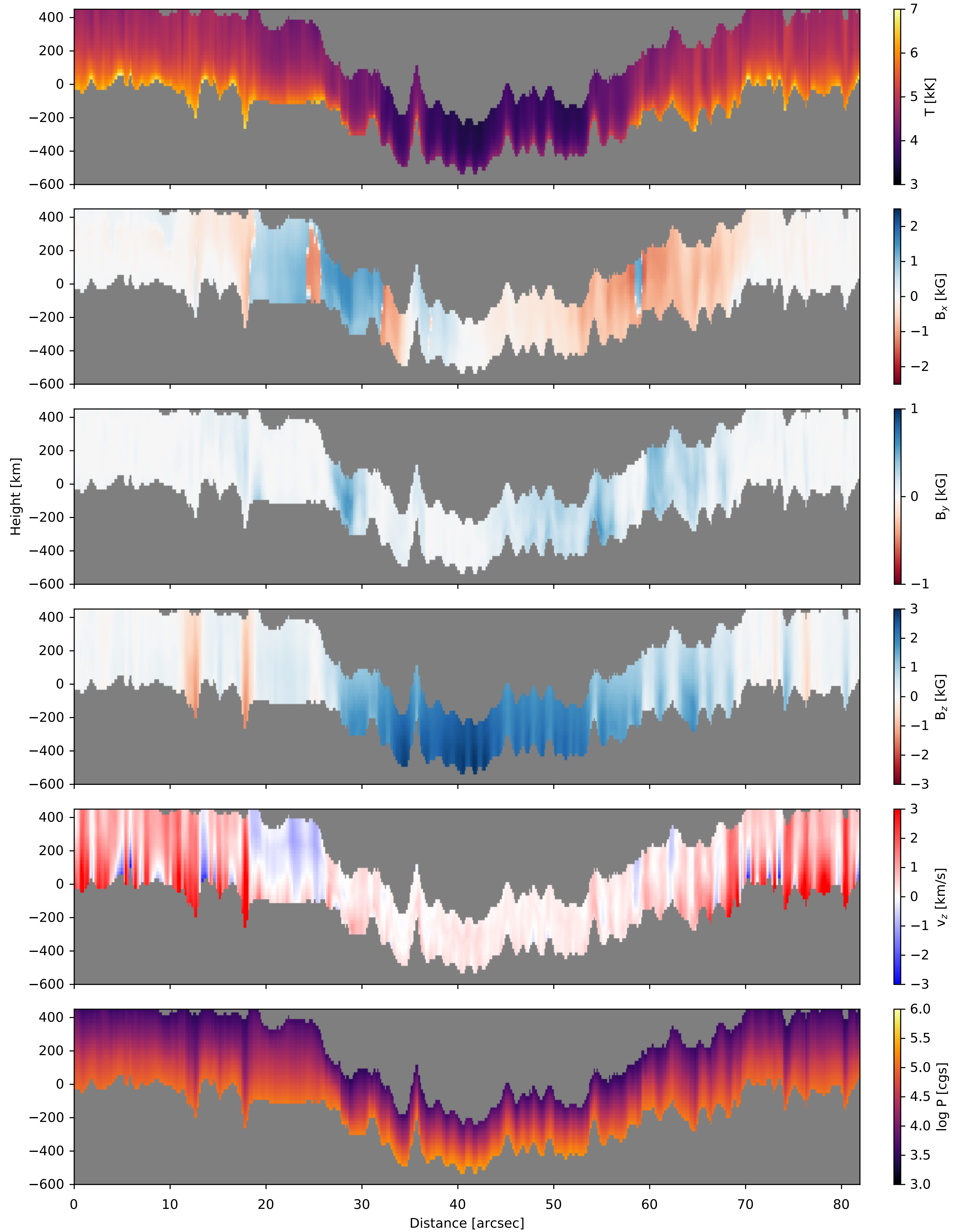


DO WE FIT THE PROFILES?



