μBooNE

Deep Neural Network Applications for LArTPC Data Reconstruction

FNAL Neutrino Seminar Series
April 2018

Kazuhiro Terao

SLAC National Accelerator Laboratory

Outline

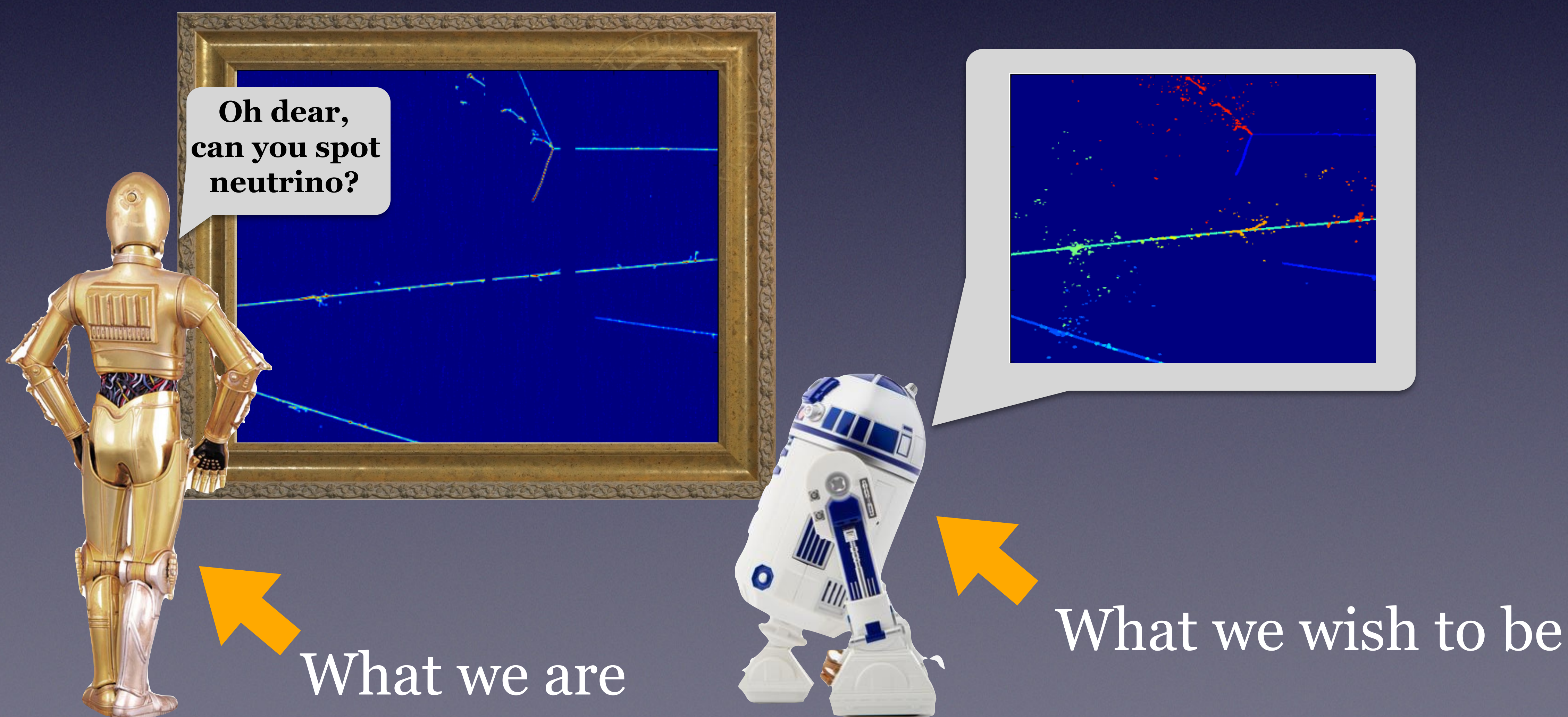
- Liquid Argon Time Projection Chambers
- Recent innovations in Computer Vision
- Deep Neural Networks for data reconstruction
- Wrap-up

Some people like buzzwords, some people don't.
My topic is about applying buzzword on buzzword.
Some of you might hate it, but hopefully some of you love it.

Disclaimer

In this talk ...

- Mainly story-telling of what's going on
- No physics results
- No comparison with traditional reco
- Very few distributions, mainly figures



Liquid Argon Time Projection Chambers

Outline

- **Liquid Argon Time Projection Chambers**
- Recent innovations in Computer Vision
- Deep Neural Networks for data reconstruction
- Wrap-up

75 cm

Run 3493 Event 41075, October 23rd

Neutrino Oscillation Measurements

What detector technology?

Three important detector features for oscillation measurement

$$P(\nu_\mu \rightarrow \nu_e) = \sin^2 2\theta \sin^2\left(\frac{1.27\,\Delta m^2\, L}{E_\nu}\right)$$

Good Energy Resolution

Precise E_ν reduce oscillation uncertainty

Large Mass (scalable)

"More" statistics to measure rare physics process

Particle ID Capability

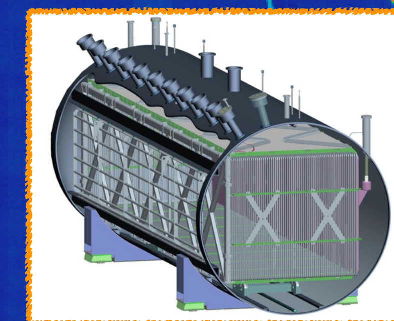
Better ν identification background rejection

LArTPC: Particle Imaging Detector

μBooNE

ν_μ

~mm/pixel spatial resolution
~MeV level sensitivity



MicroBooNE
~87 ton (school bus size)

Bubble Chamber

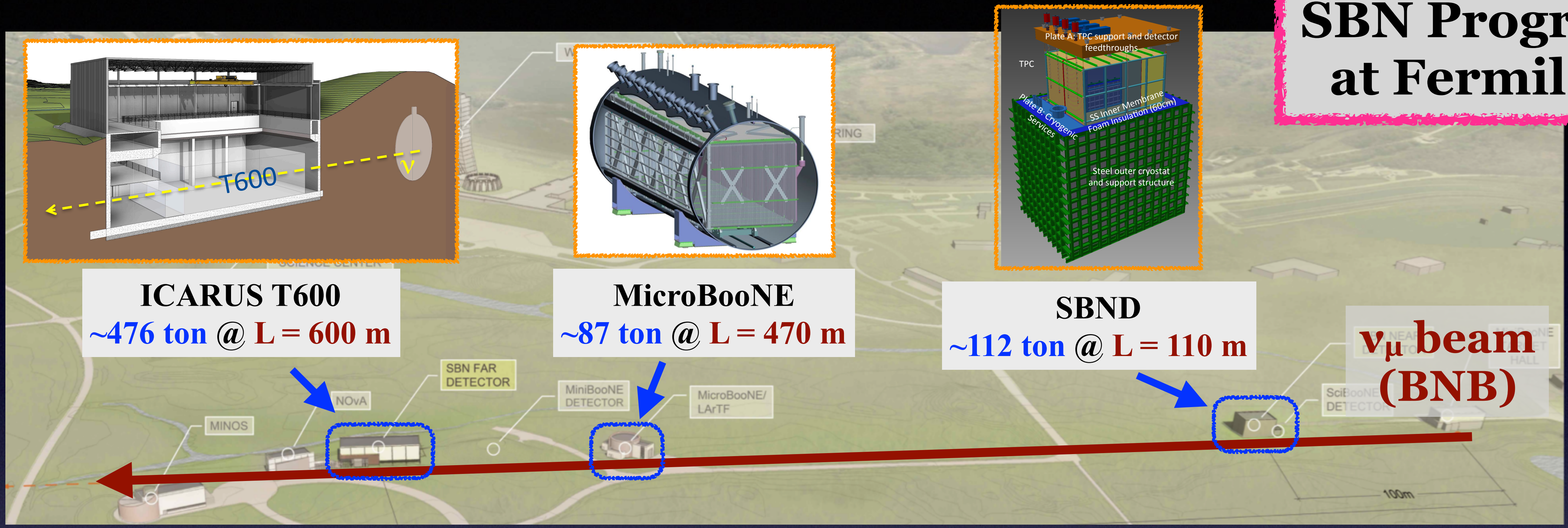
2015

Liquid Argon Time Projection Chamber

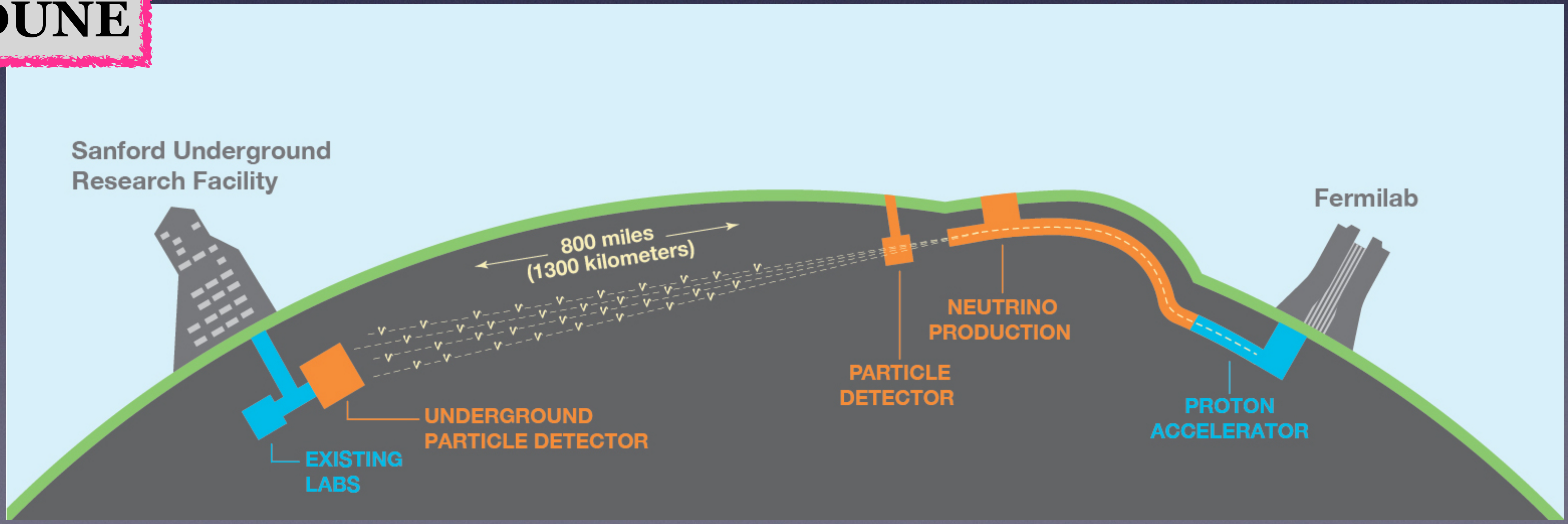
- Chamber-like images: digitized electronics readout
- Calorimetric measurement + scalability to a large mass

LArTPCs for Neutrino Oscillation Experiments

SBN Program at Fermilab

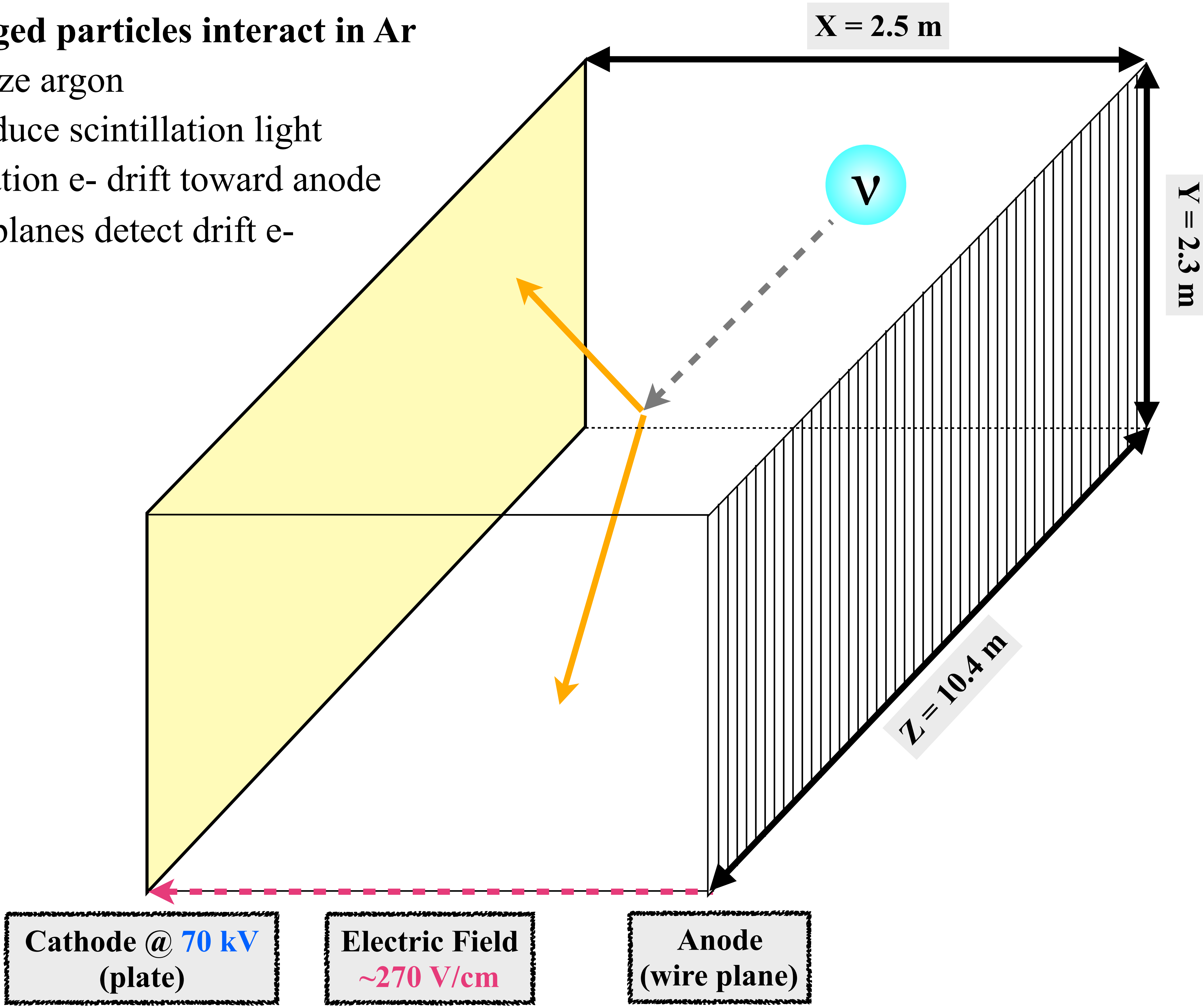


DUNE



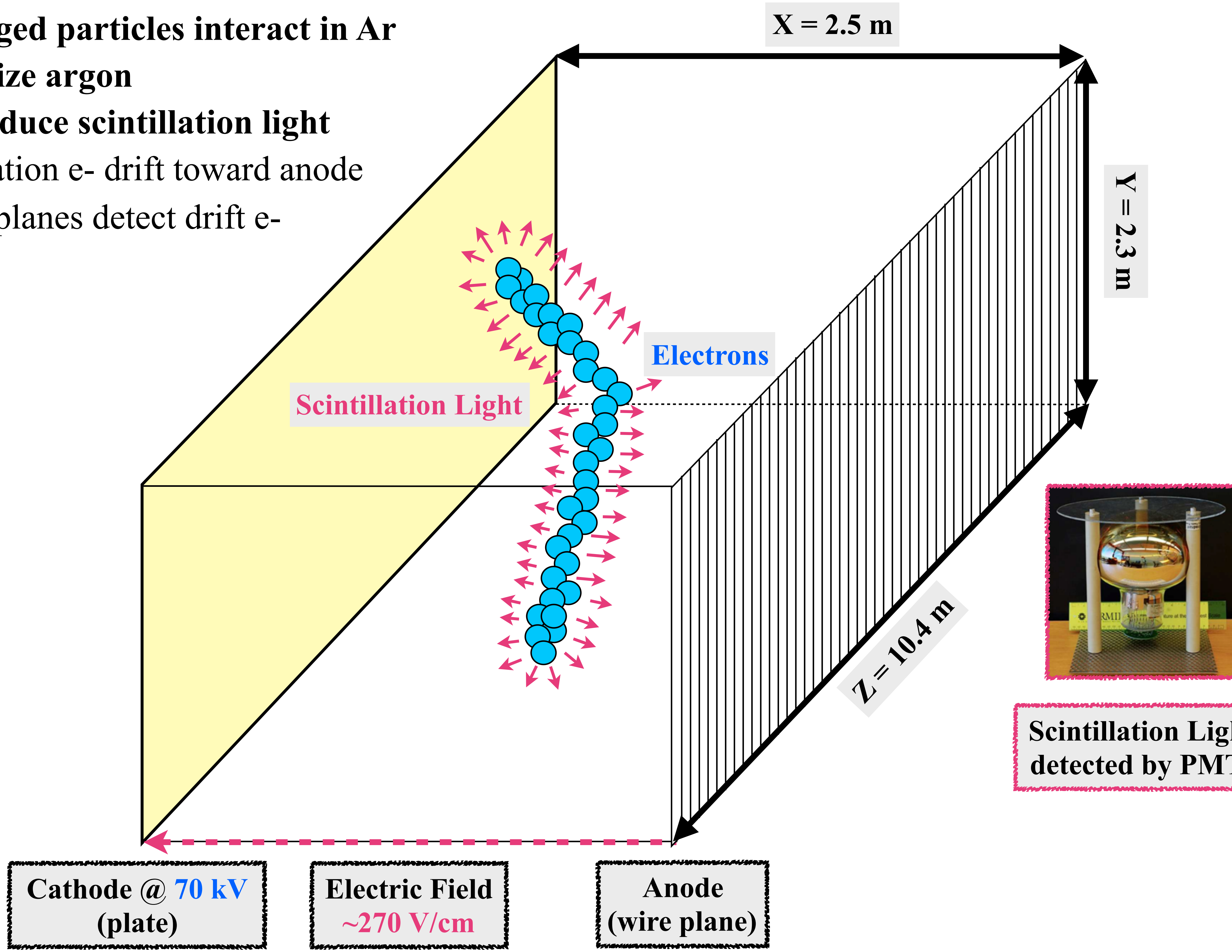
How MicroBooNE LArTPC Work (I)

1. **Charged particles interact in Ar**
    - Ionize argon
    - Produce scintillation light
2. Ionization e- drift toward anode
3. Wire planes detect drift e-



How MicroBooNE LArTPC Work (II)

1. **Charged particles interact in Ar**
    - **Ionize argon**
    - **Produce scintillation light**
2. Ionization e- drift toward anode
3. Wire planes detect drift e-

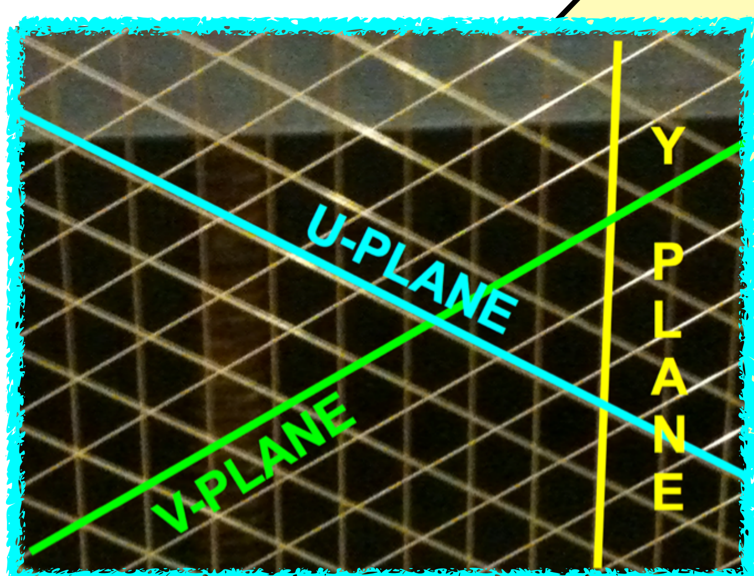


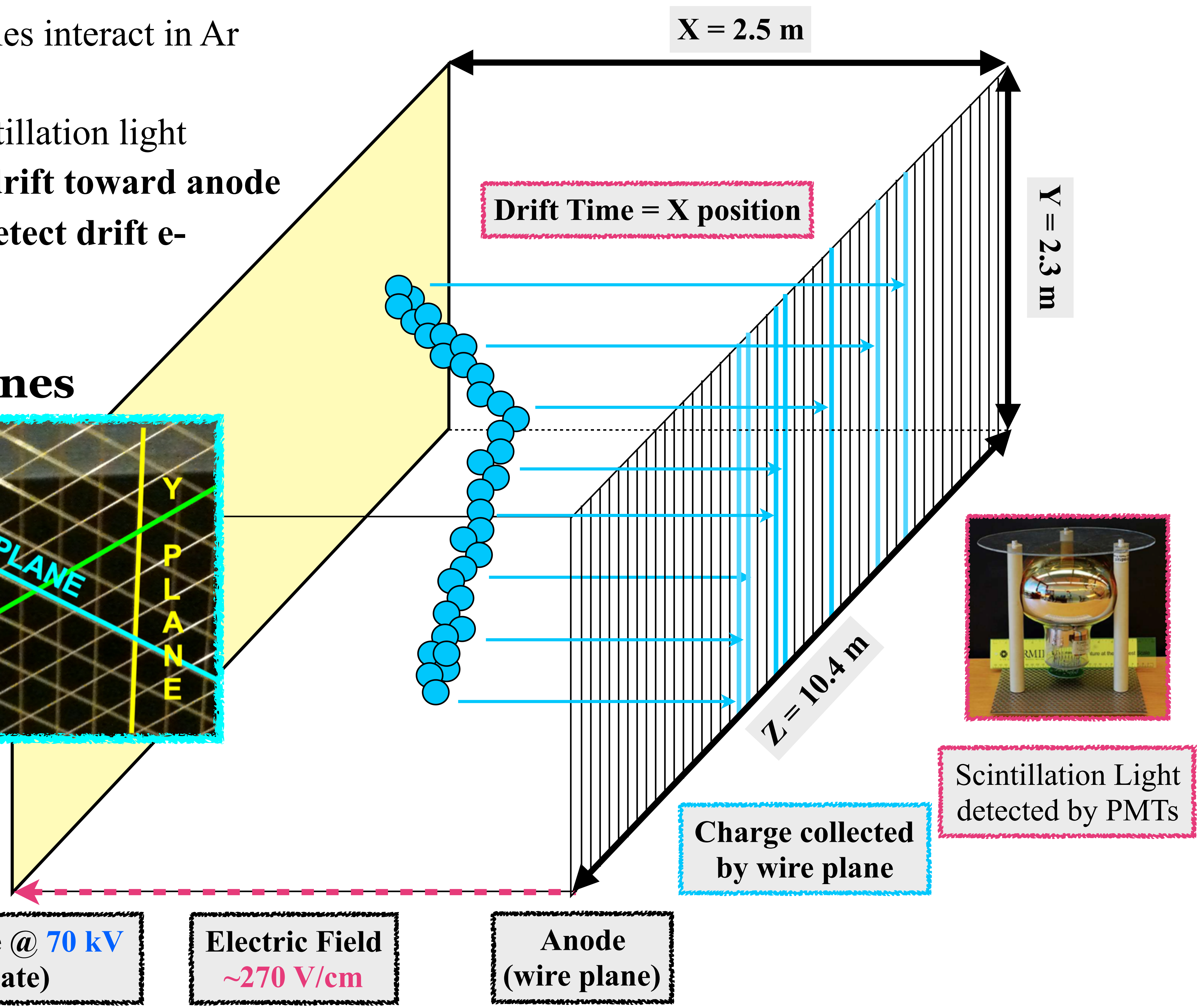
Scintillation Light detected by PMTs

How MicroBooNE LArTPC Work (III)

1. Charged particles interact in Ar
    - Ionize argon
    - Produce scintillation light
2. **Ionization e- drift toward anode**
3. **Wire planes detect drift e-**

Three Wire Planes



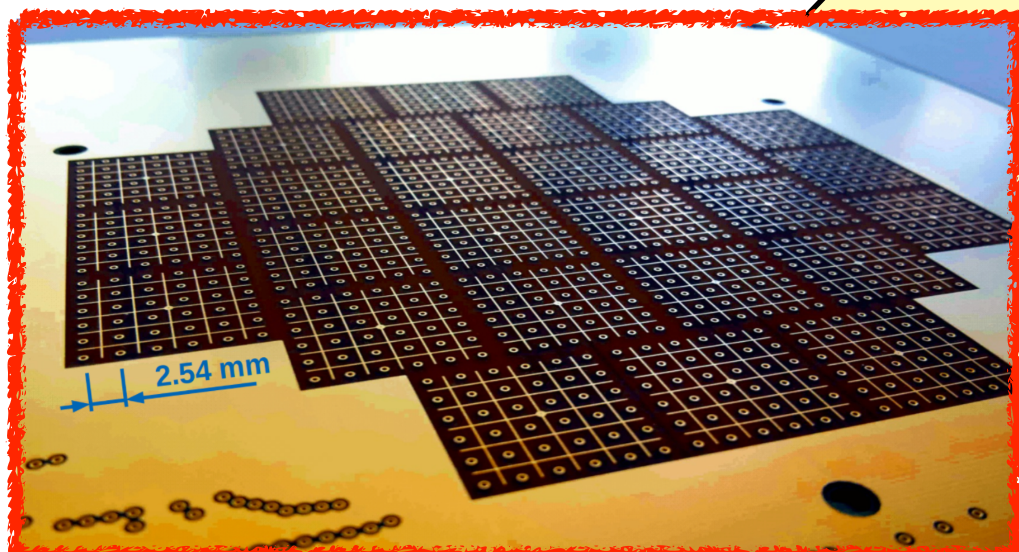


How ~~MicroBooNE~~ LArTPC Work (IV)
Pixel (DUNE ND)

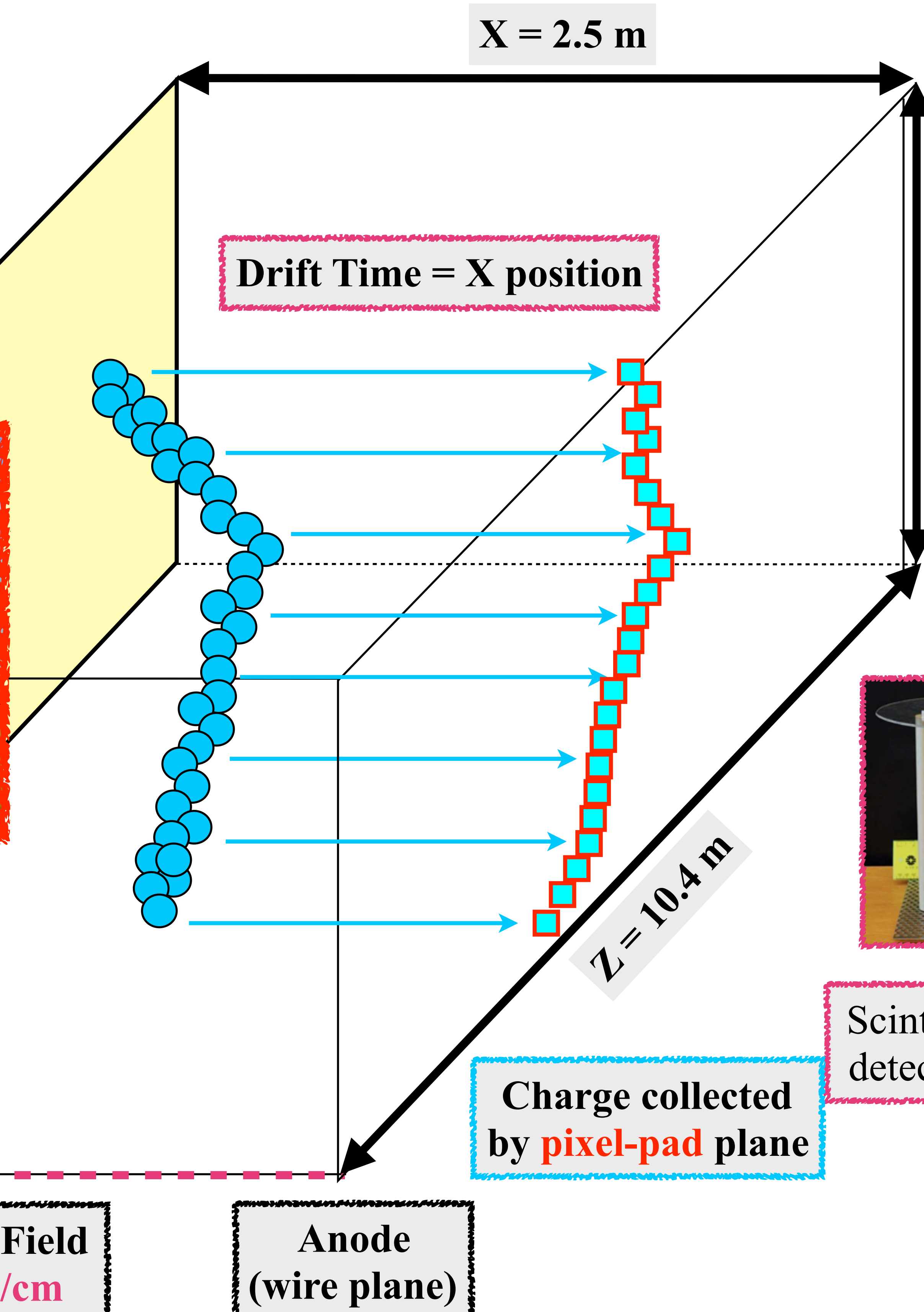
1. Charged particles interact in Ar
    - Ionize argon
    - Produce scintillation light

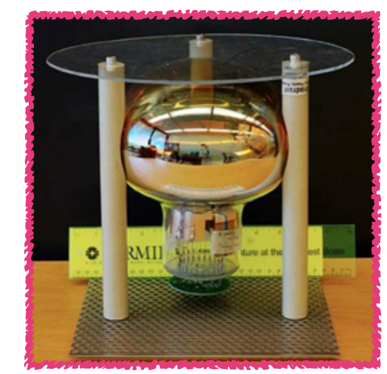
2. Ionization e- drift toward anode

3. ~~Wire planes~~ detect drift e-
pixel detector



[J. Assadi et al. arxiv 1801.08884]