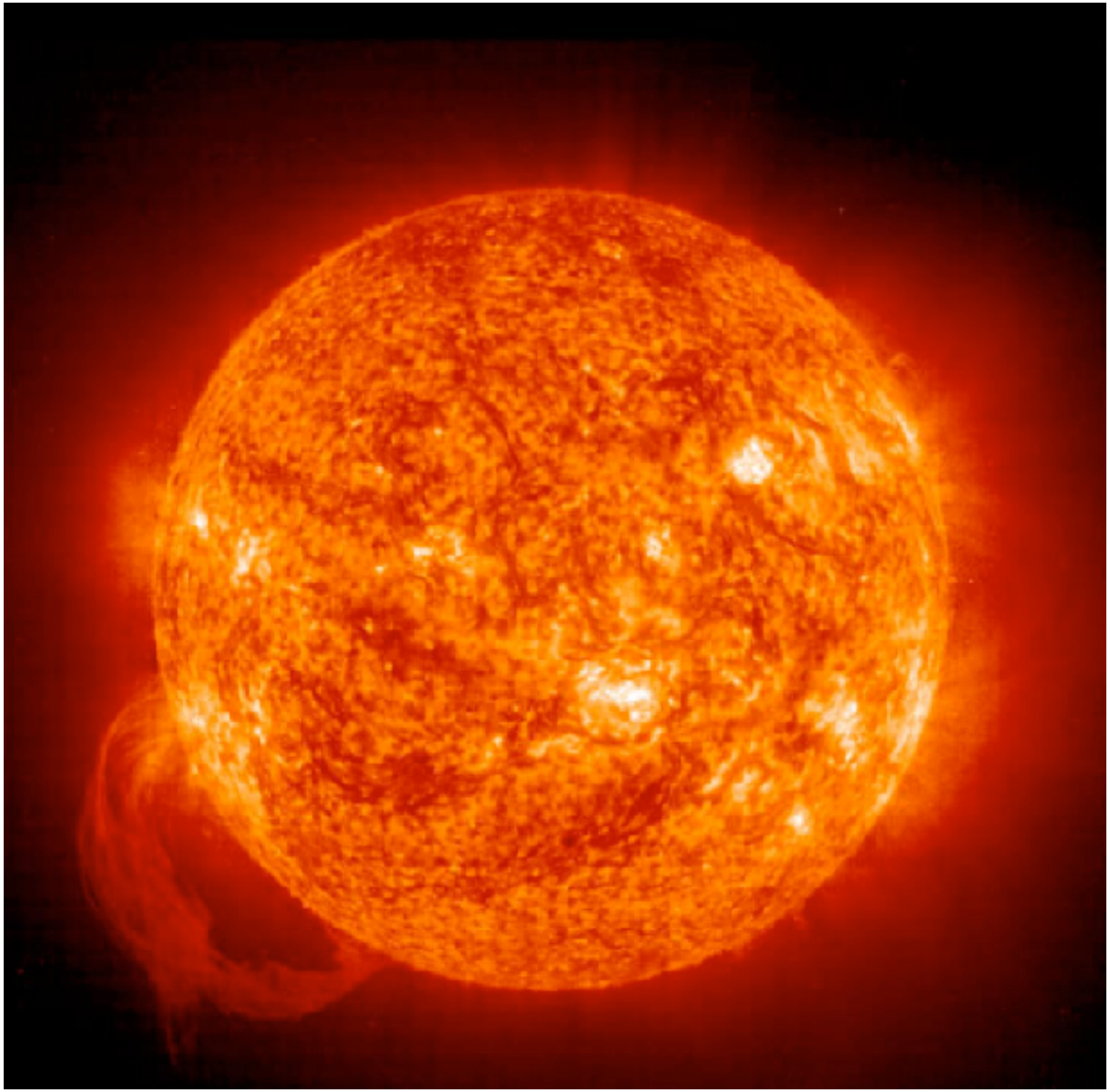
LIGHT BRIDGE





farside enhacement

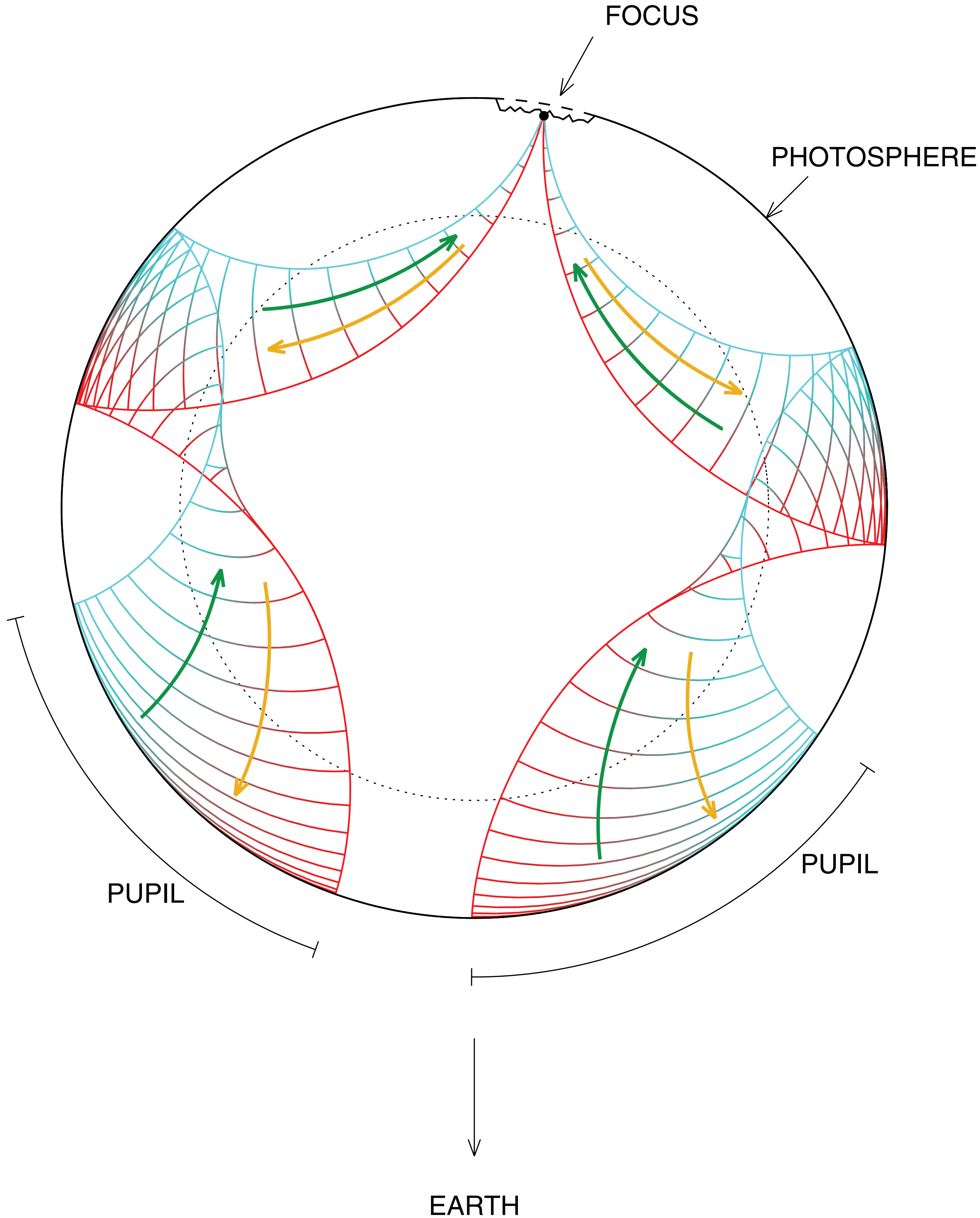
FARSIDE PROBLEM



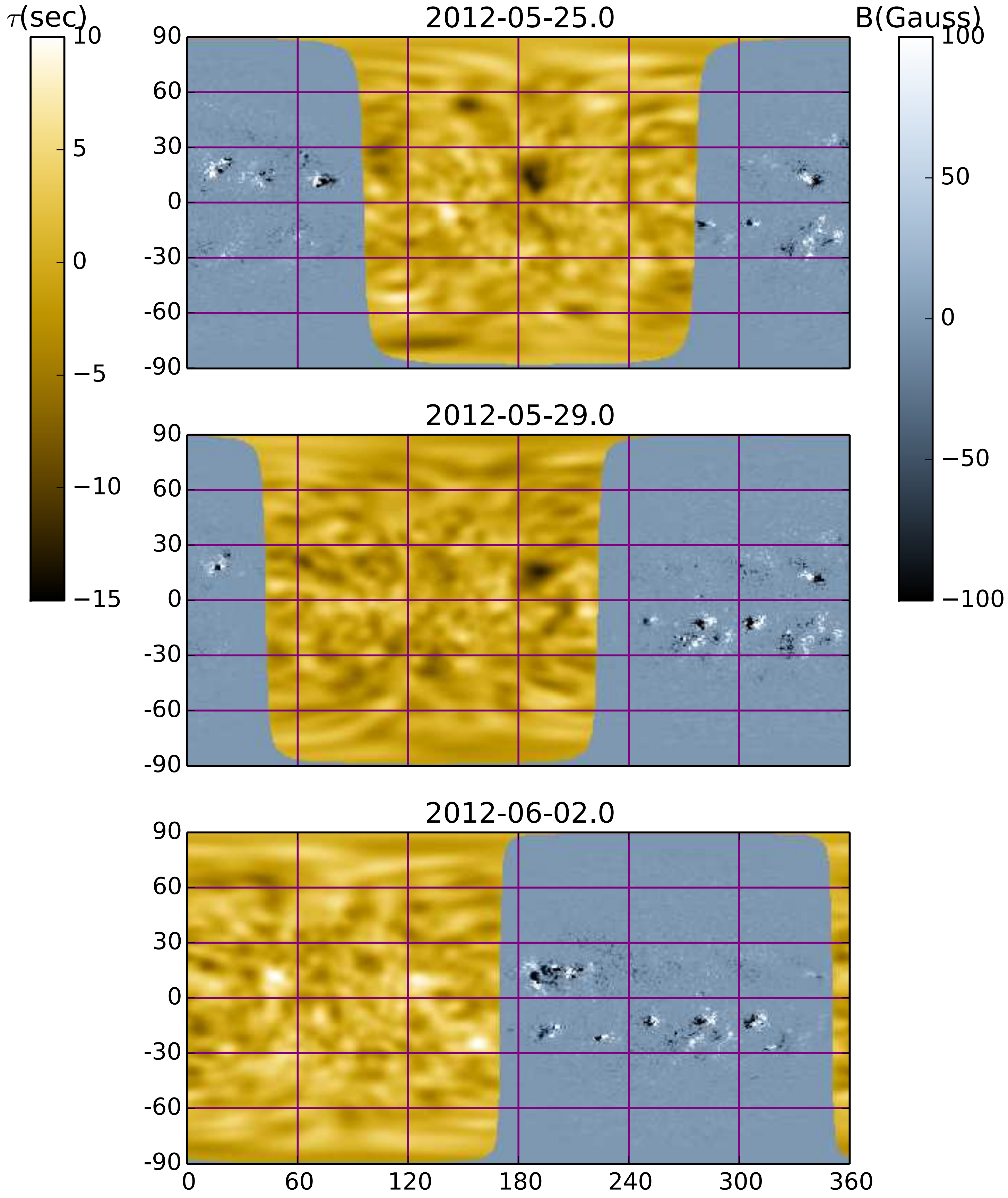
Forecasting

- Solar UV irradiance
- Global solar magnetic index
- Coronal magnetic field
- ...