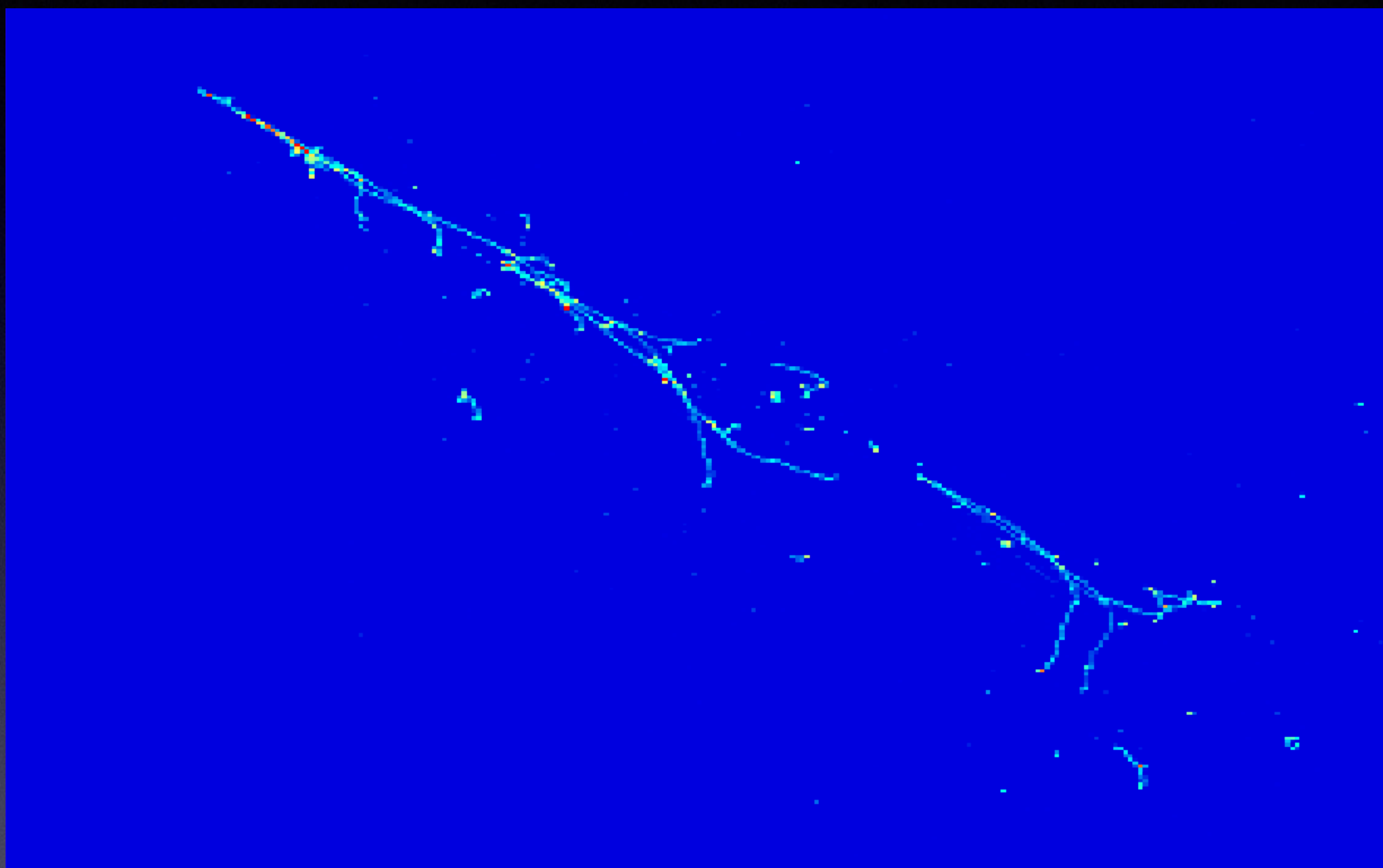




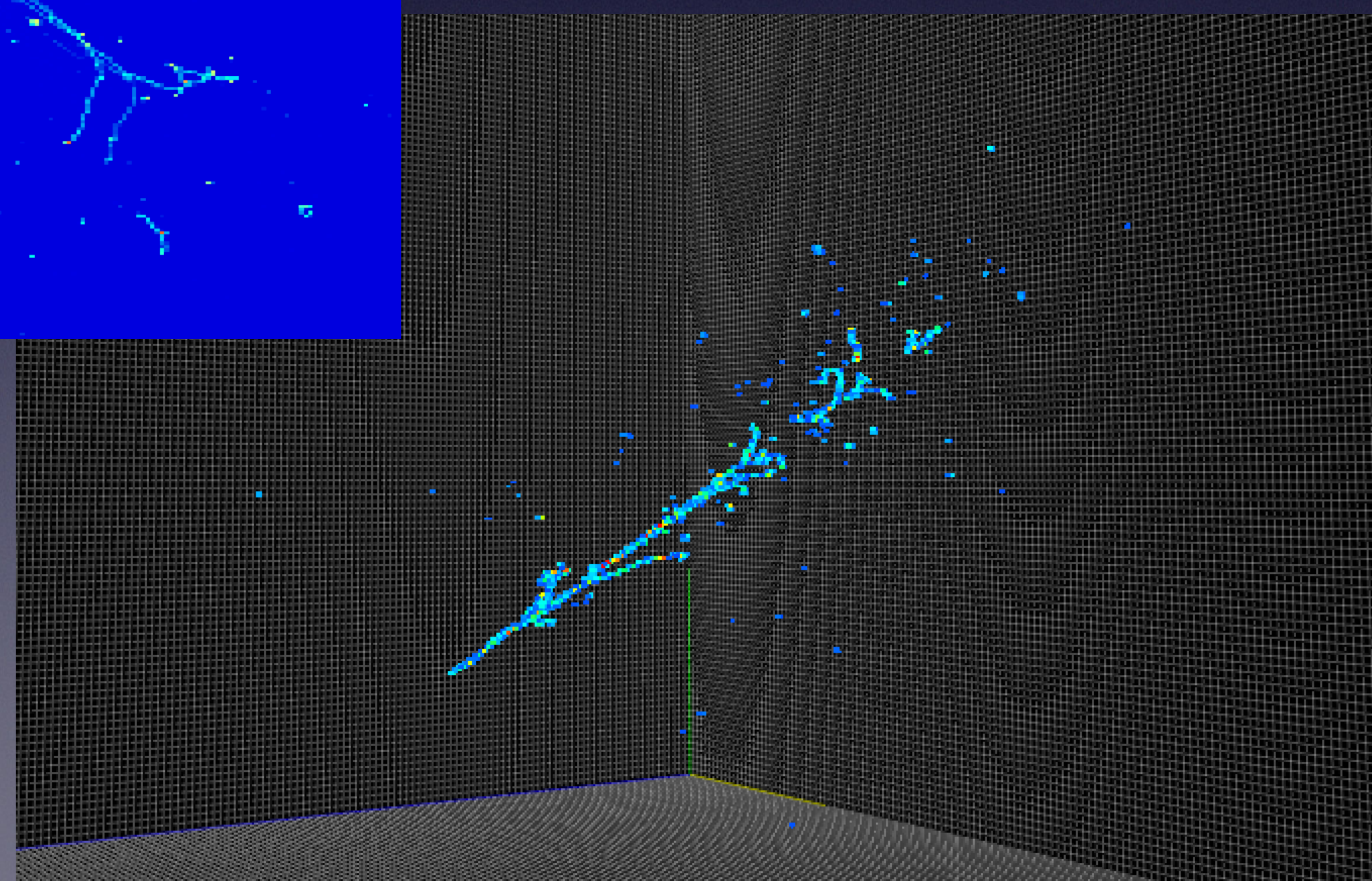
Scintillation Light detected by PMTs

LArTPC: Particle Imaging Detector

... putting everything together ...



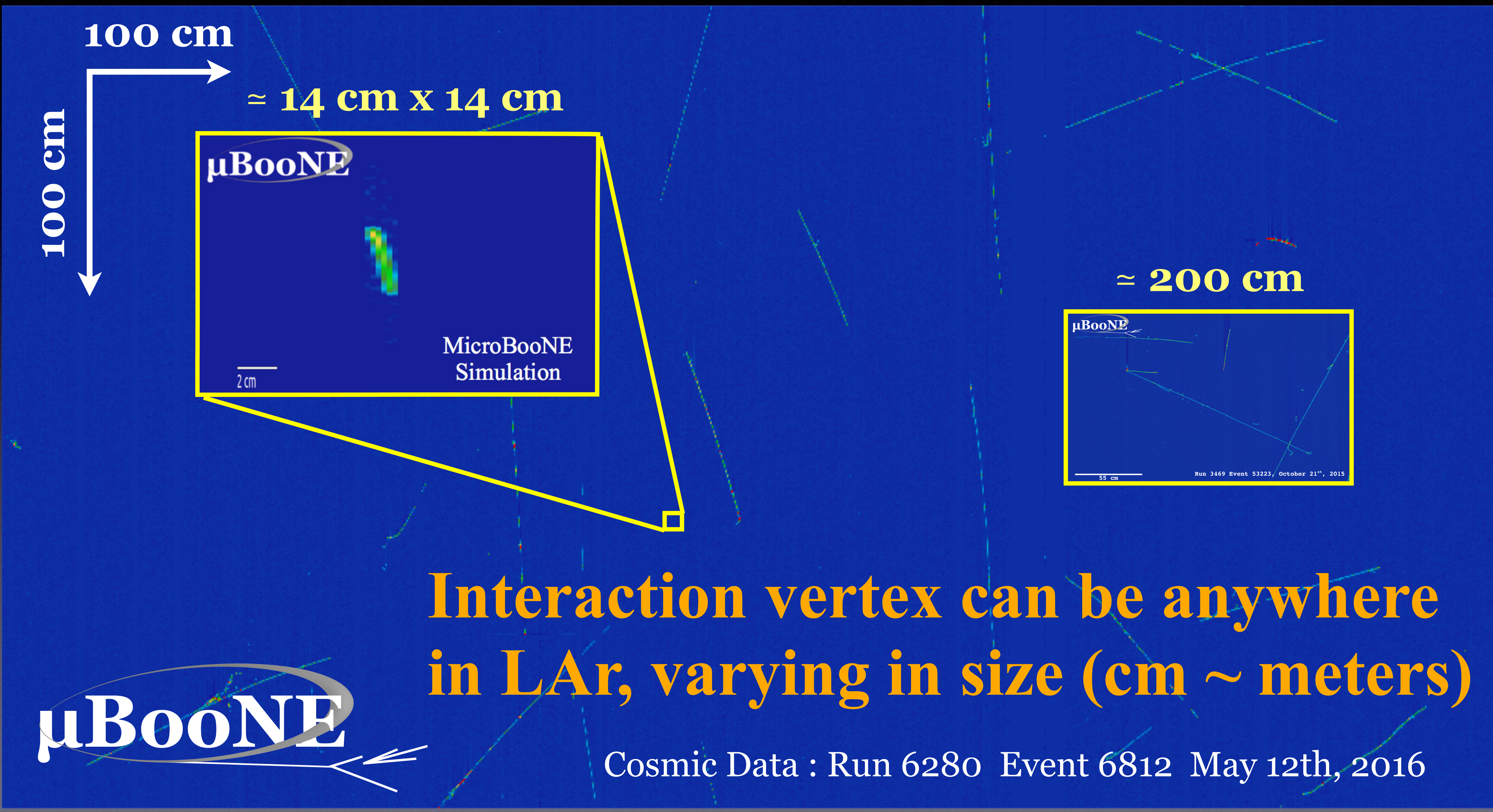
3D Imaging
(Pixel Detector)



2D Projection
(Wire Detector)

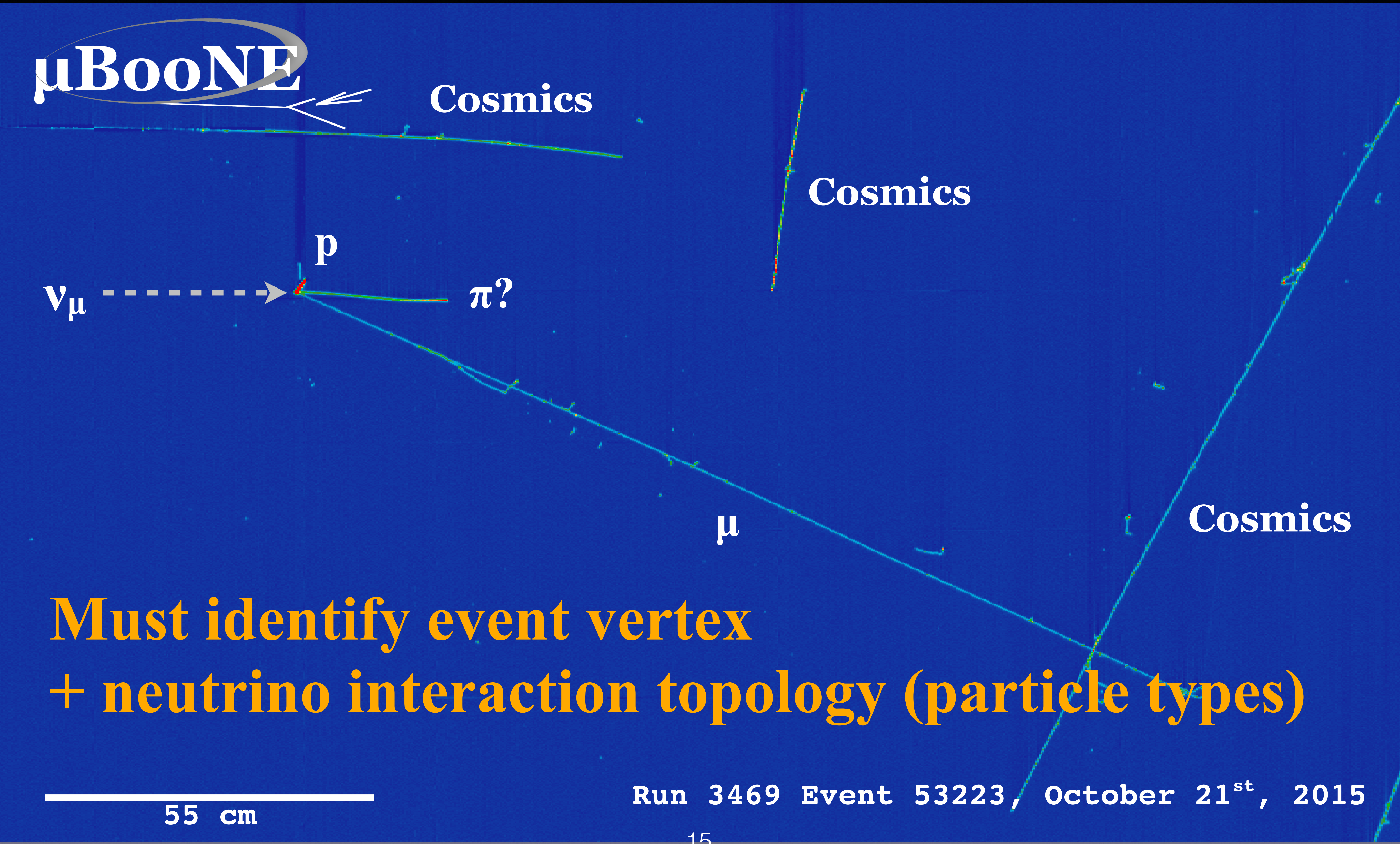
Challenges in LArTPC Data Analysis?

Challenges in LArTPC Data Analysis?



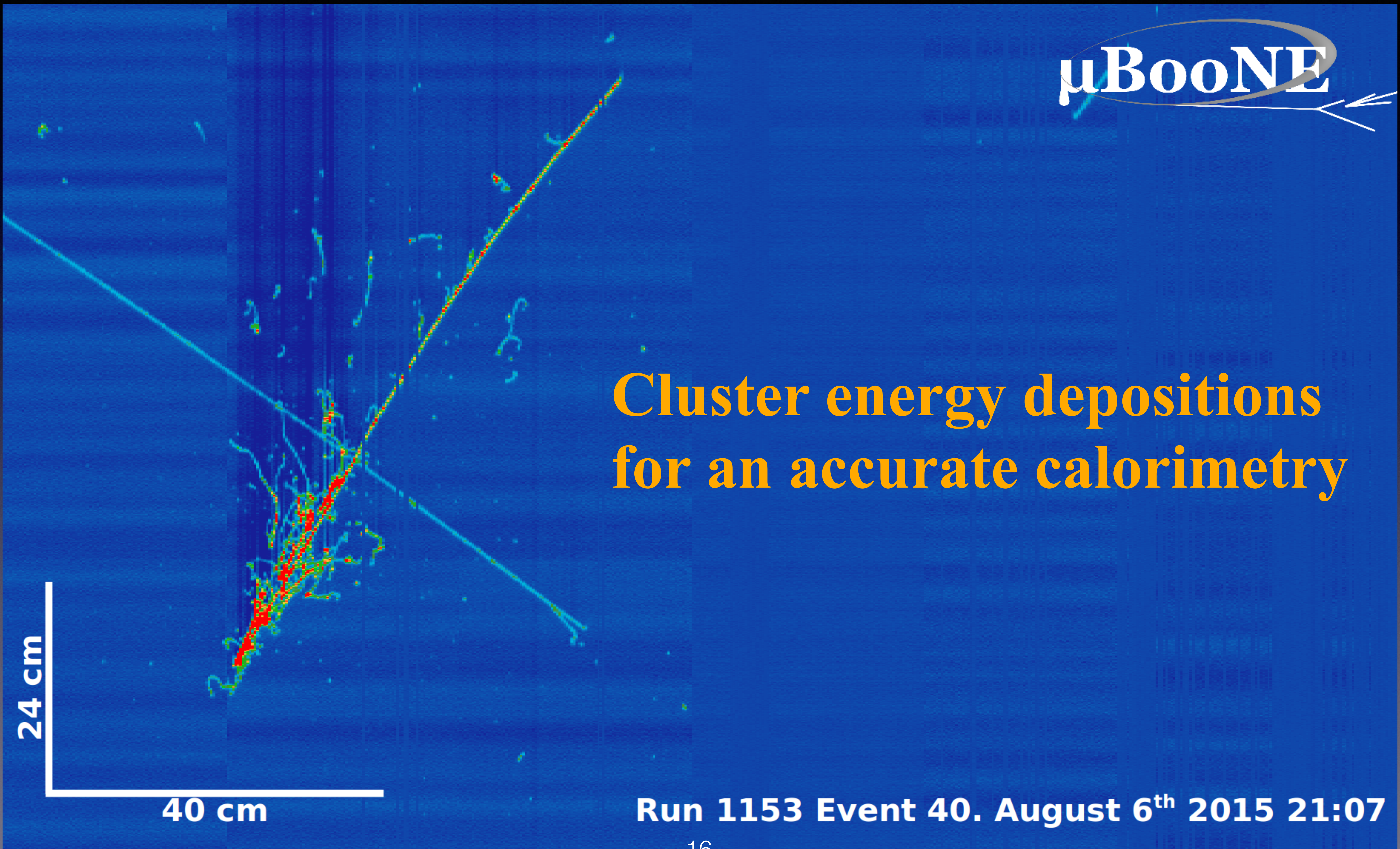
Interaction vertex can be anywhere in LAr, varying in size (cm ~ meters)

Challenges in LArTPC Data Analysis?

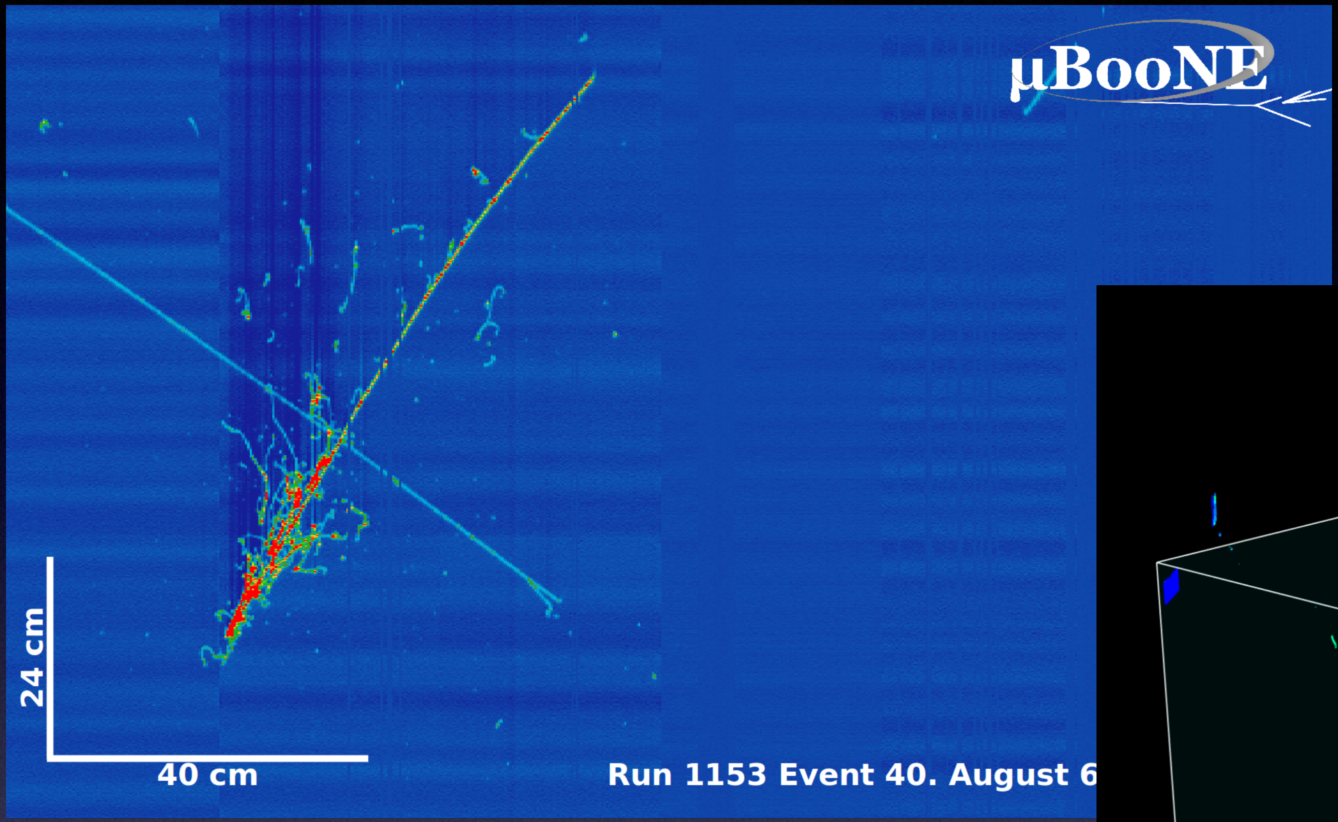


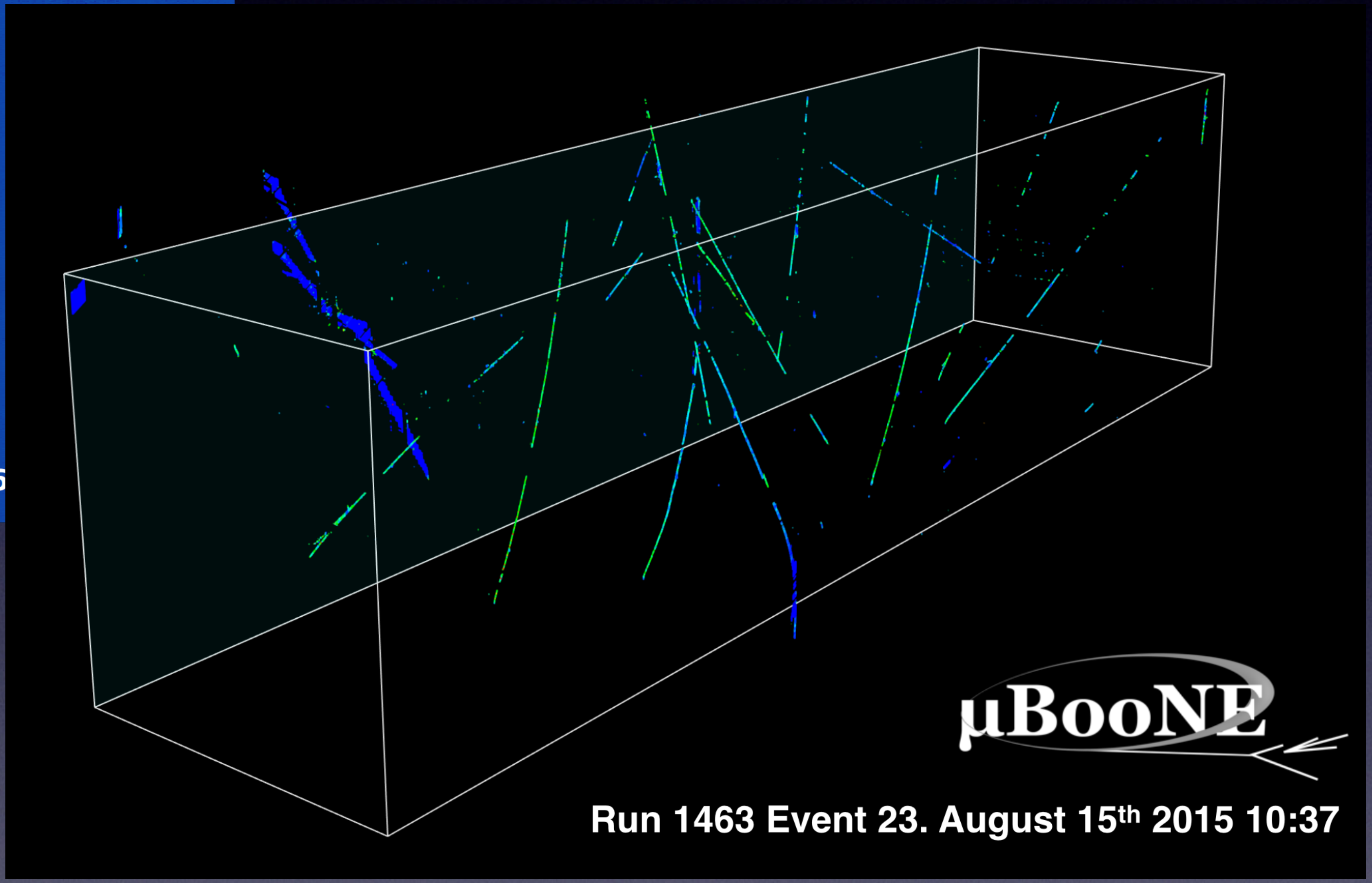
Must identify event vertex
+ neutrino interaction topology (particle types)

Challenges in LArTPC Data Analysis?



Challenges in LArTPC Data Analysis?

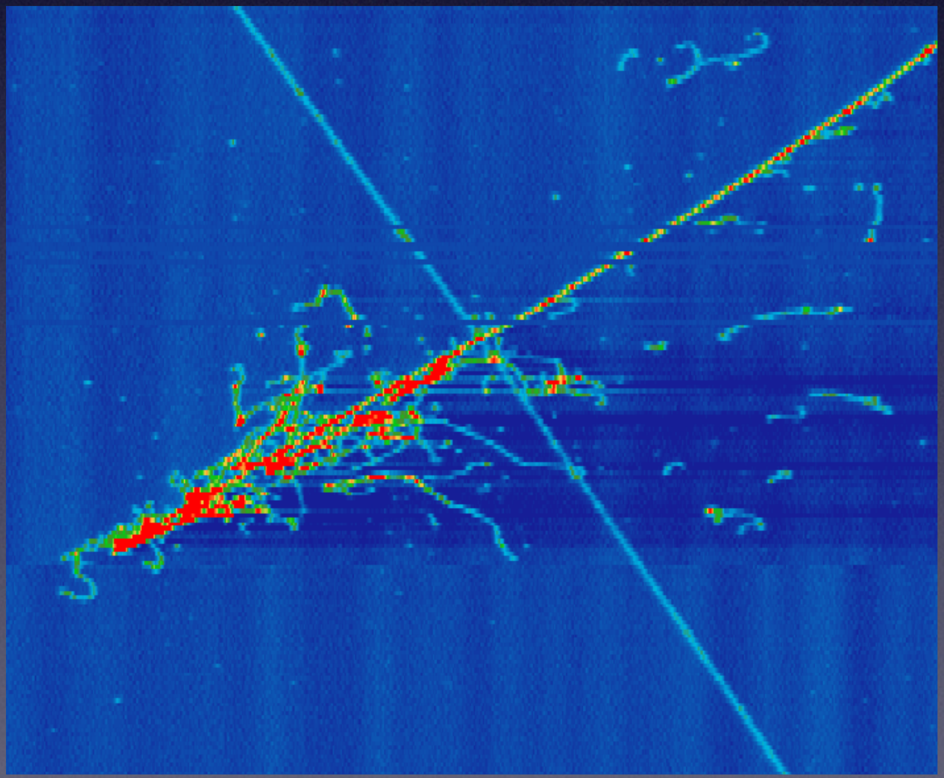




Deal with optical illusions in 2D projections + 3D pattern recognitions

Challenges in LArTPC Data Analysis?

Programming pattern recognition algorithms is non-trivial, need full-chain optimization



Our data is an "image", a matrix of numbers

we wish

Not how it looks in our code

in reality

```
0110101010010101101010100101101 0
10111010101010010101000100101011 01
0101001011010101001010110101010
01011010101001010110101010101101
0101001010110101010010110101010
01011010101001010110101010010110
10101001010110101010101101010100
1010110101010011010110101010100101
```

How it *actually* looks in our code

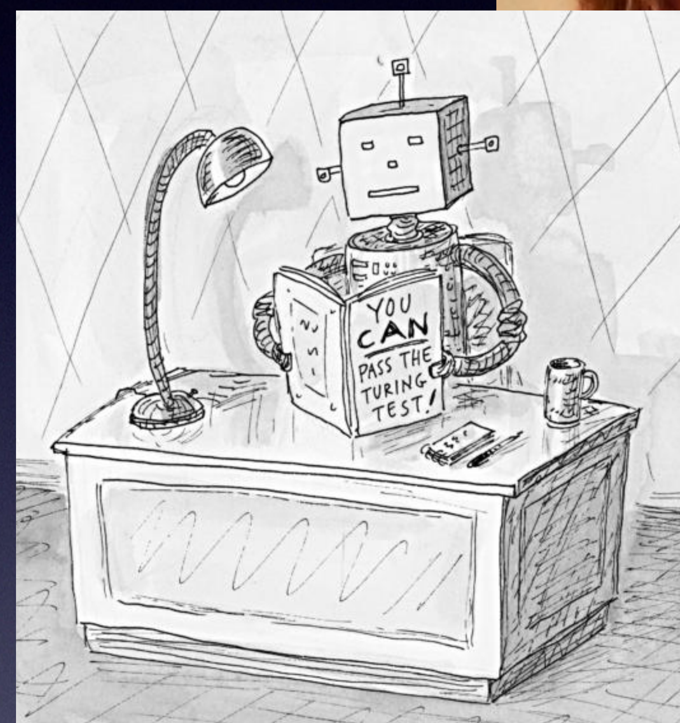
Challenges in LArTPC Data Analysis?

Solution?

- The core: **pattern recognition challenge**
- Most of analysis/reconstruction **“trivial by eyes”**
  - motivation to try neuromorphic algorithms
- Solve the “**full chain optimization**” issue by design
  - machine learning algorithms

Recent advancement in computer vision

- Recent breakthrough in complex pattern recognition
  - Machine learning solution: “Deep Neural Networks”

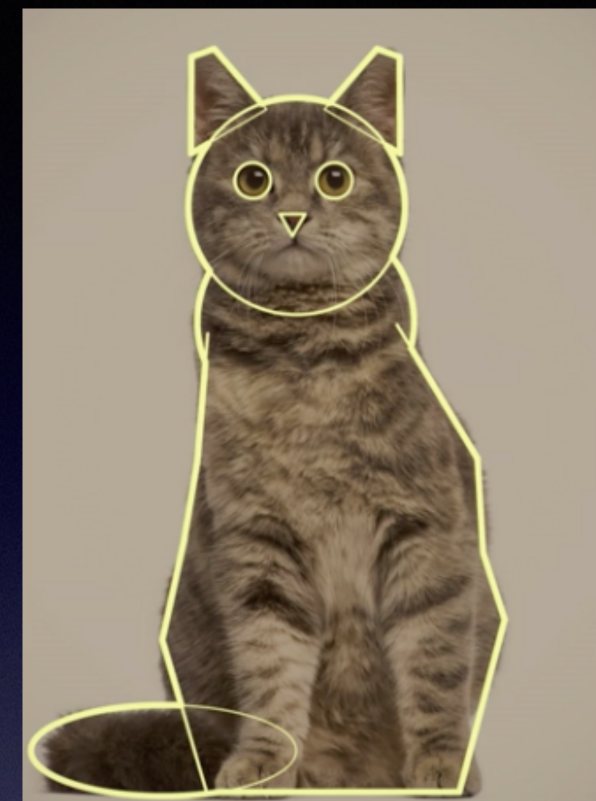


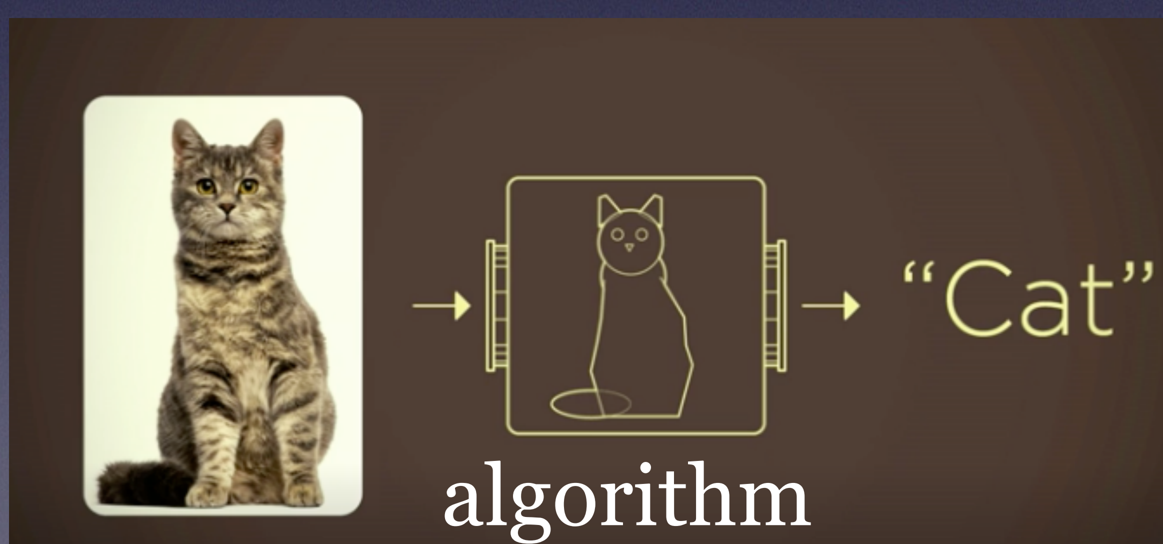
“Fake” celebrity images generated by DNN in 1024 x 1024 resolution

Outline

- Liquid Argon Time Projection Chambers
- **Recent innovations in Computer Vision**
- Deep Neural Networks for data reconstruction
- Wrap-up

Classic Problem: Image Categorization





A cat
= collection of
certain shapes

Taken from slides by Fei-Fei's TED talk

Classic Problem: Image Categorization

… how about these?

Partial cat
(escaping fiducial volume)

Stretching cat
(DIS?)