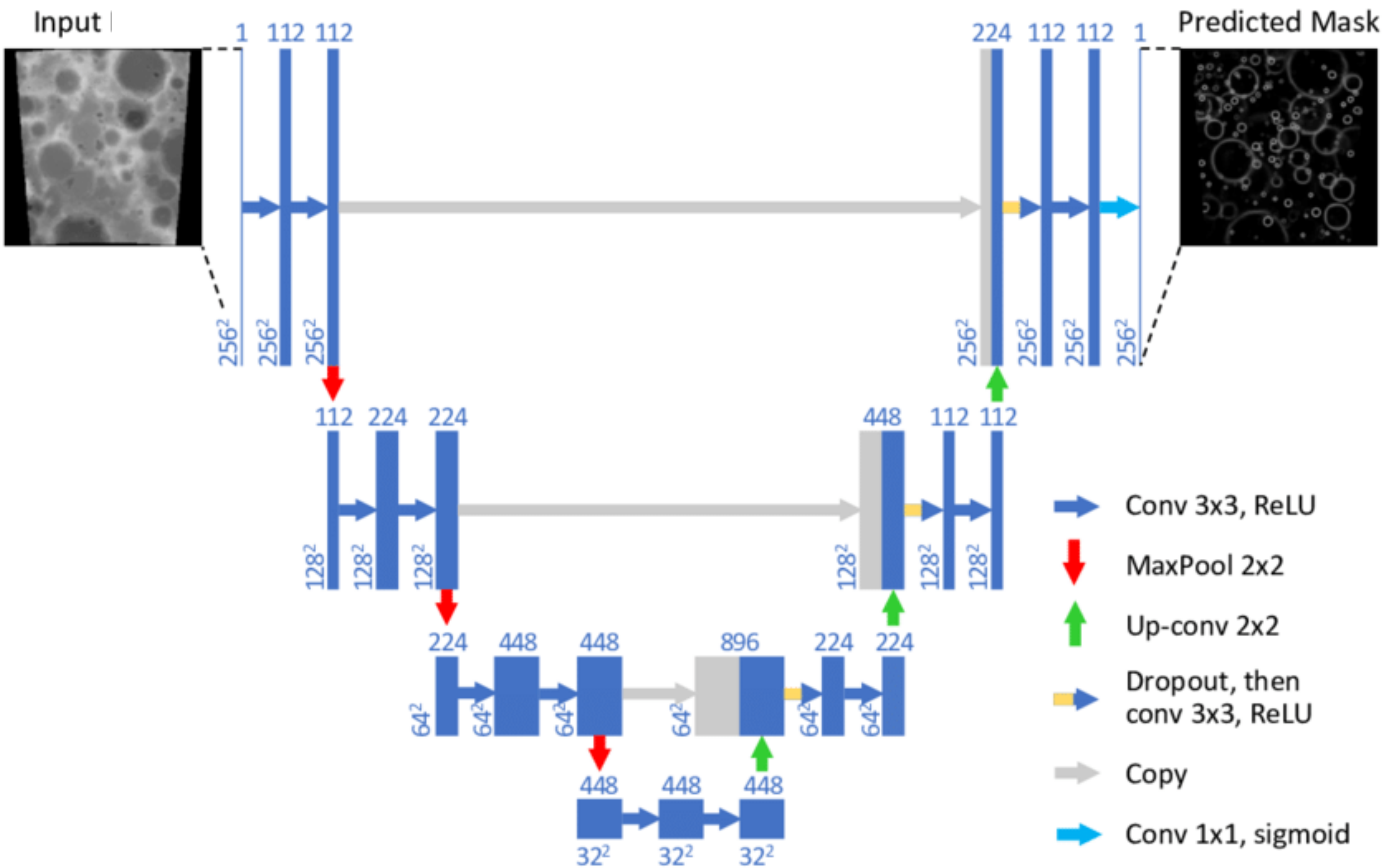
FARSIDE PROBLEM



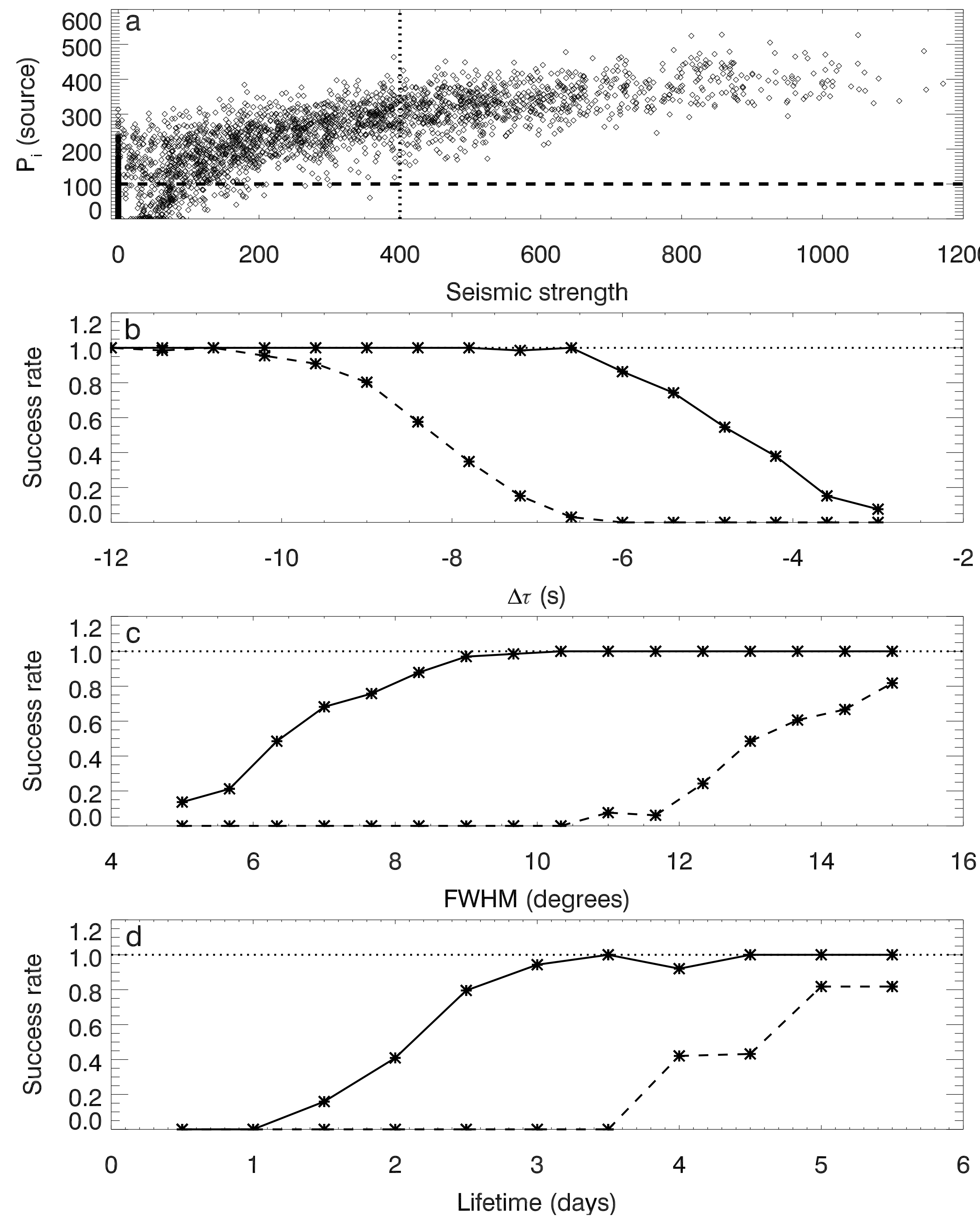
CURRENT FARSIDE PREDICTIONS



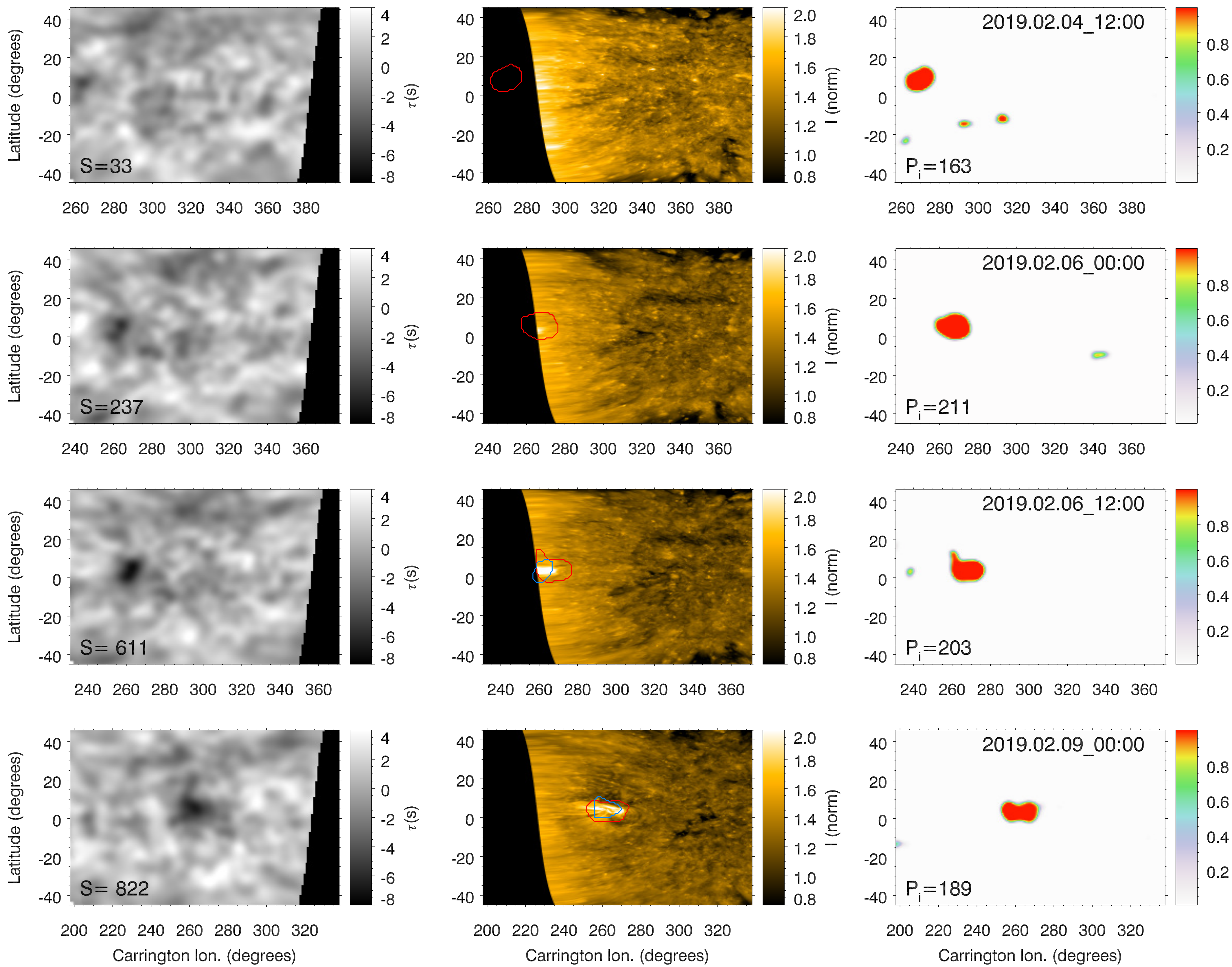
U-NET ARCHITECTURE



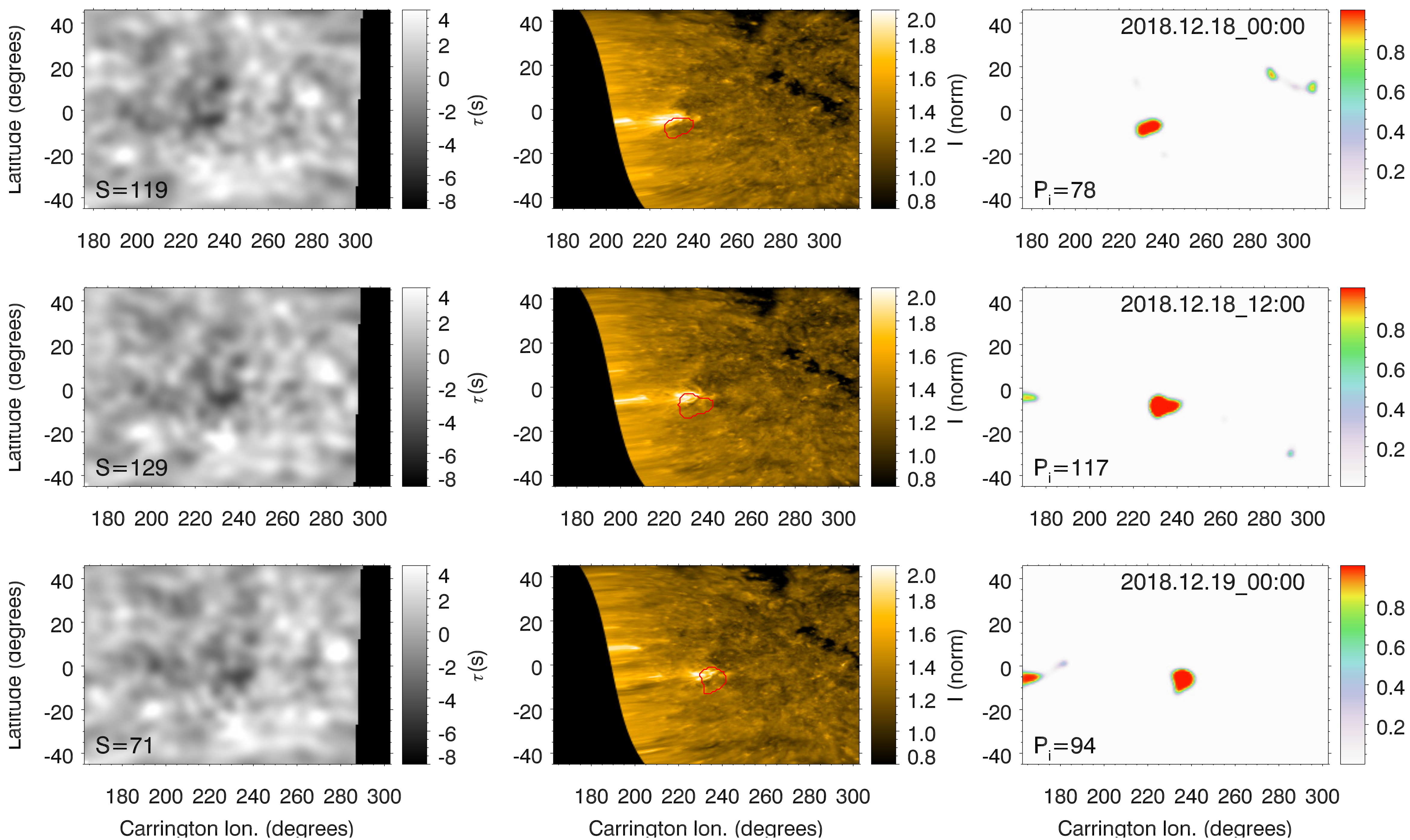
INJECTING ACTIVE REGIONS



OUR PREDICTIONS

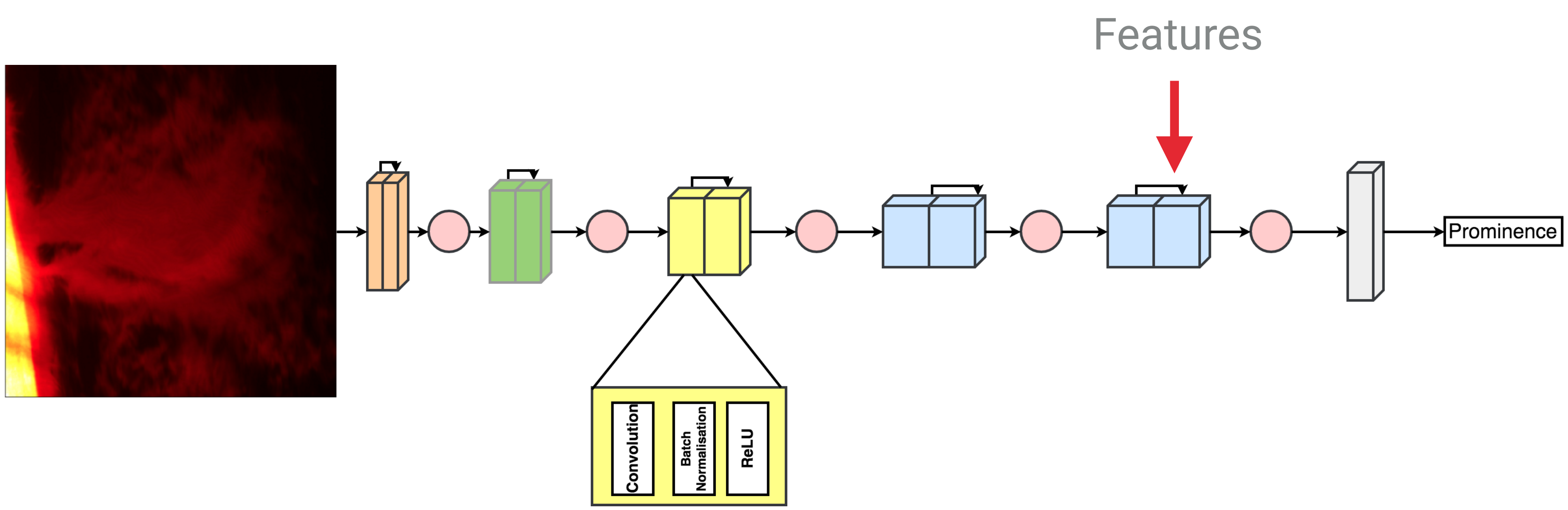


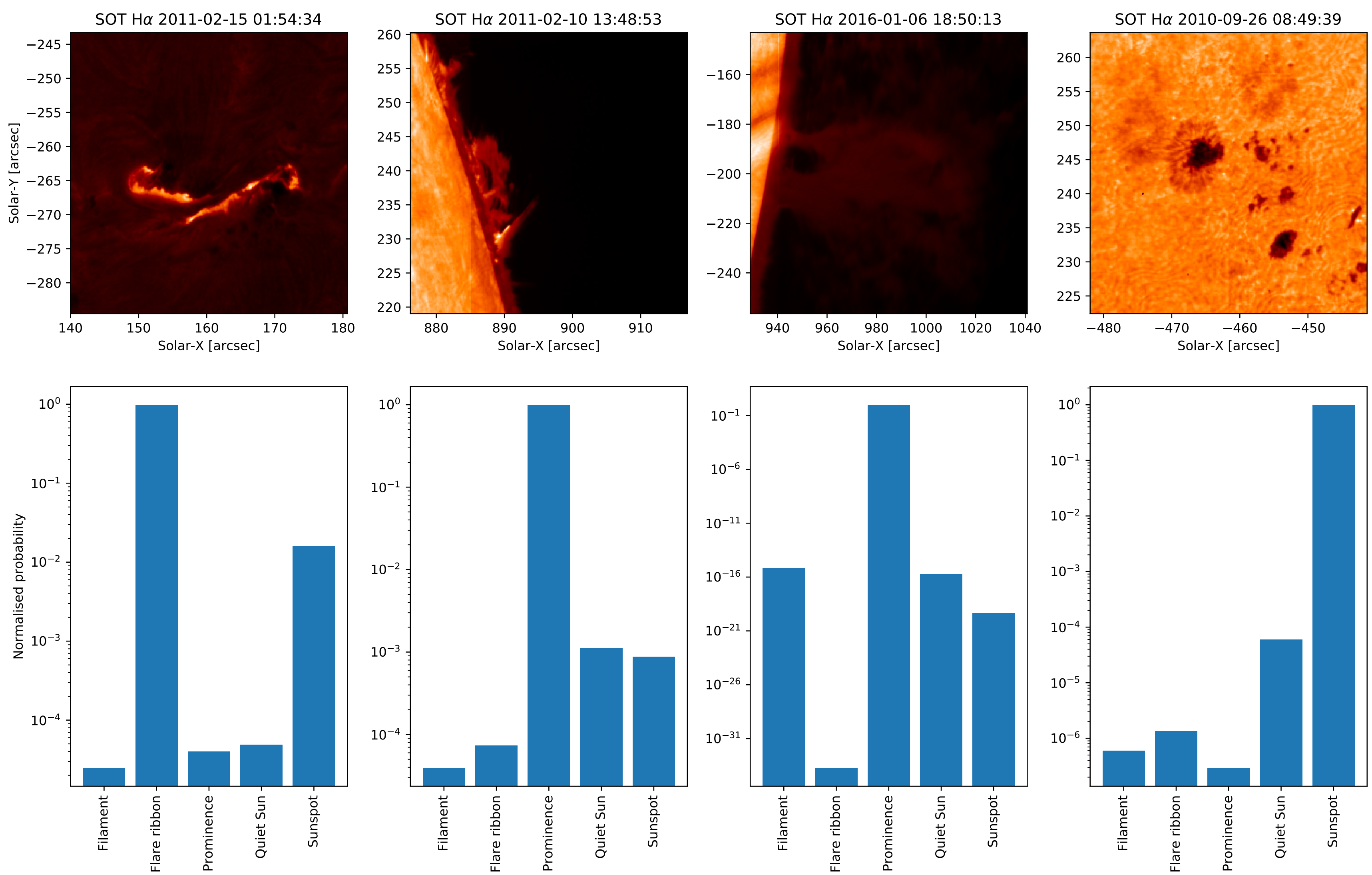