Outliers
(axions/dark matter)

Taken from slides by Fei-Fei’s TED talk

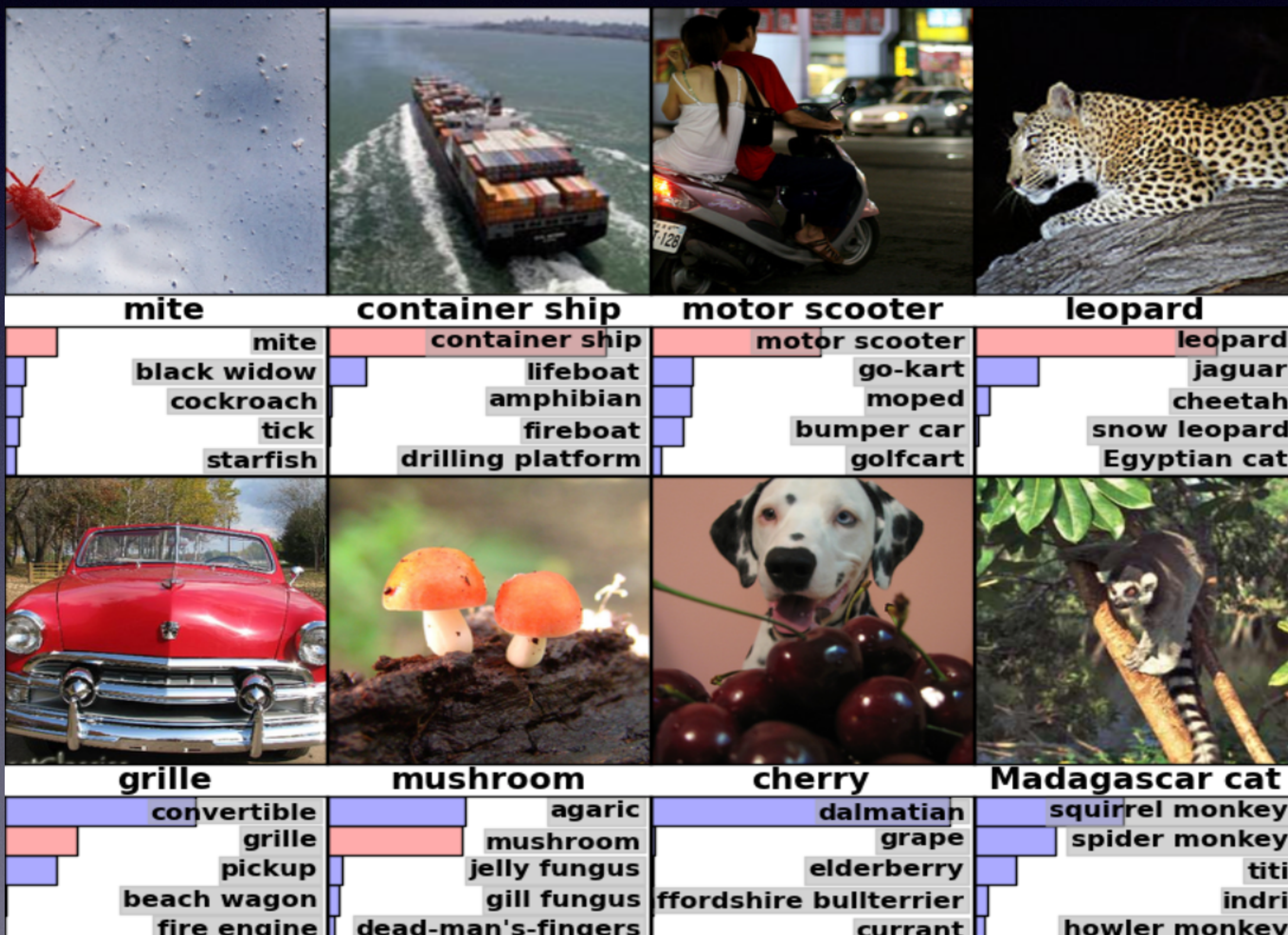
The Year of Breakthrough: 2012

The field of computer vision celebrated

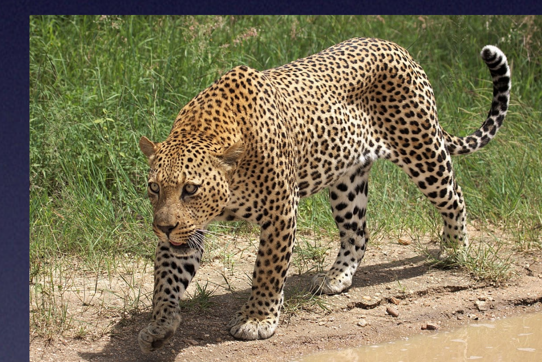
Birth of "Deep Learning"

AlexNet: 8-layers deep neural network

> 20,000 citations!



For my reference

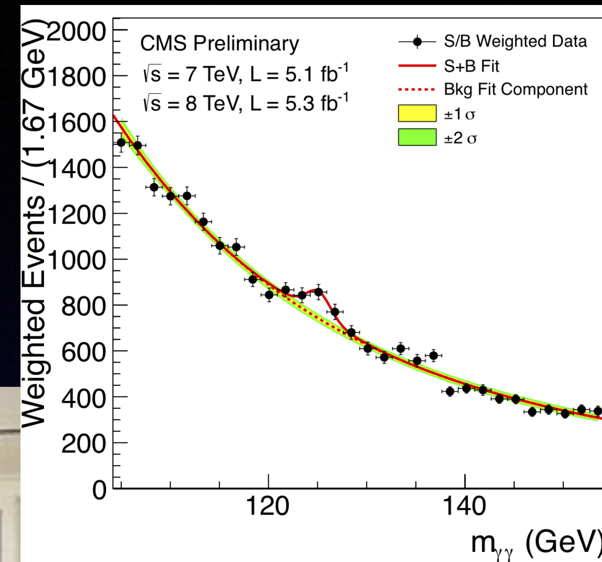


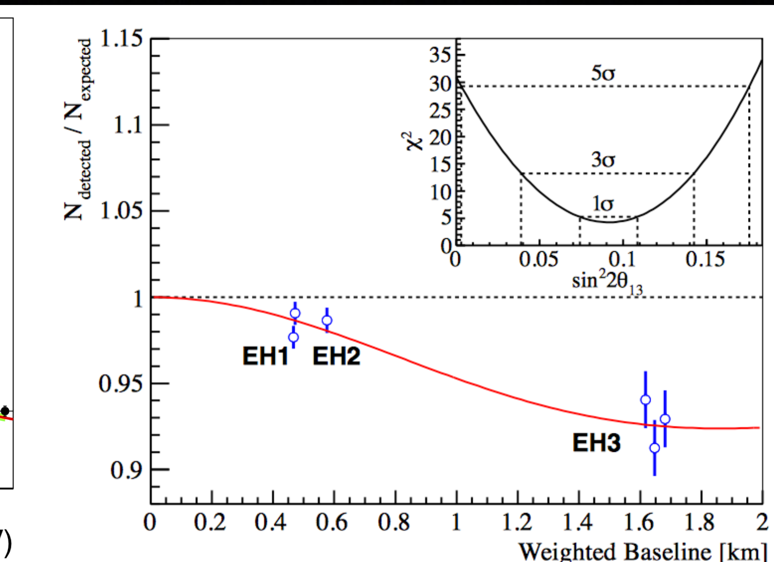
Leopard

Jaguar

The Year of Breakthrough: 2012

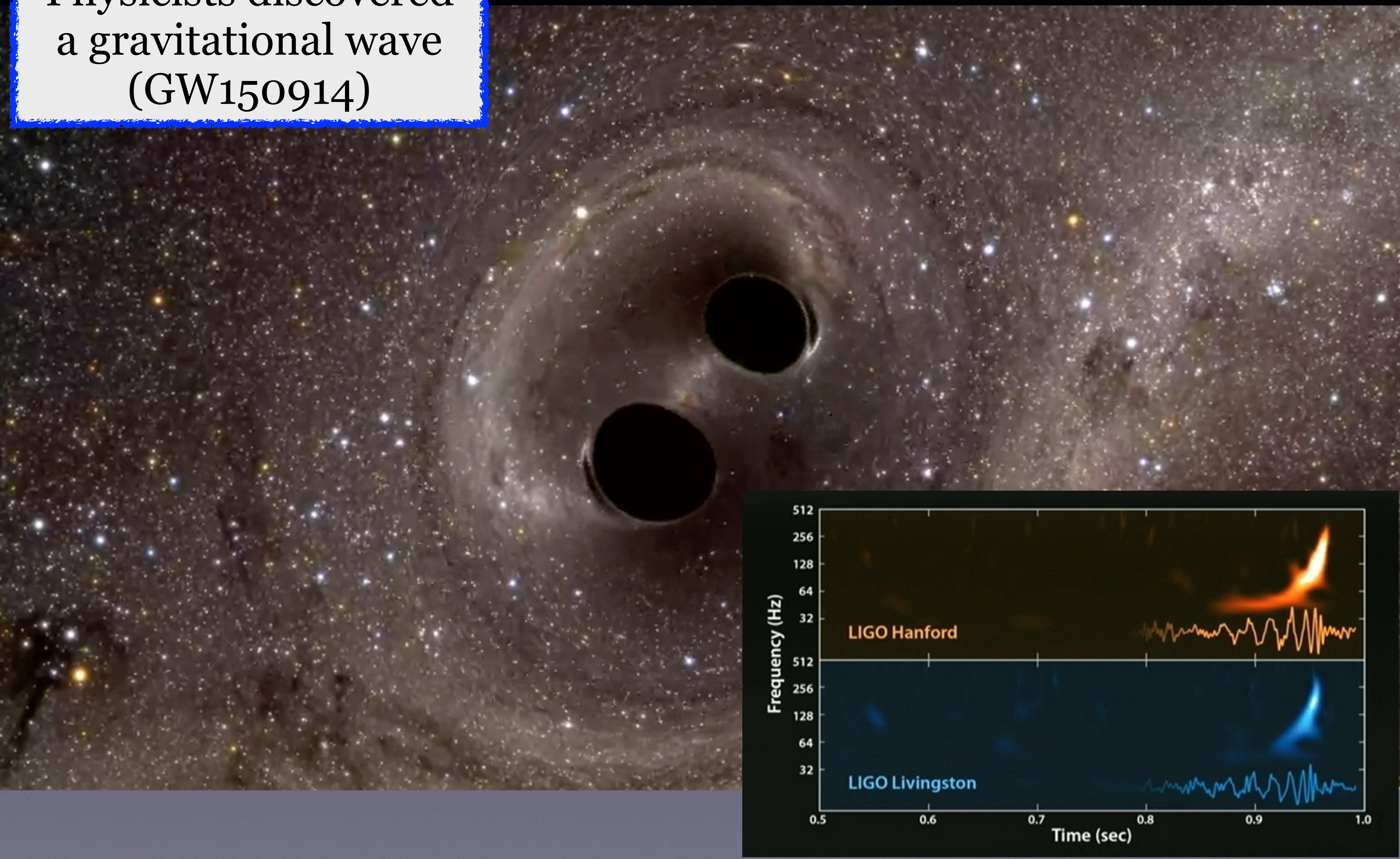
We celebrated discovery of non-zero θ_{13}, discovery of Higgs, etc.





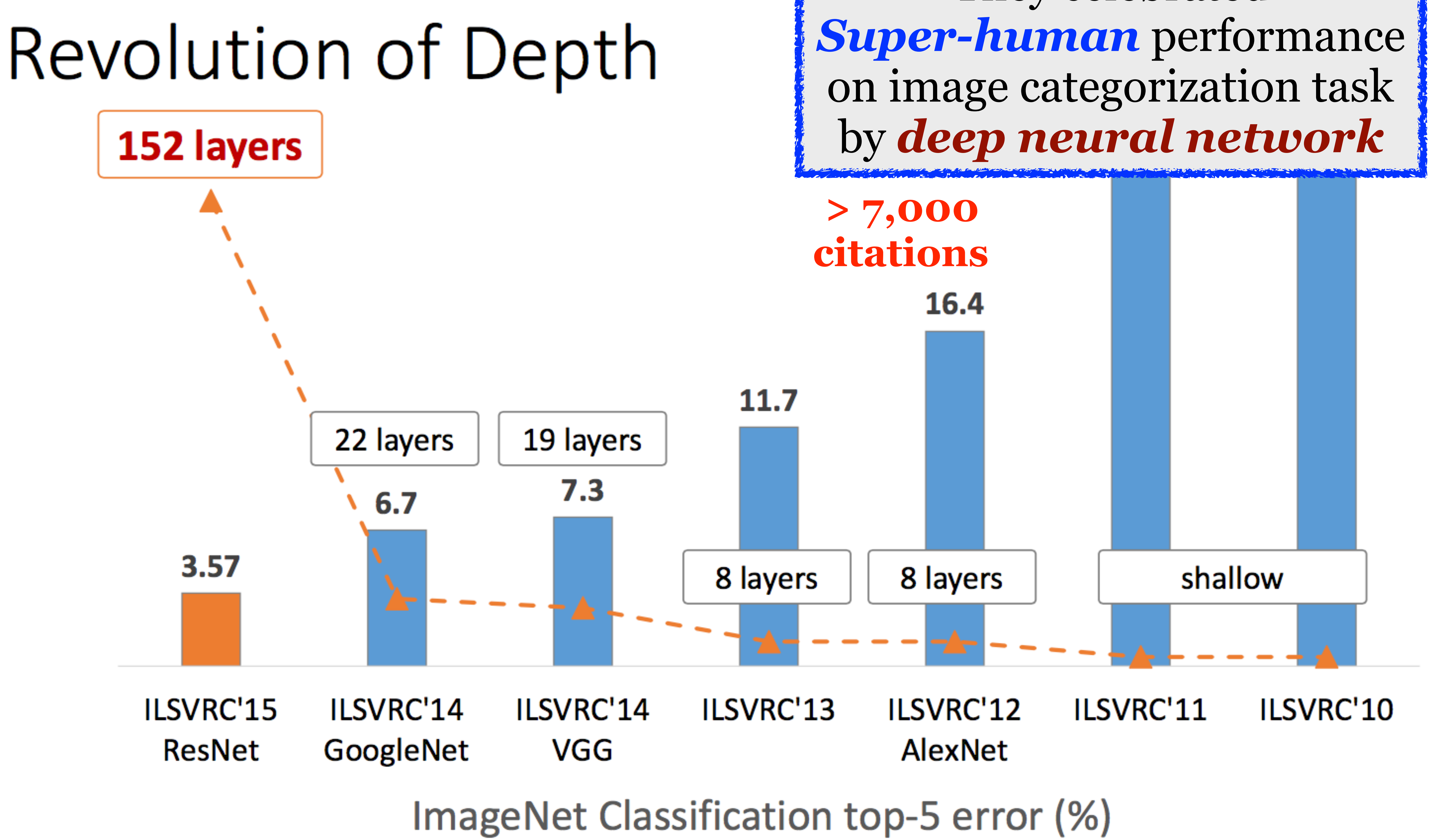
The Year of Breakthrough: 2015

Physicists discovered a gravitational wave (GW150914)



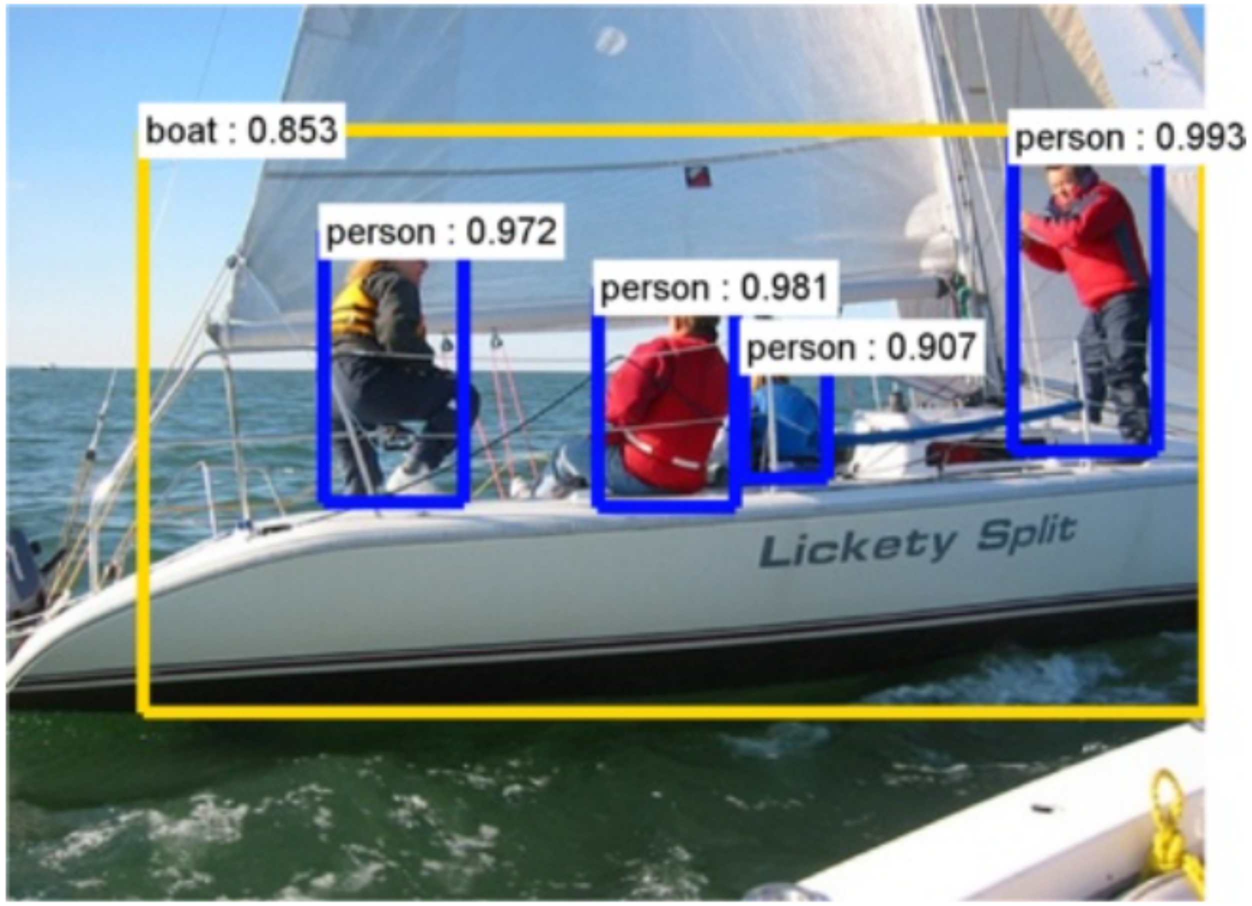
The Year of Breakthrough: 2015

They celebrated
Super-human performance
on image categorization task
by ***deep neural network***



The Year of Breakthrough: 2015

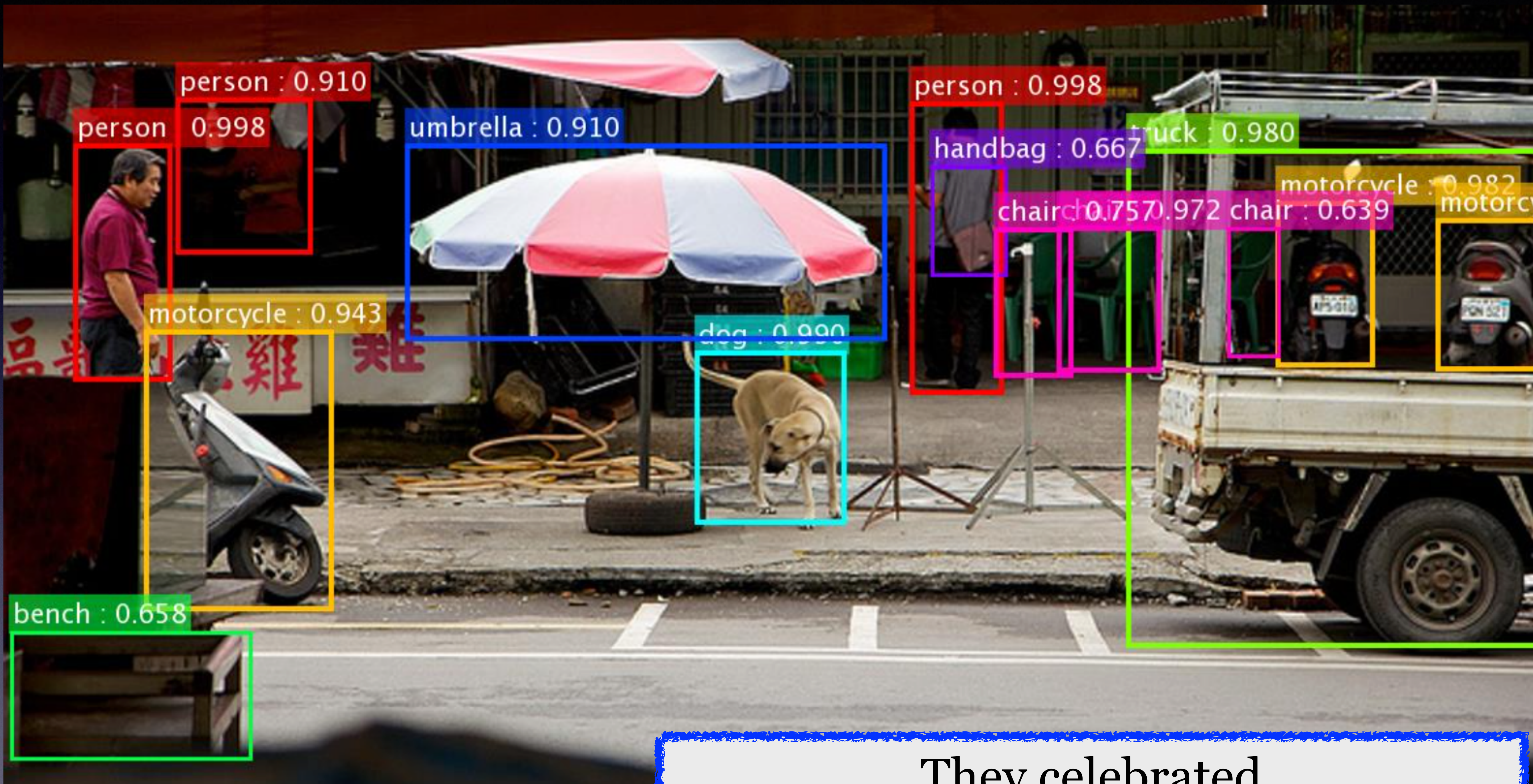
Image Classification
(what?)



Object Detection
(what + where?)

They celebrated
a demonstration of generalizability to
problems beyond image classification

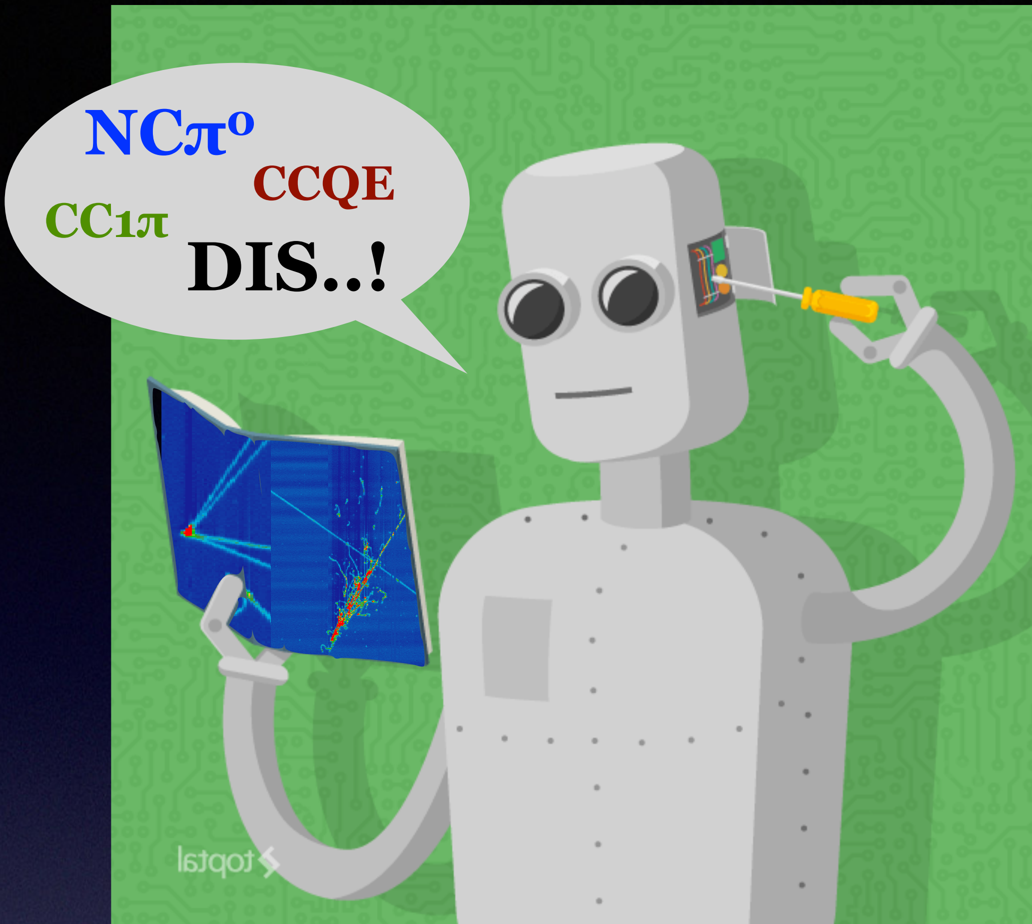
The Year of Breakthrough: 2015

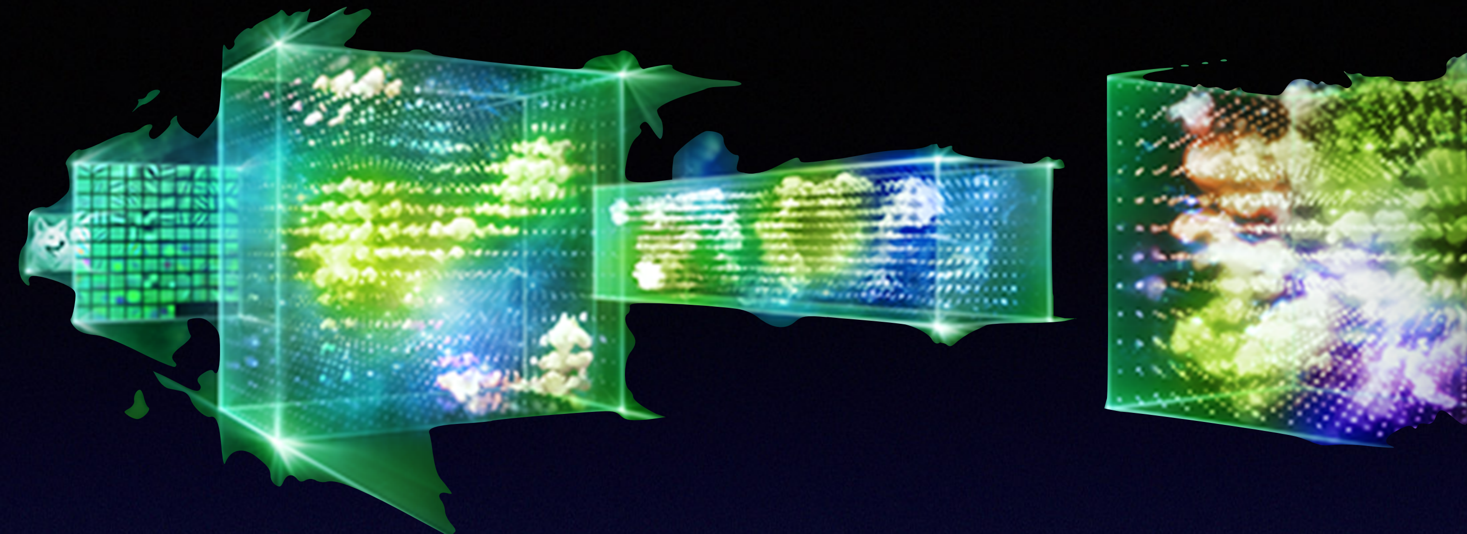


They celebrated
a demonstration of generalizability to
problems beyond image classification

The Year of Breakthrough: 2015

Pixel-level donuts detection





Deep Neural Network Applications

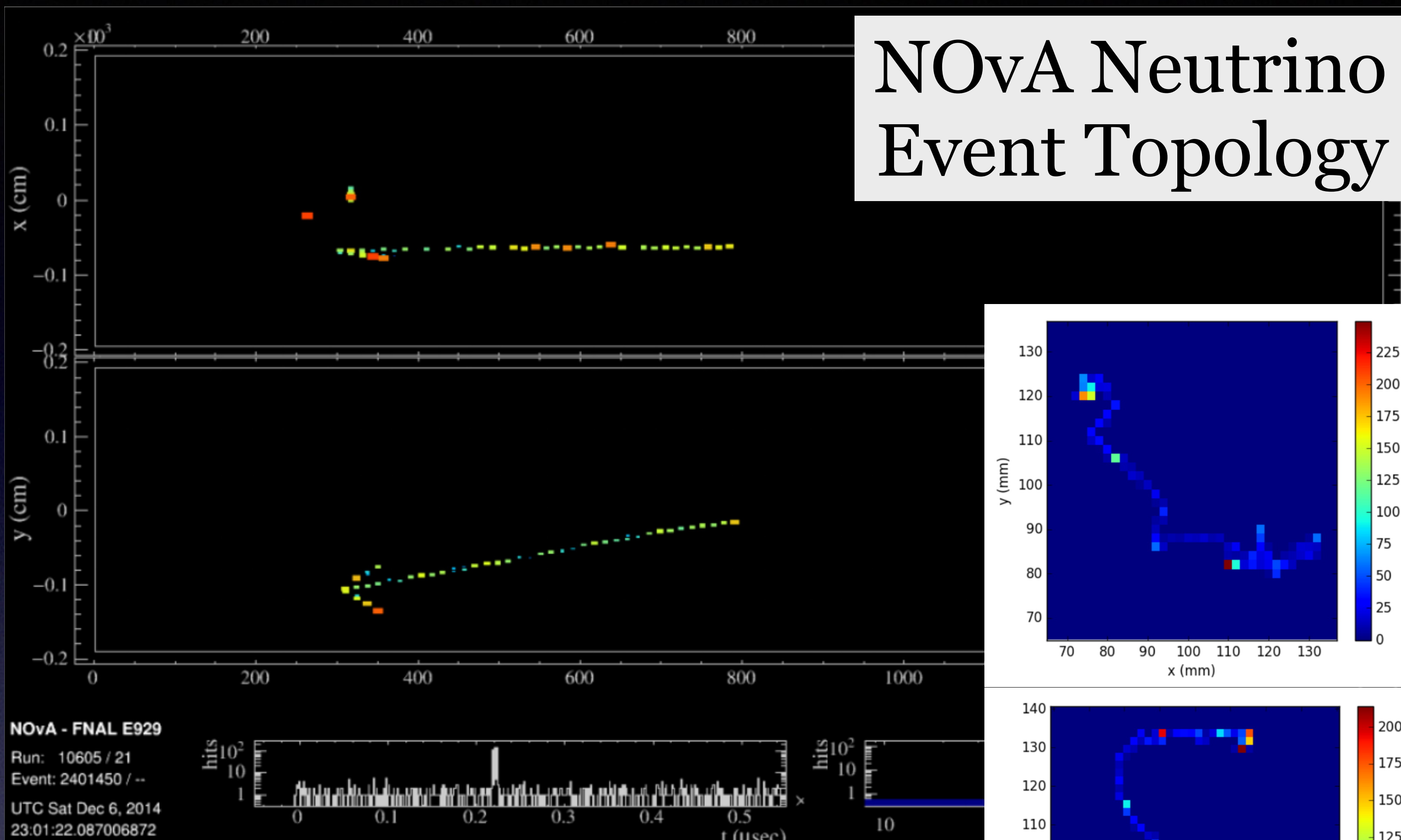
Outline

- Liquid Argon Time Projection Chambers
- Recent innovations in Computer Vision
- **Deep Neural Networks for data reconstruction**
- Wrap-up

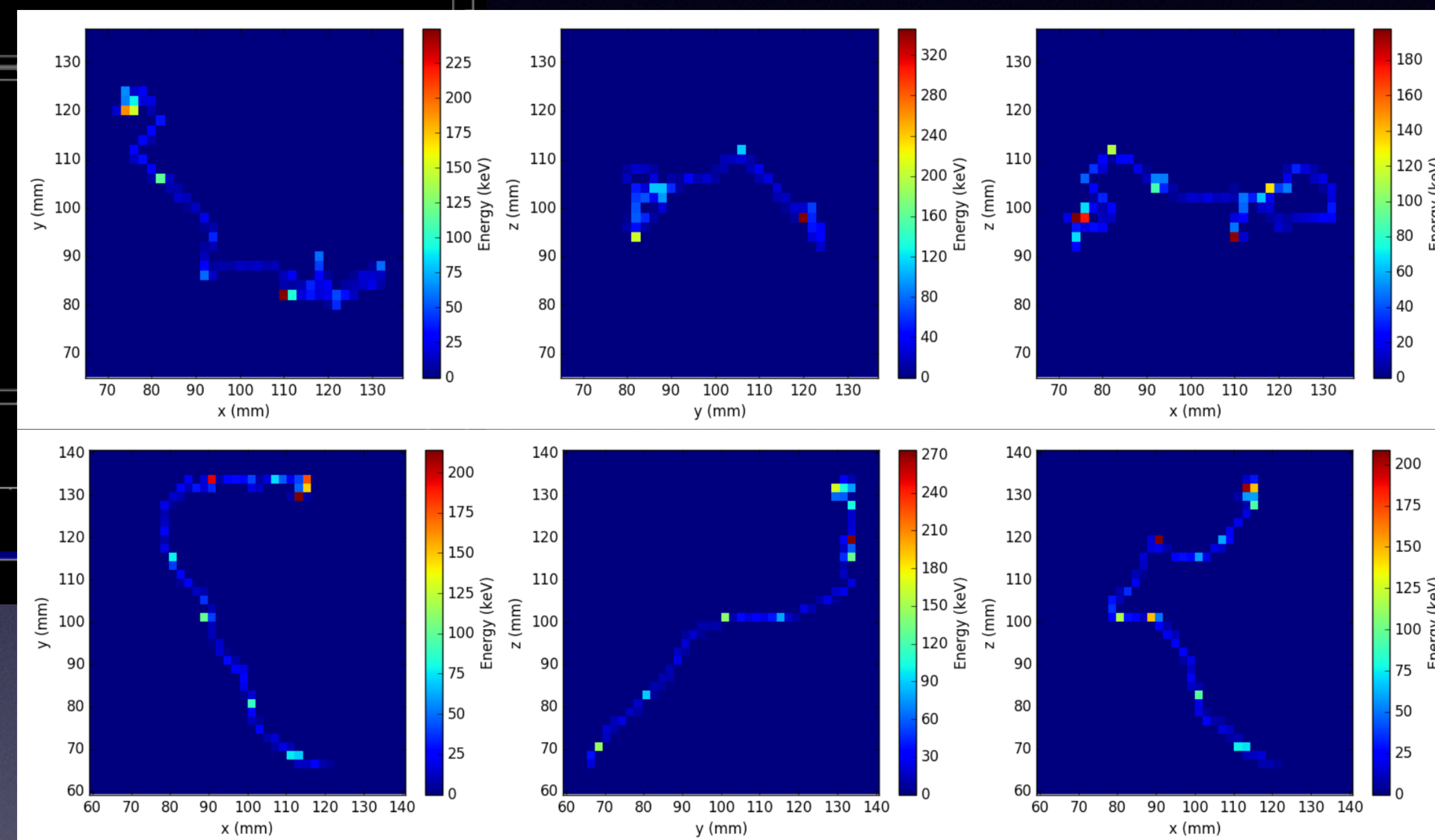
(~2016)