OUR PREDICTIONS



classification of solar structures

CLASSIFICATION

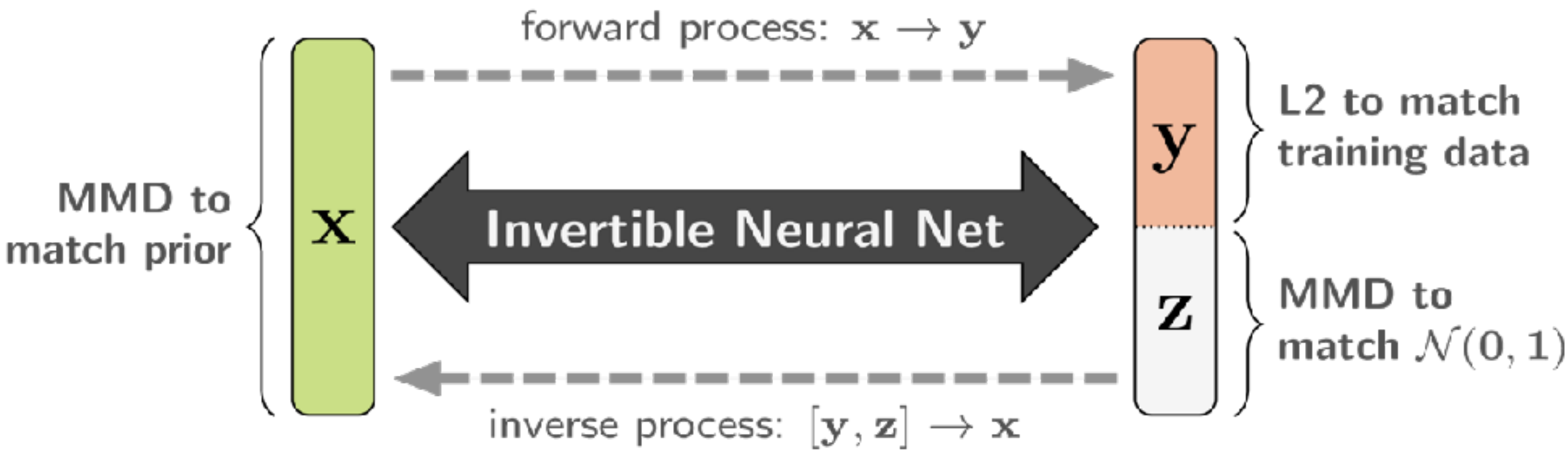




Armstrong & Fletcher (2018)

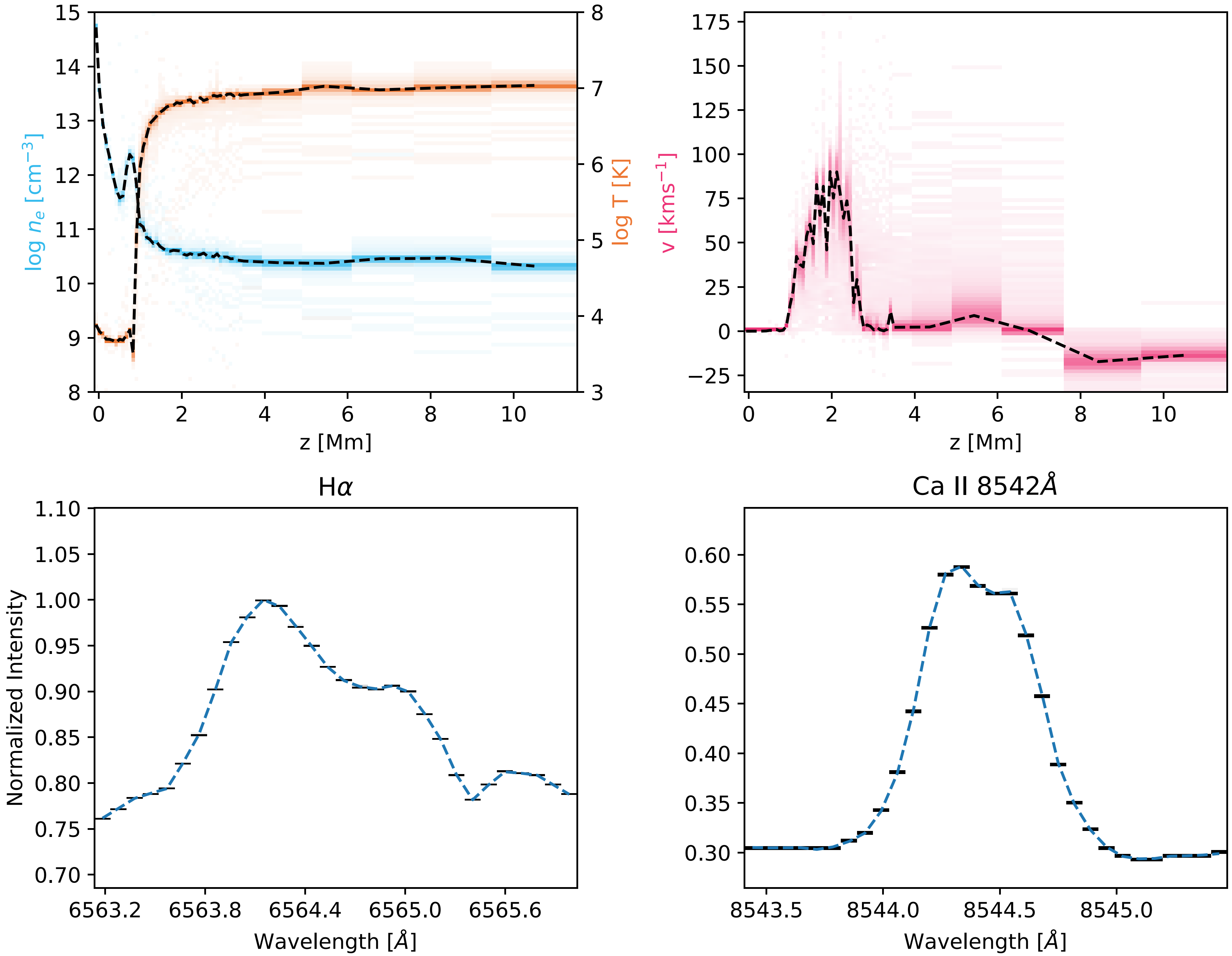
physical conditions in flares

INVERTIBLE NEURAL NETWORKS



Ardizzone et al. (2018)

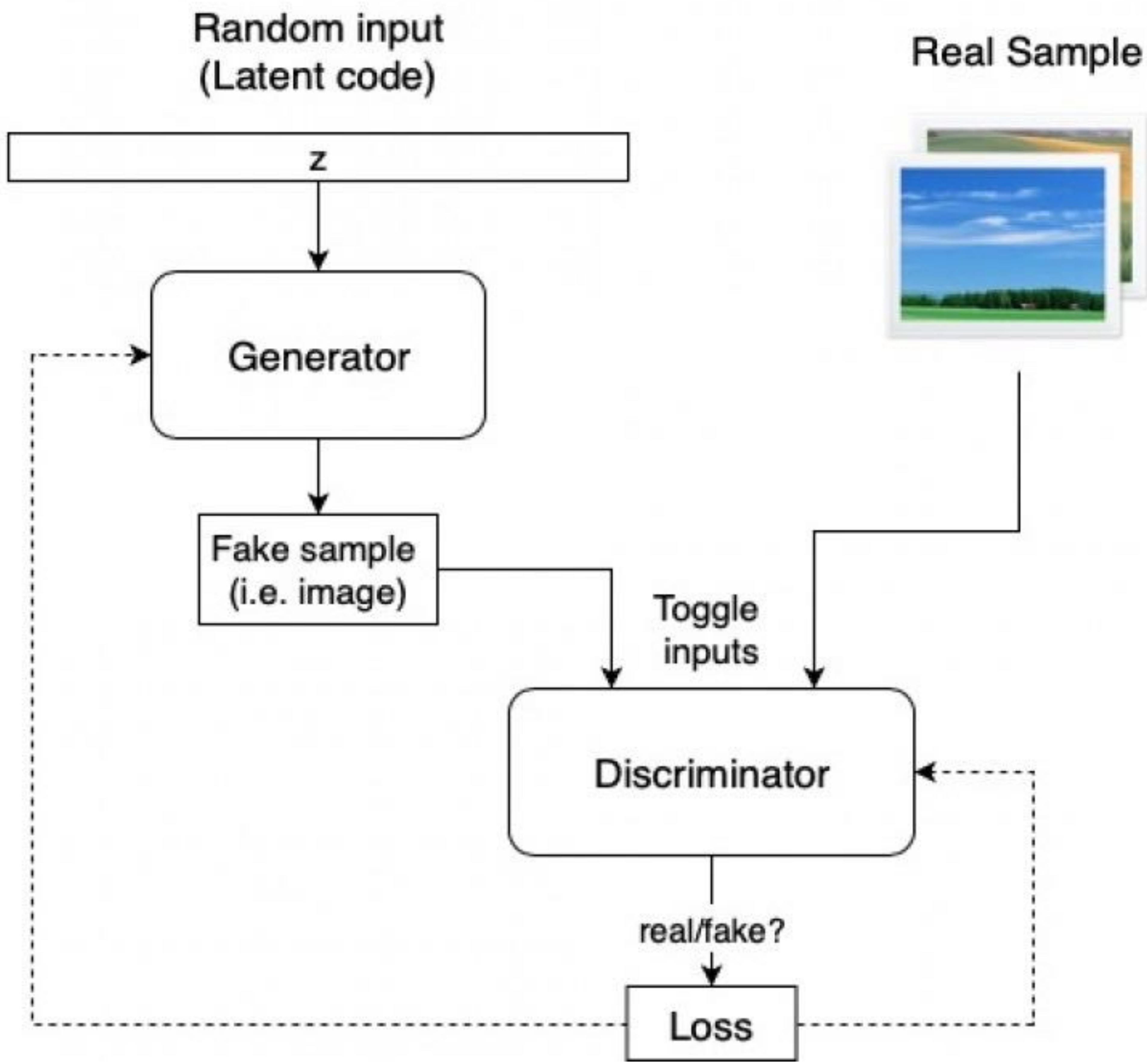
FLARE RIBBON



Osborne, Armstrong & Fletcher (2019)