Image Classification for Physics Analysis

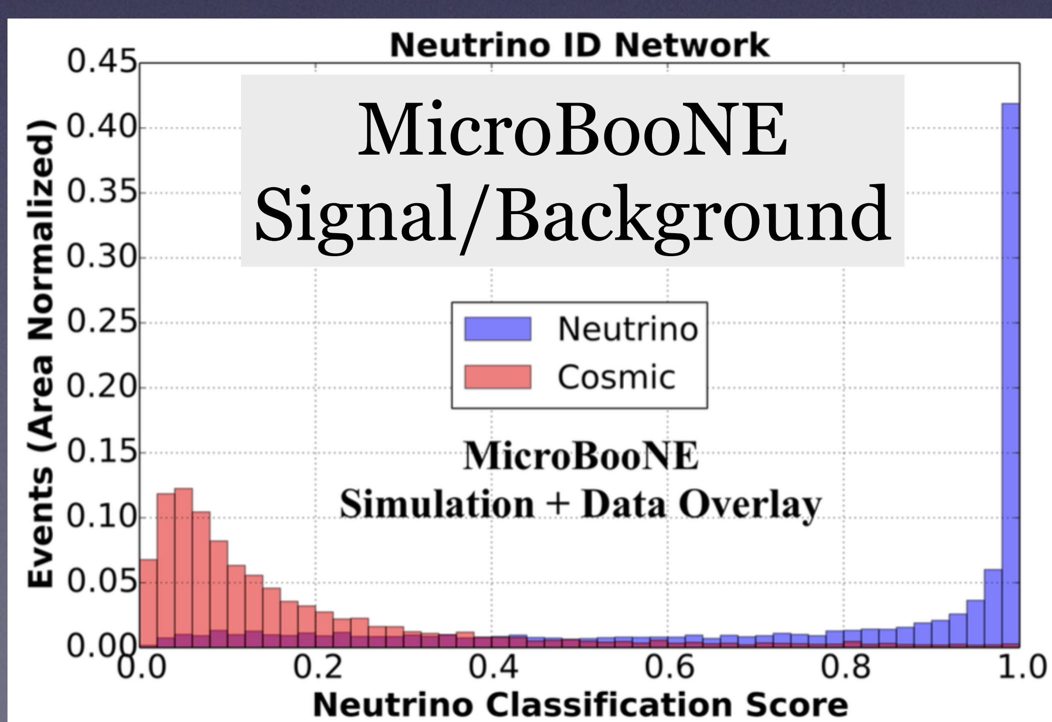
NOvA Neutrino Event Topology

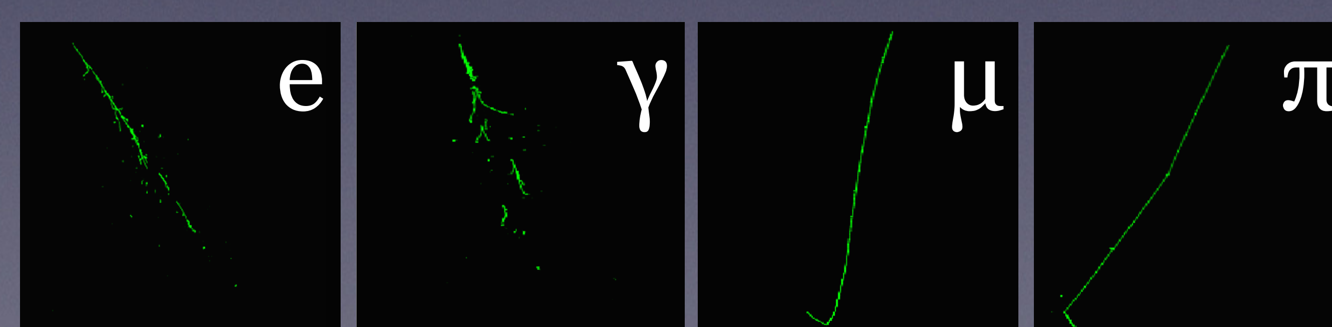


NEXT
Signal vs. Background



MicroBooNE
Signal/Background





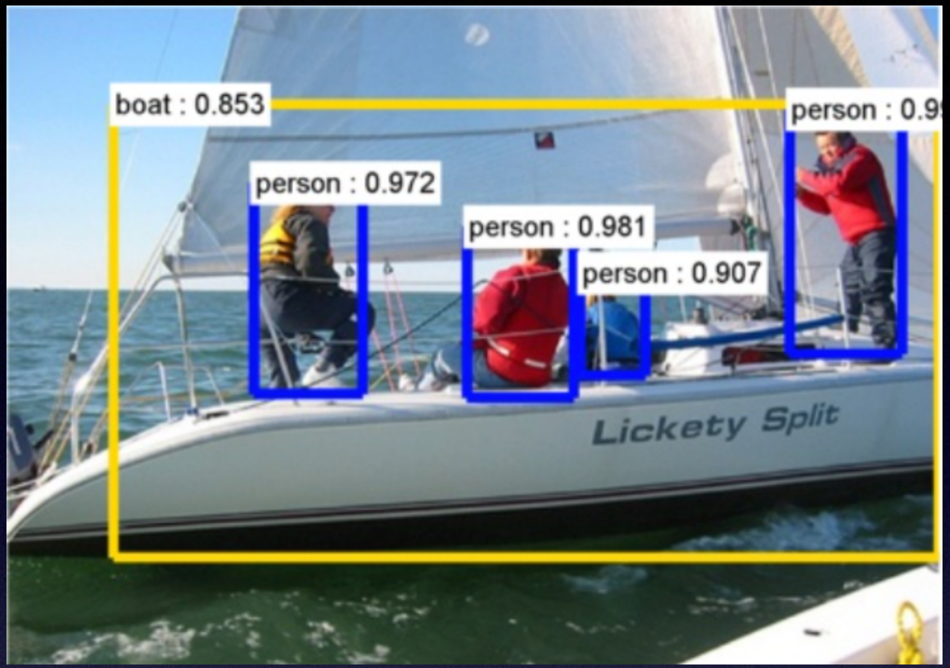
MicroBooNE Particle ID

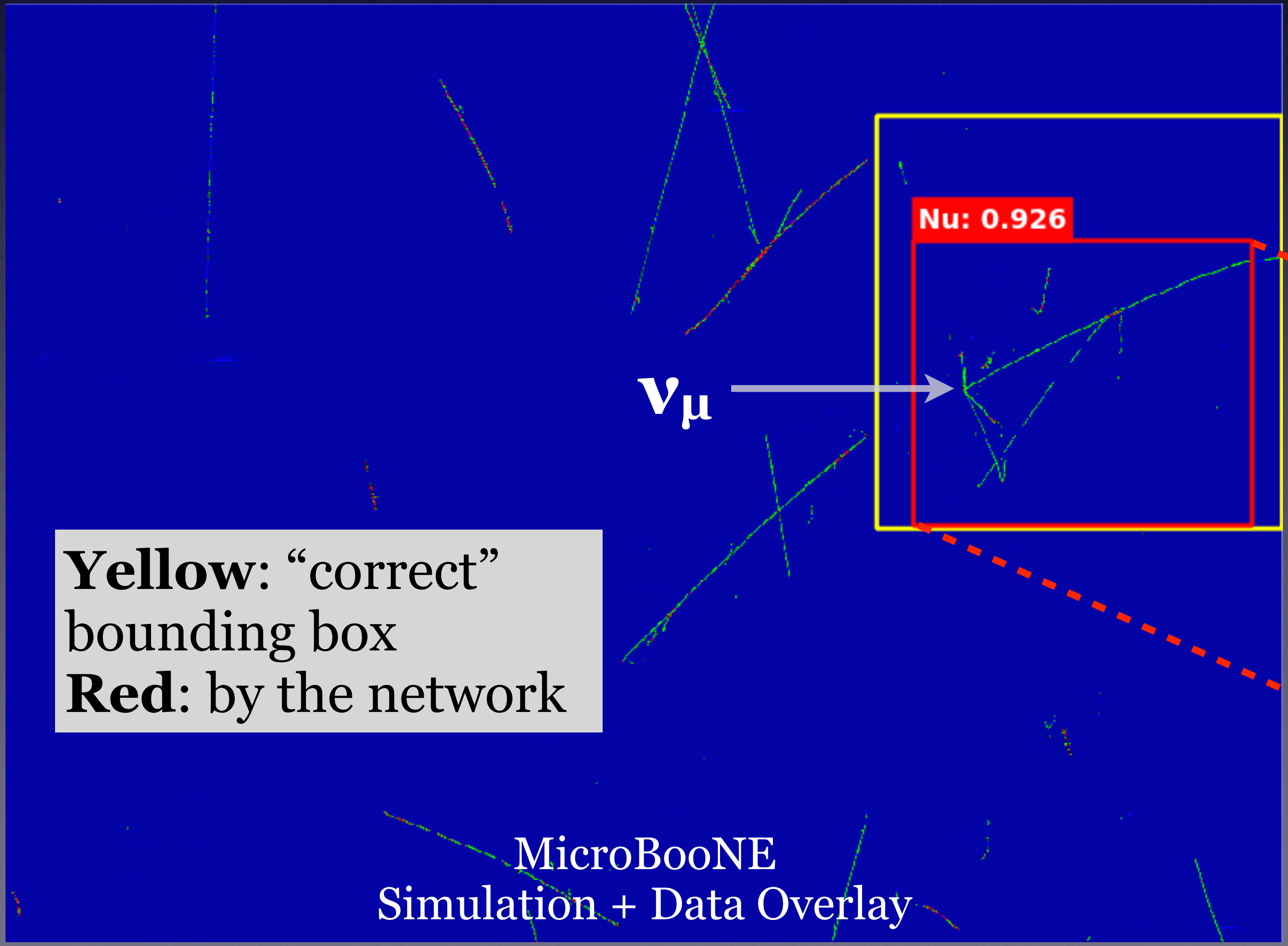
(~2016)

~~Image Classification~~ for Physics Analysis

Beyond: Object Detection

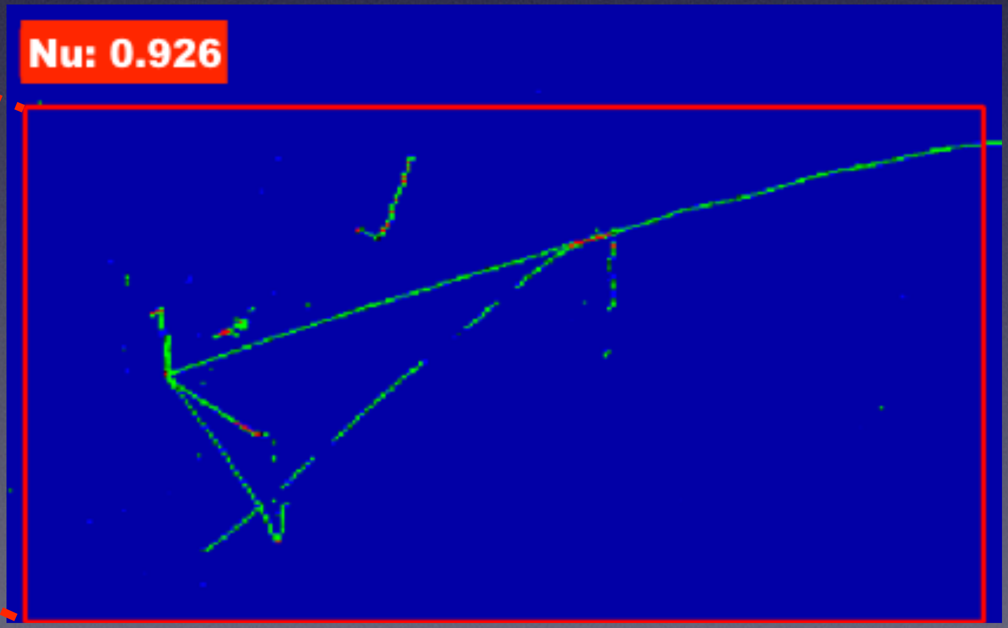
- Key insight: "**localize**" a "**distinct feature**" in data
- First step toward utilizing DNN for LArTPC data reconstruction