Other examples…

GENERATIVE ADVERSARIAL NETWORKS

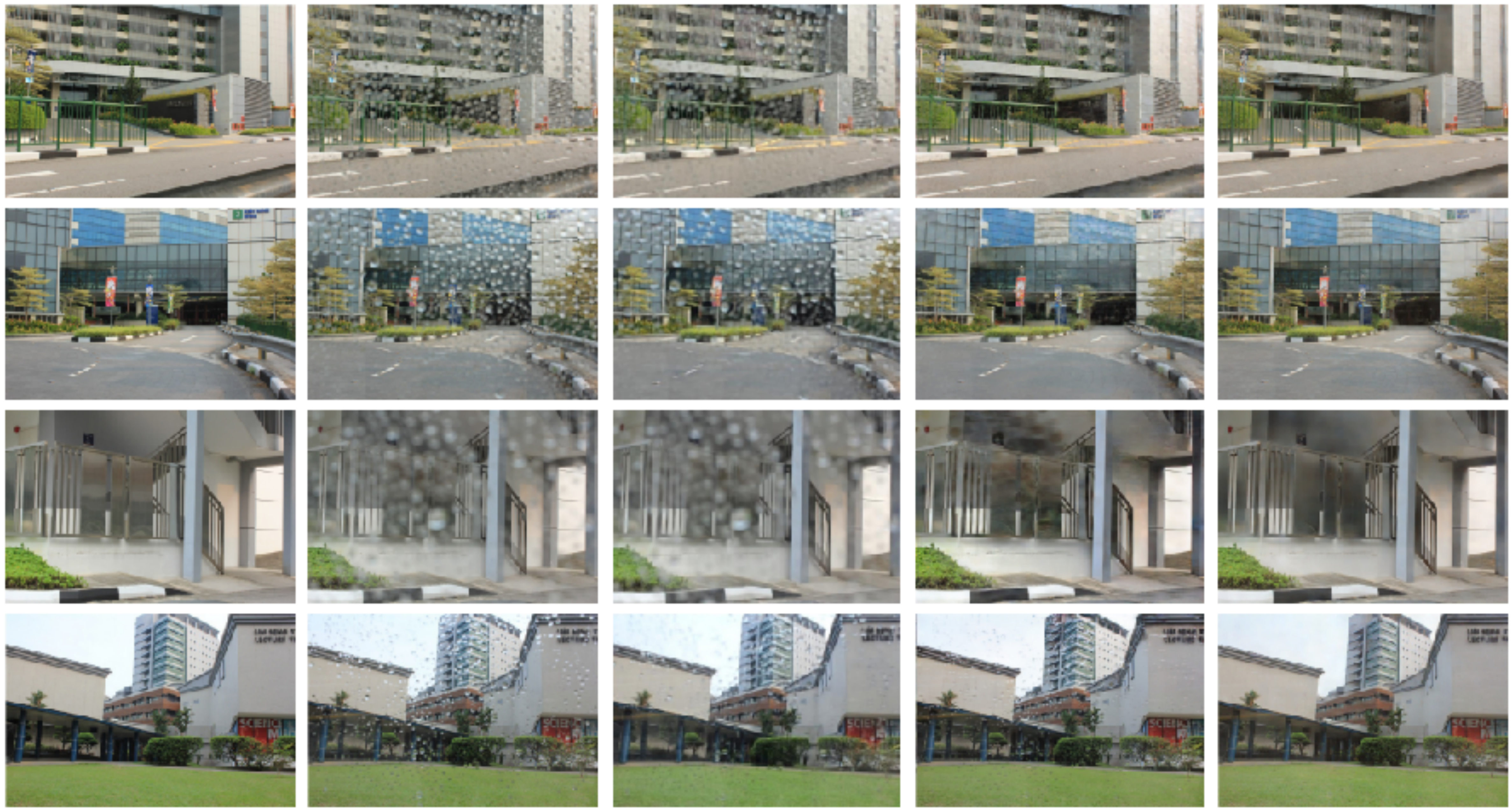


FACES

thispersondoesnotexist.com

SUPER RESOLVE GAMES

RAINDROP REMOVAL

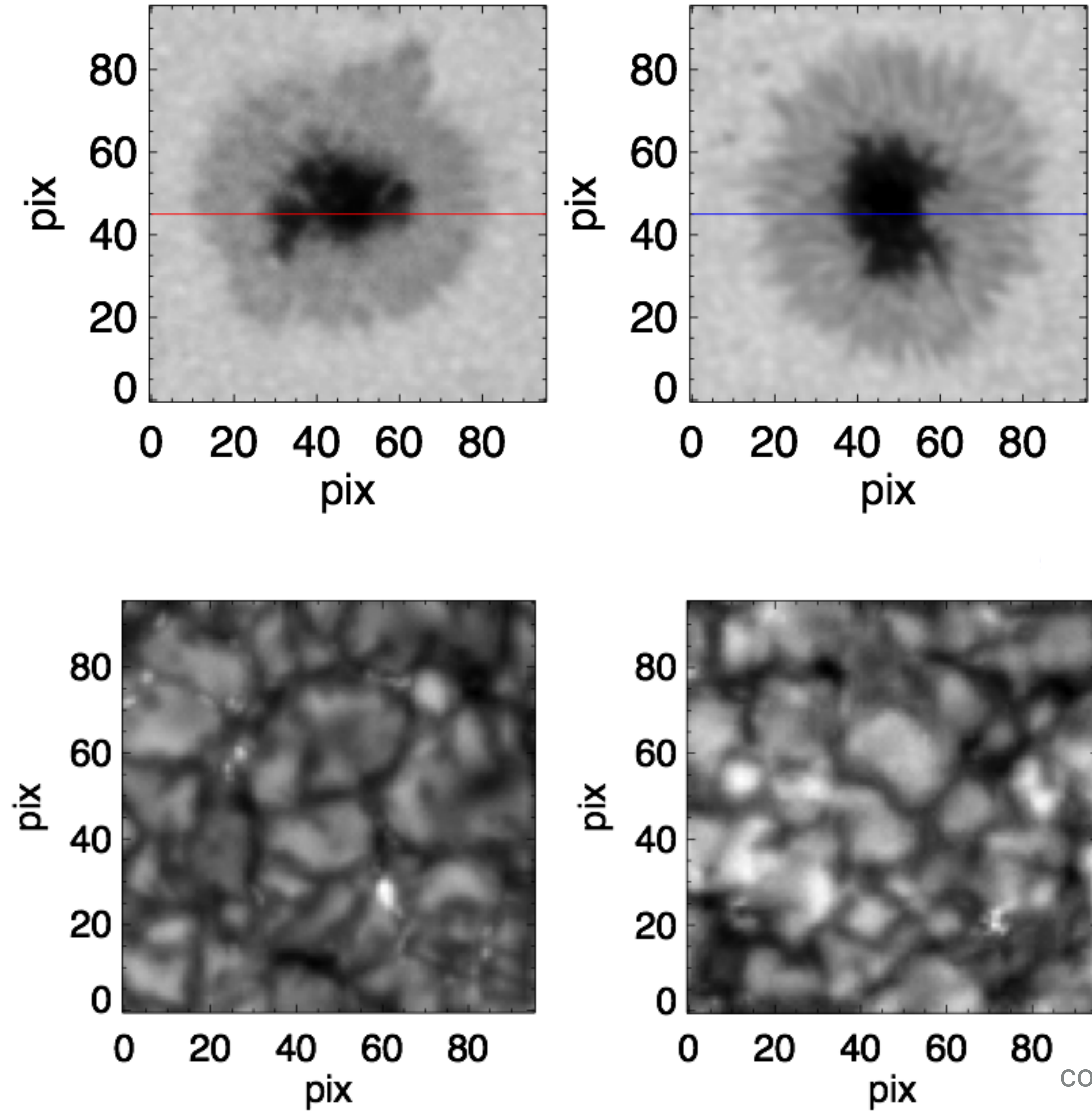


(a) **Ground truth** (b) **Raindrop image** (c) **Eigen[1]** (d) **Pix2pix-cGAN[10]** (e) **Our method**

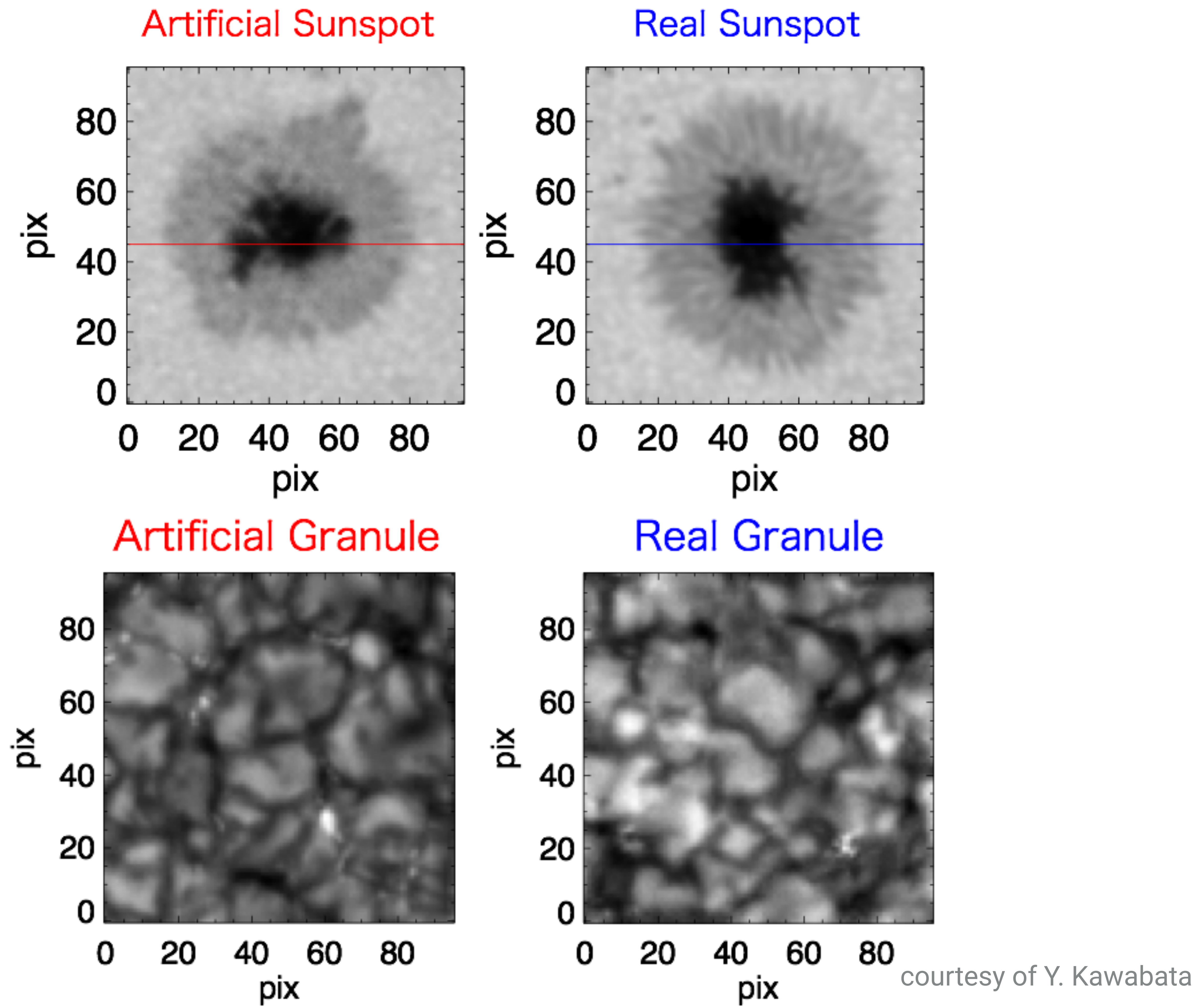
Figure 6. Results of comparing a few different methods. From left to right: ground truth, raindrop image (input), Eigen13 [1], Pix2Pix [10] and our method. Nearly all raindrops are removed by our method despite the diversity of their colors, shapes and transparency.

Quian et al. (2018)

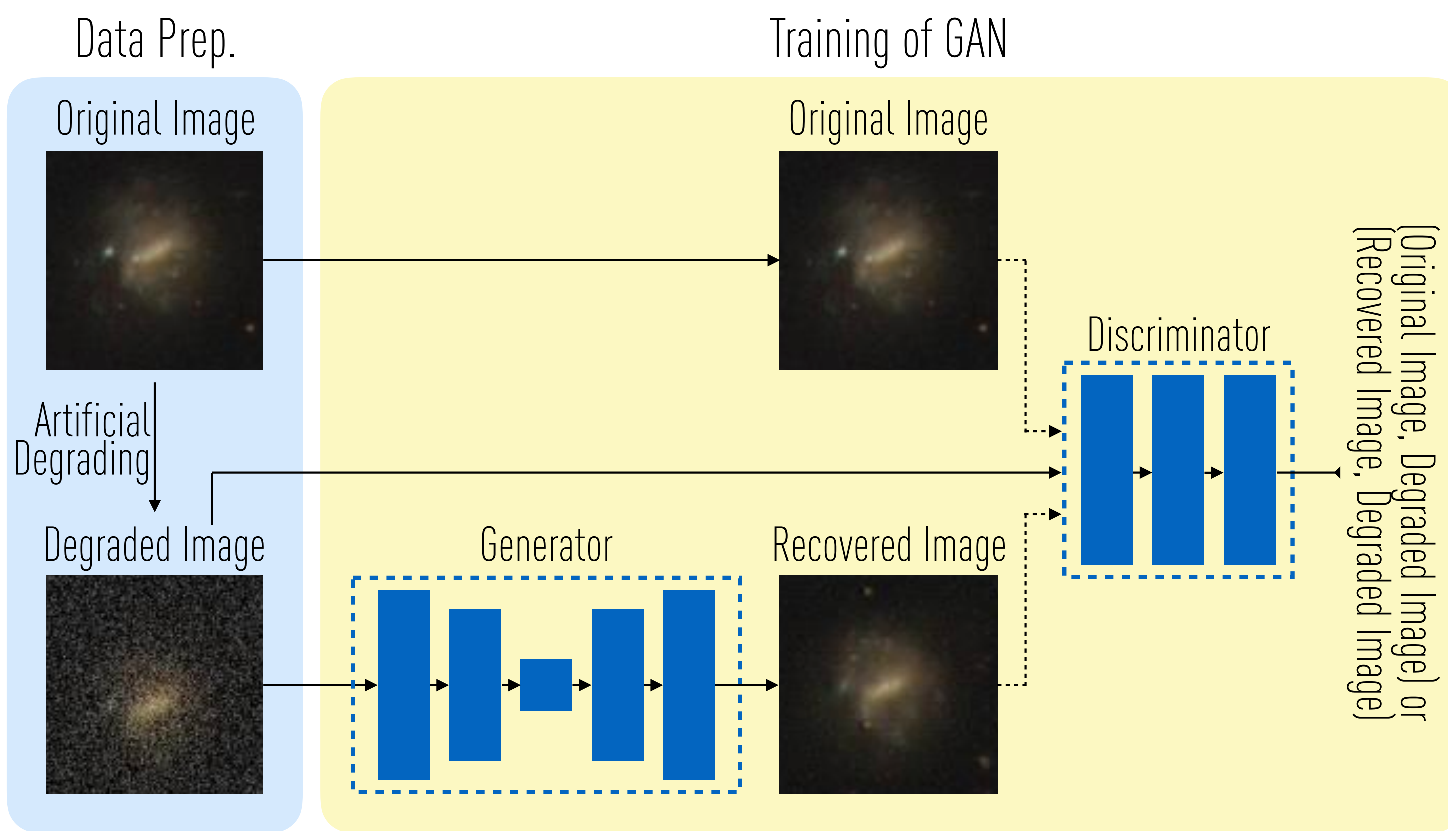
GENERATIVE ADVERSARIAL NETWORKS

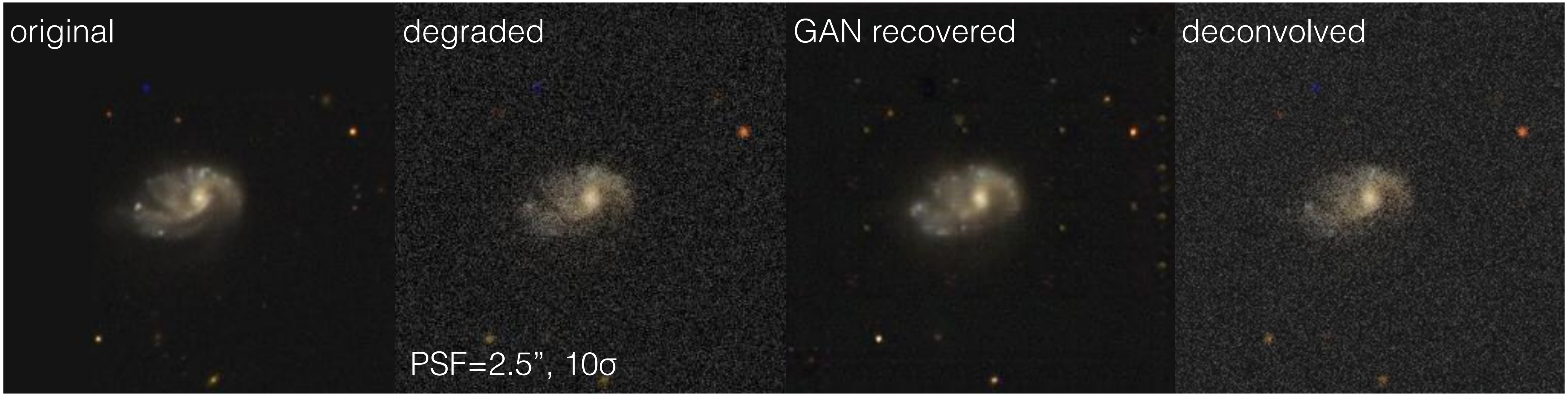


courtesy of Y. Kawabata


Artificial Sunspot
Real Sunspot
Artificial Granule
Real Granule
pix
0
20
40
60
80
courtesy of Y. Kawabata

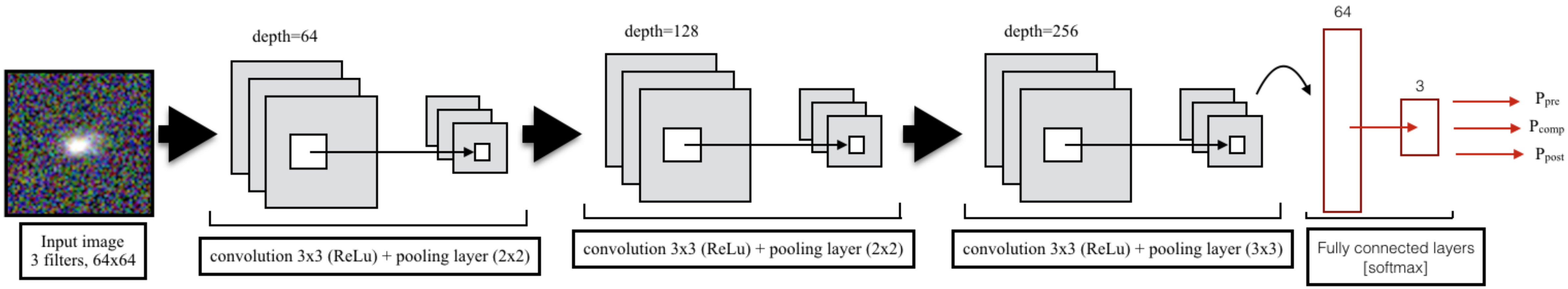
DECONVOLUTION OF GALACTIC IMAGES: GAN

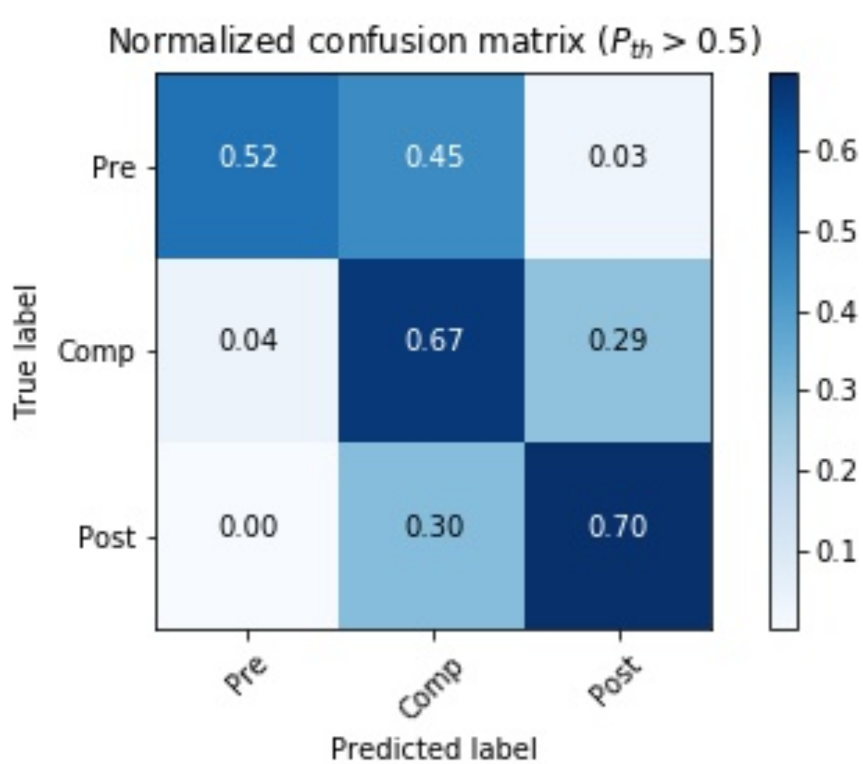




Schawinski et al (2017)

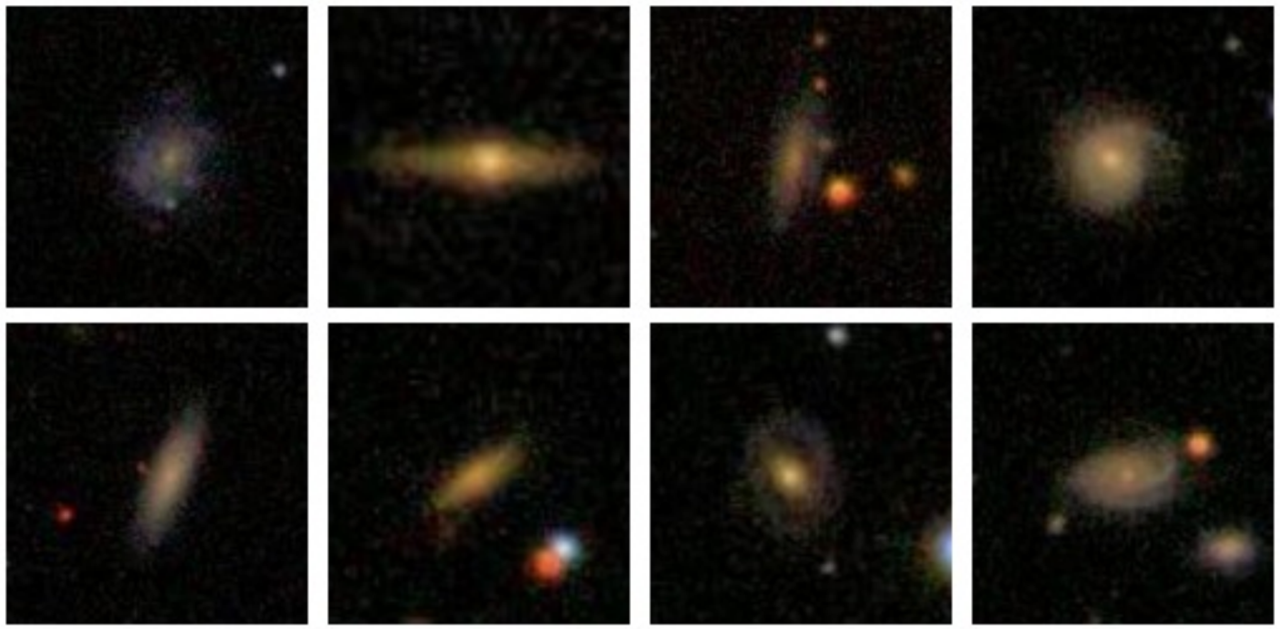
CLASSIFYING GALAXIES AT HIGH REDSHIFT

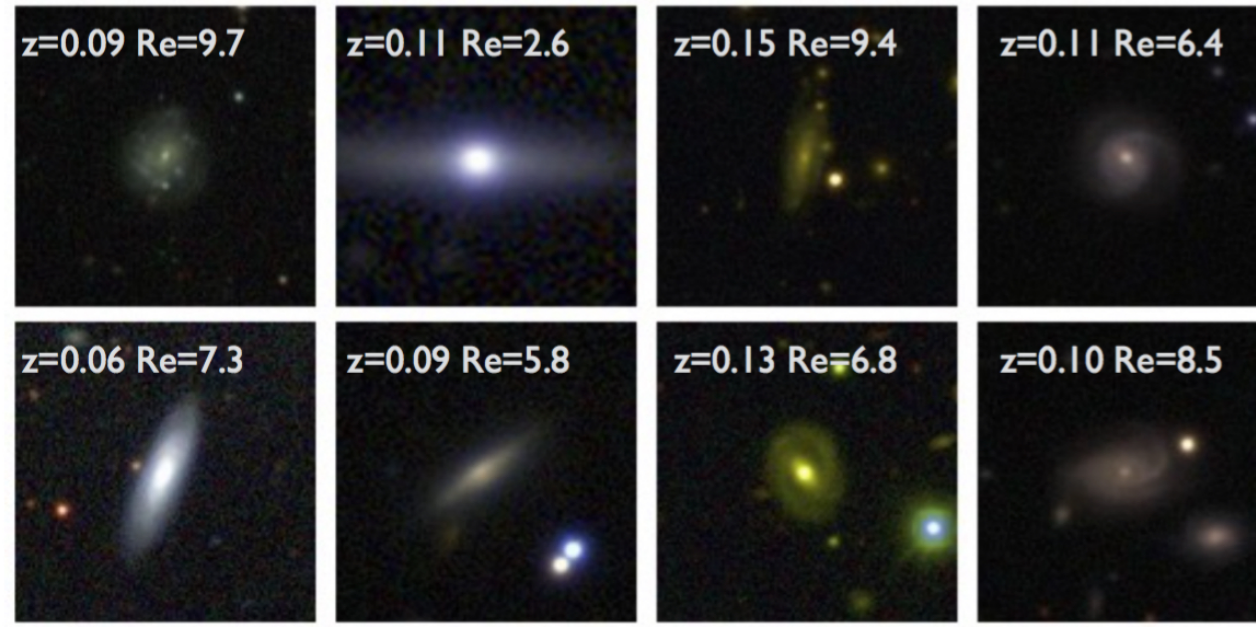


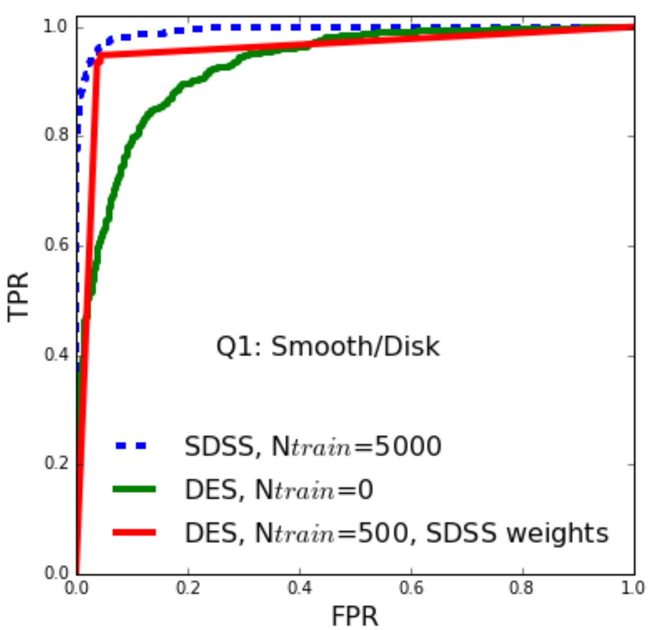


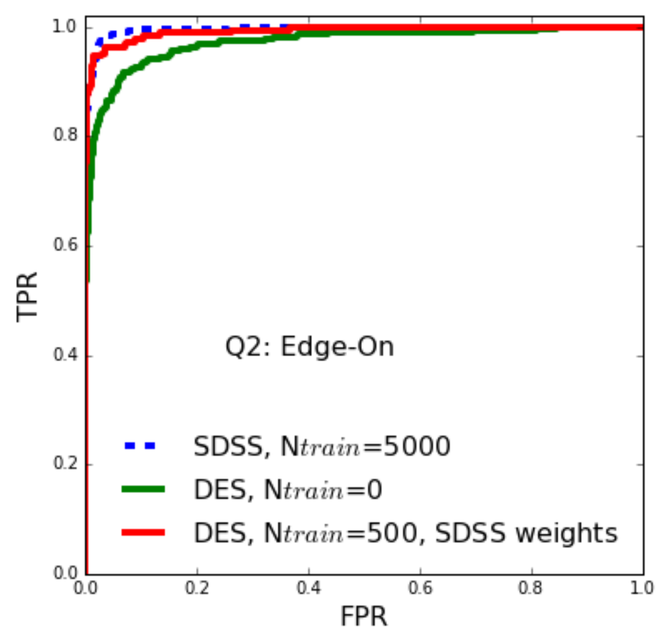
Huertas-Company et al (2018)

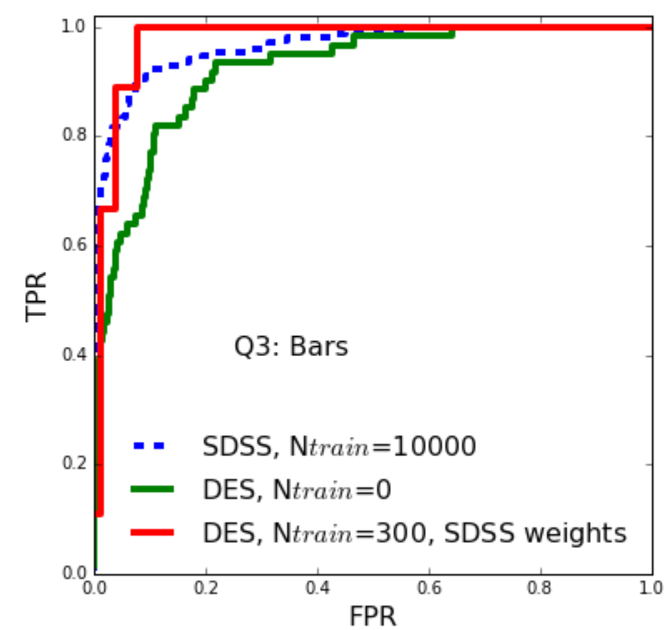
TRANSFER LEARNING FOR FUTURE SURVEYS











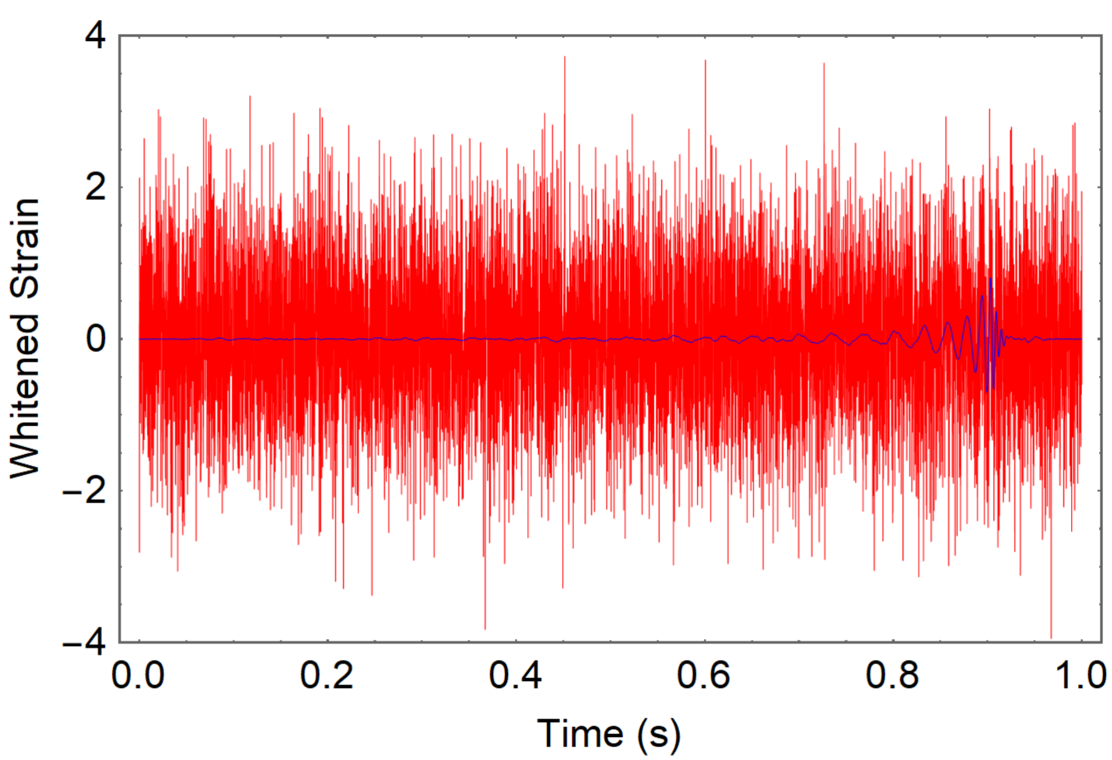
Domínguez-Sánchez et al (2018)

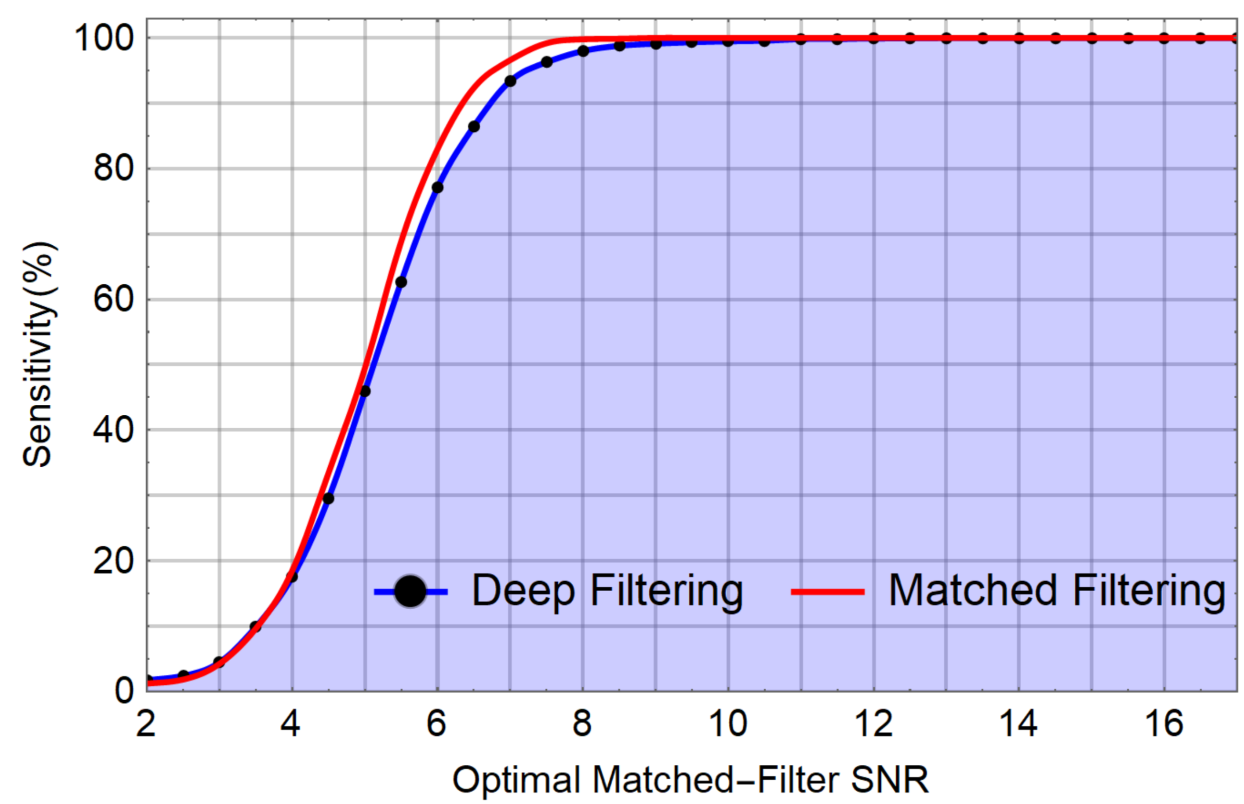
DETECTION OF GRAVITATIONAL WAVES

Deep Learning for Real-time Gravitational Wave Detection and Parameter Estimation: Results with Advanced LIGO Data

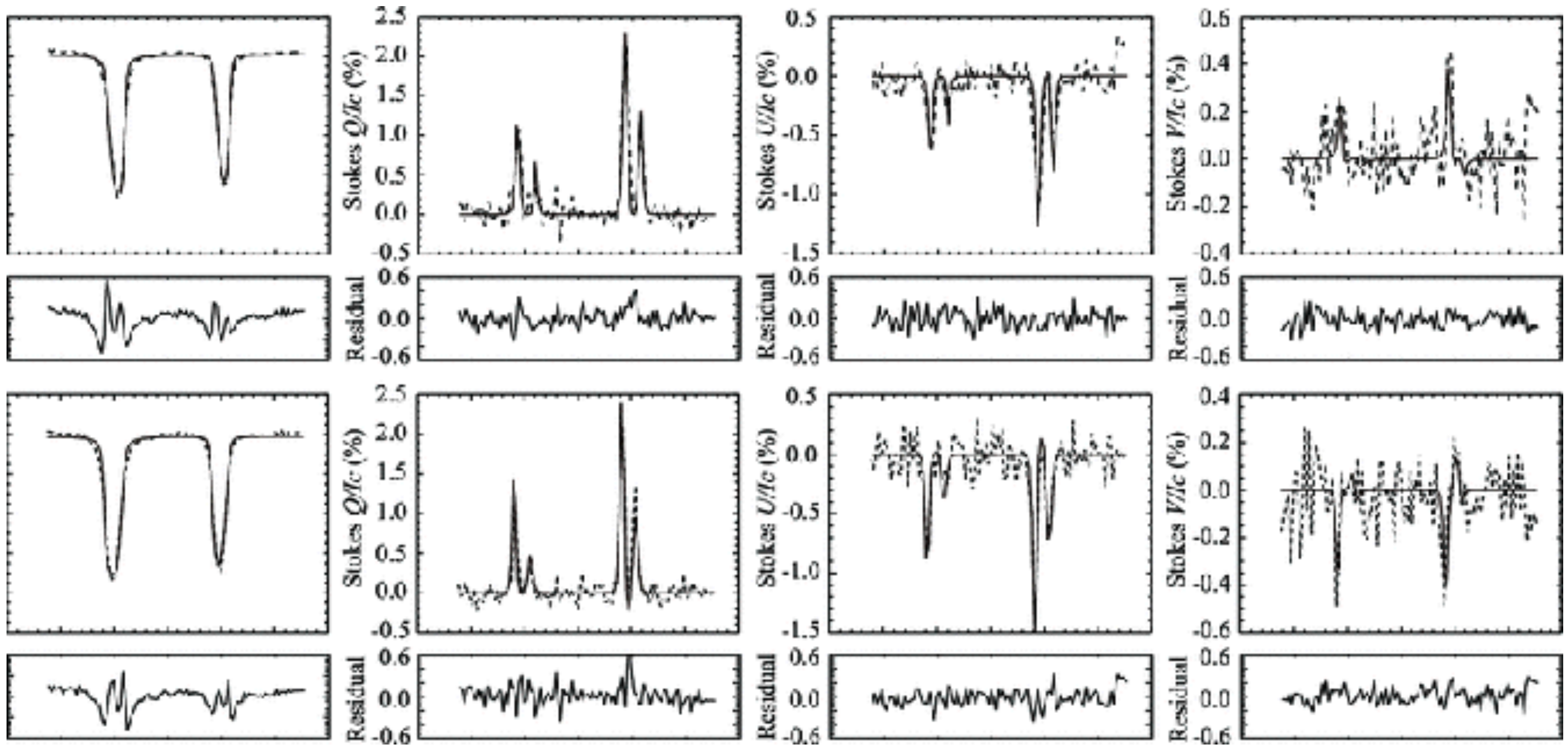
Daniel George[1,2] and E. A. Huerta[2]

[1]*Department of Astronomy, University of Illinois at Urbana-Champaign, Urbana, Illinois, 61801*
[2]*NCSA, University of Illinois at Urbana-Champaign, Urbana, Illinois, 61801*



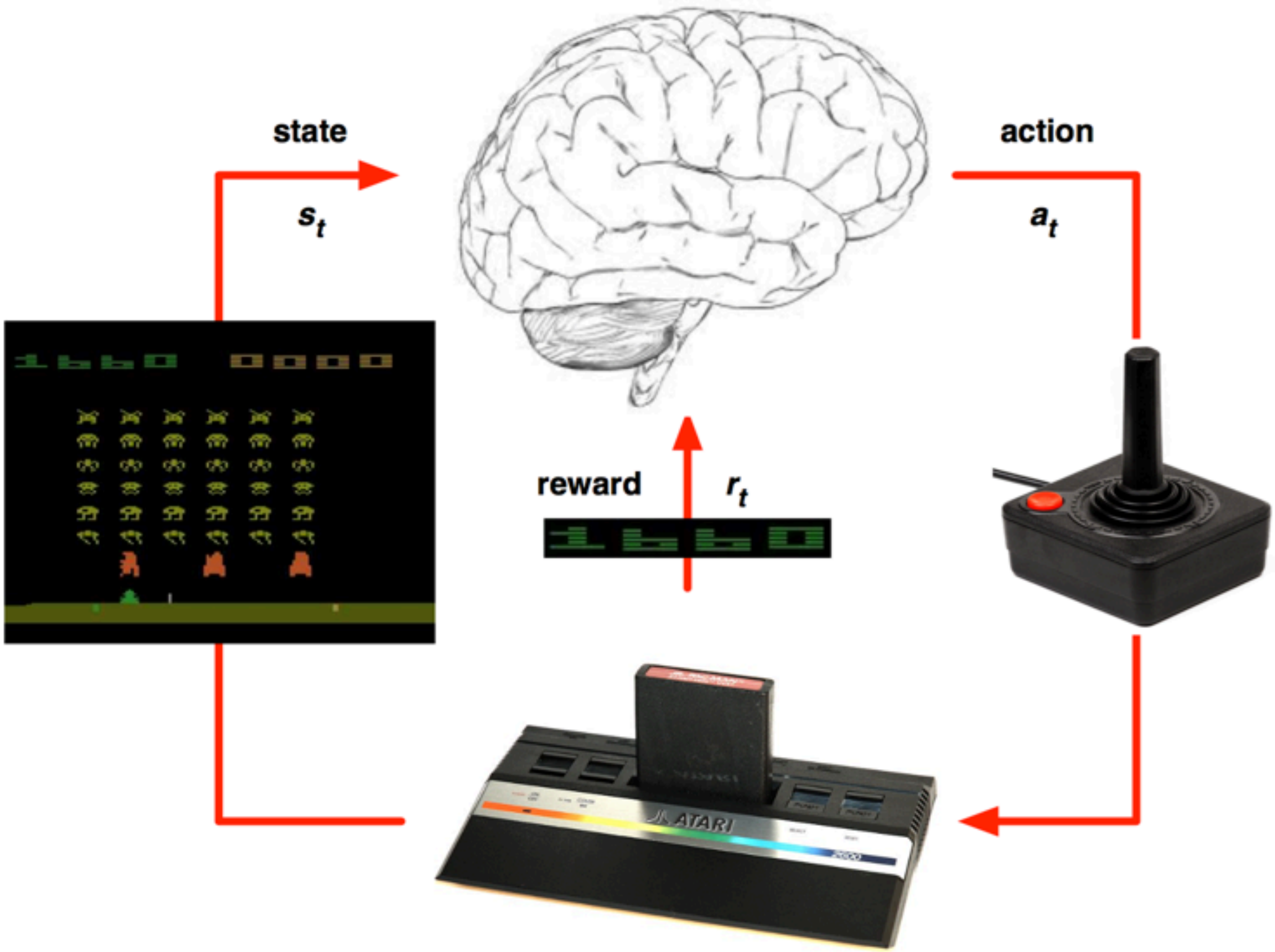


INVERSIONS WITHOUT RESPONSE FUNCTIONS



Reinforcement learning

REINFORCEMENT LEARNING



PACKAGES FOR DEEP LEARNING

TensorFlow

Keras

PyTorch

NOTEBOOK

https://bit.ly/2Kh35Kv