Yellow: "correct" bounding box
Red: by the network

arxiv:1611.05531

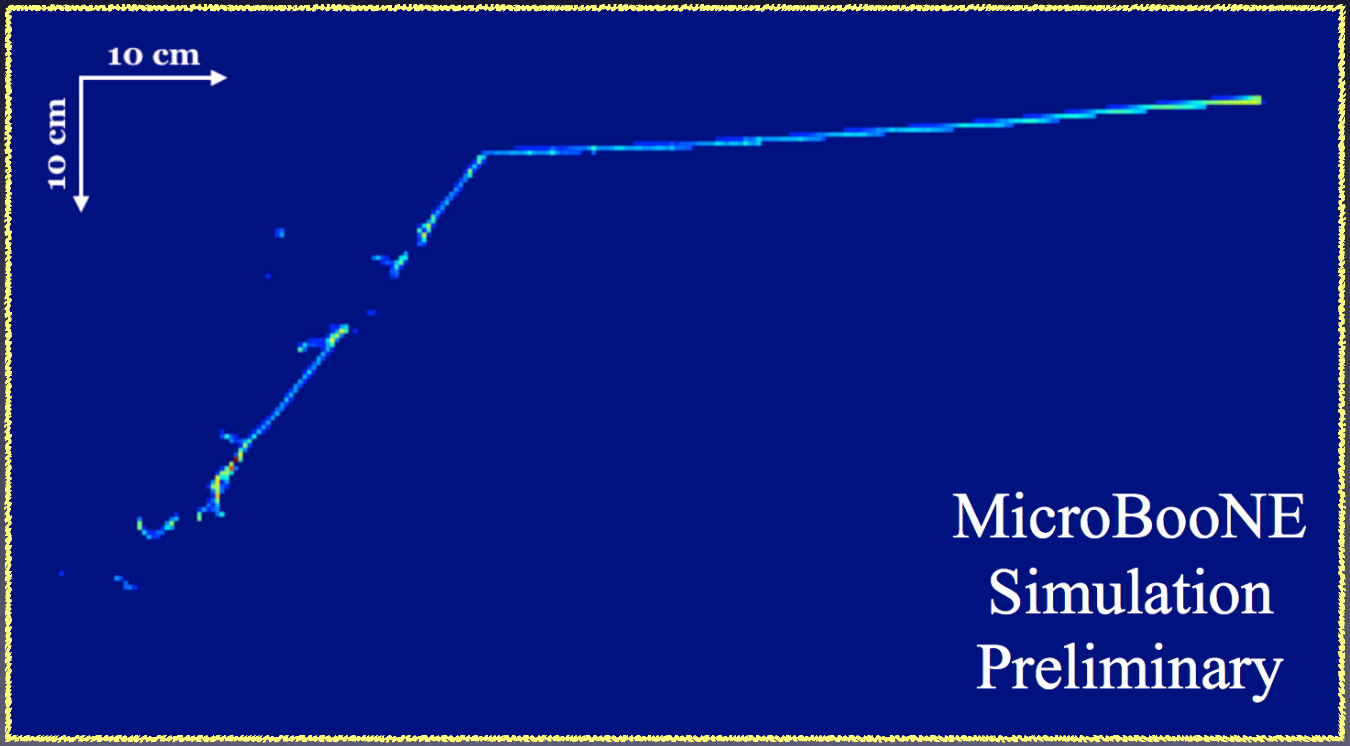


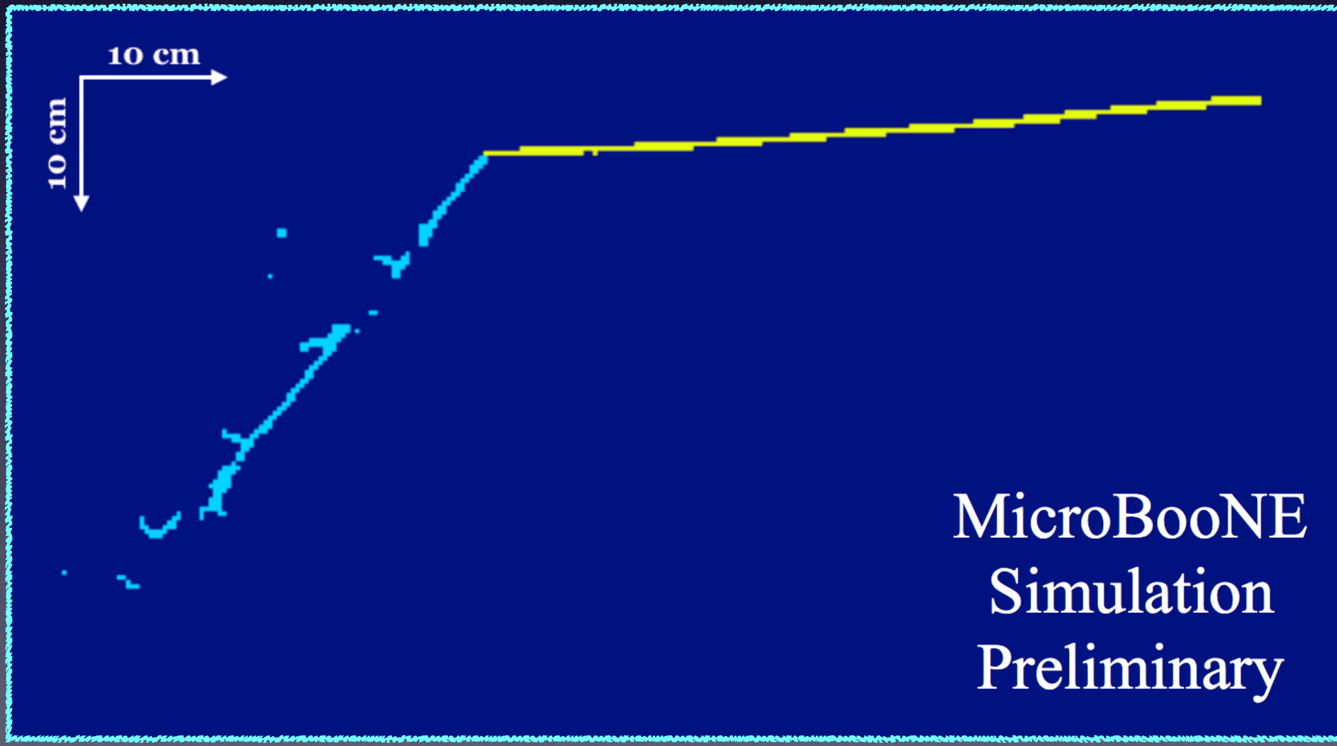
Network Output
≃ 2.6m (width) x 1 m (height)

Deep Neural Network for Reconstruction

Extreme localization at the pixel level

- U-ResNet can identify the pixel-level features
  - *Any* categorization at the pixel level (**reusable algorithm**)
  - Made one of the first fully automated ν_e search possible





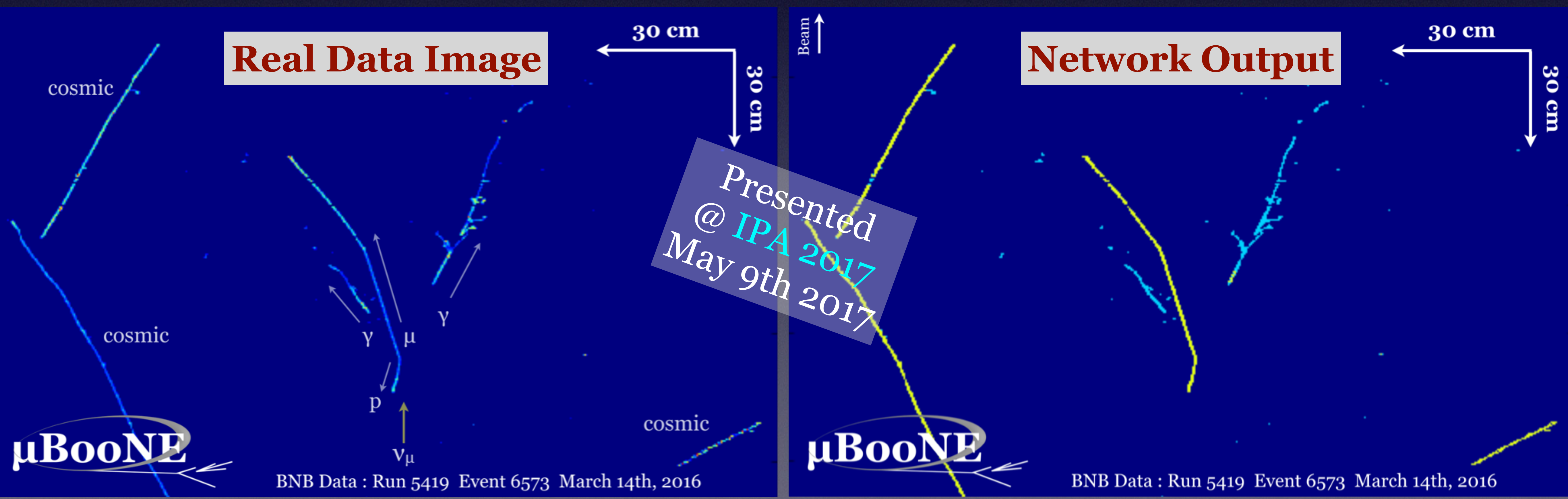
Example: **pixel-level categorization** by **U-ResNet**
(identify EM particles at full resolution)

(2016 ~2017)

Deep Neural Network for Reconstruction

Extreme localization at the pixel level

- U-ResNet can identify the pixel-level features
  - *Any* categorization at the pixel level (**reusable algorithm**)
  - Made one of the first fully automated ν_e search possible
  - Can analyze to find where "unexpected response" comes from



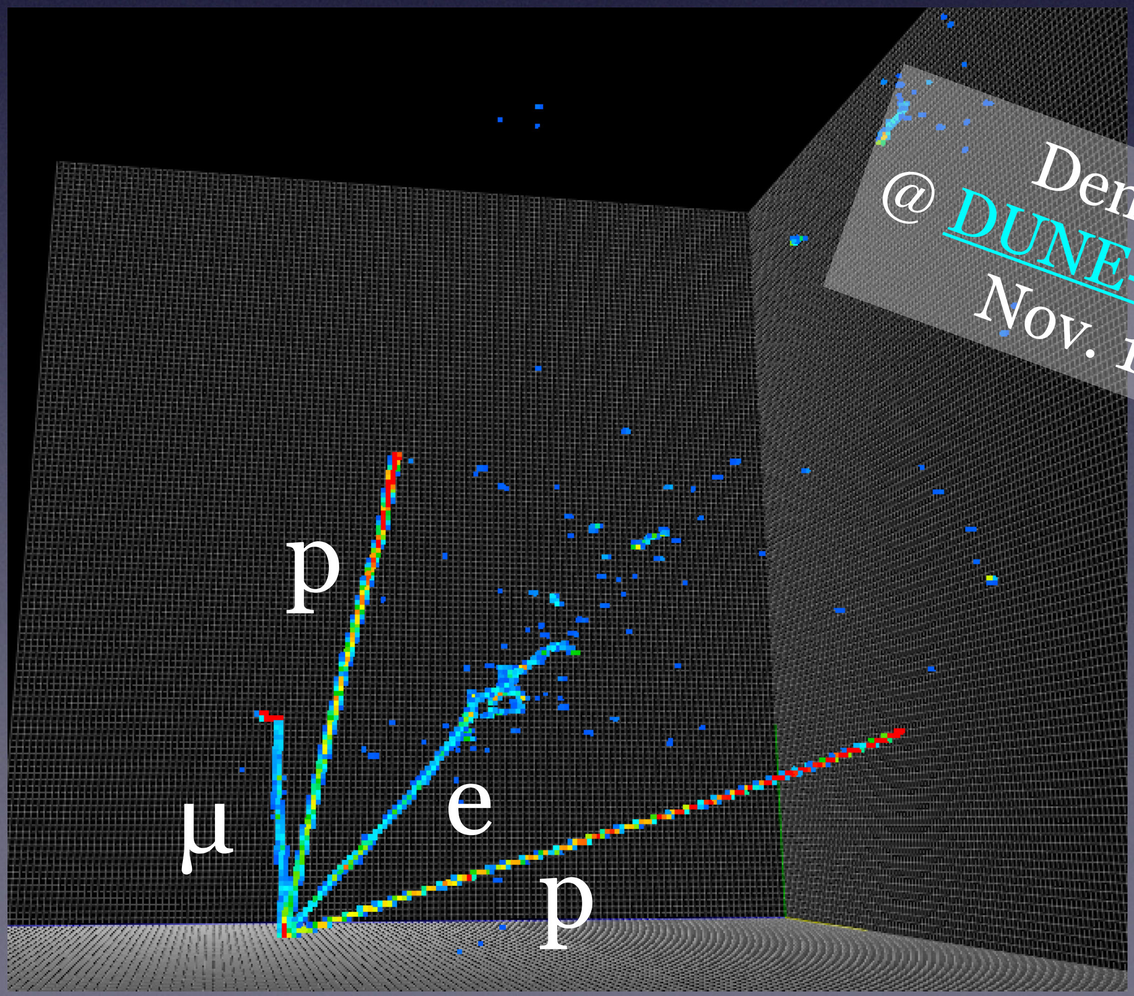
Example: **pixel-level categorization** by **U-ResNet**
(identify EM particles at full resolution)

(2017~)

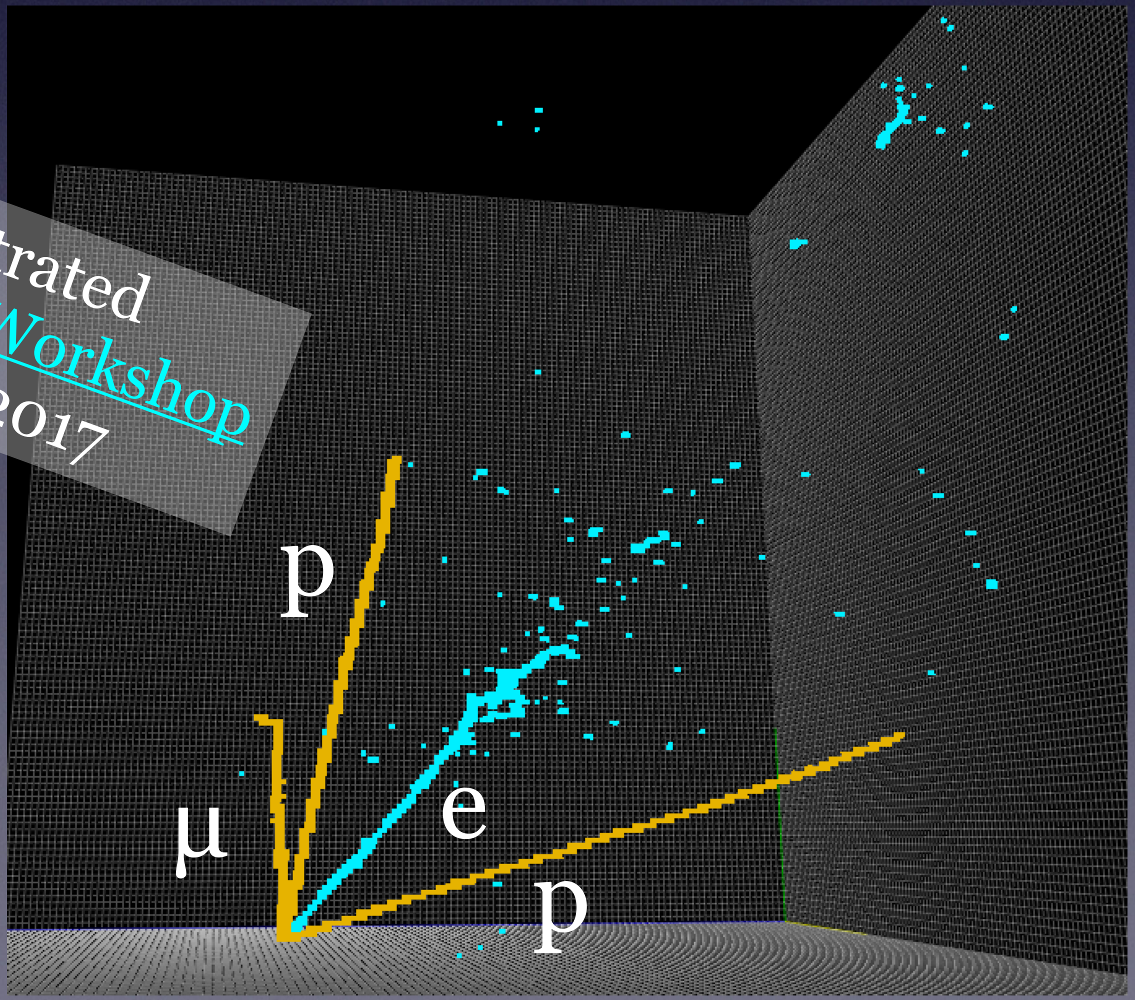
Deep Neural Network for Reconstruction

Extreme localization at the pixel level

- U-ResNet can identify the pixel-level features
  - *Any* categorization at the pixel level (**reusable algorithm**)
  - Made one of the first fully automated ν_e search possible
  - Can analyze to find where "unexpected response" comes from
  - Generalizable to both 2D and 3D data



Demonstrated @ DUNE-DL Workshop Nov. 13th 2017

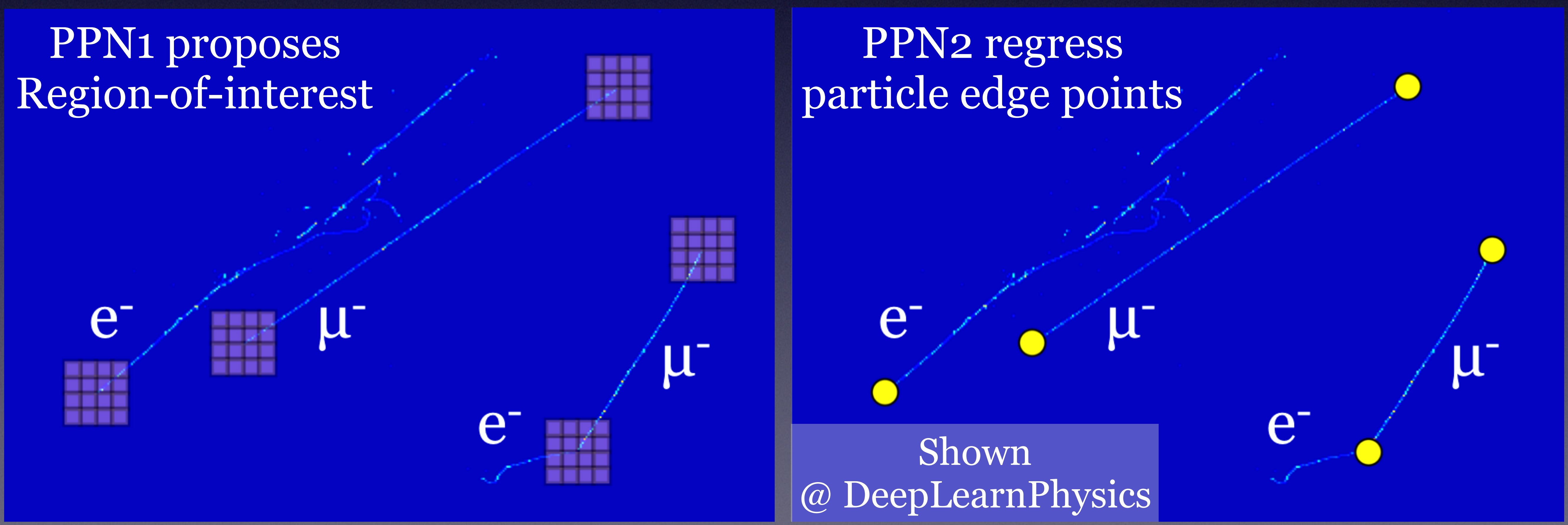


(2018~)

Deep Neural Network for Reconstruction

Feature space point finding via regression

- PPN: two piggy-backing subnetworks
  - Proposes "particle trajectory start & end" in 2D/3D
  - Can be attached to an image classification network or U-ResNet
  - Can run in *real time* (≃ 60 FPS for 756 x 756 pixels)



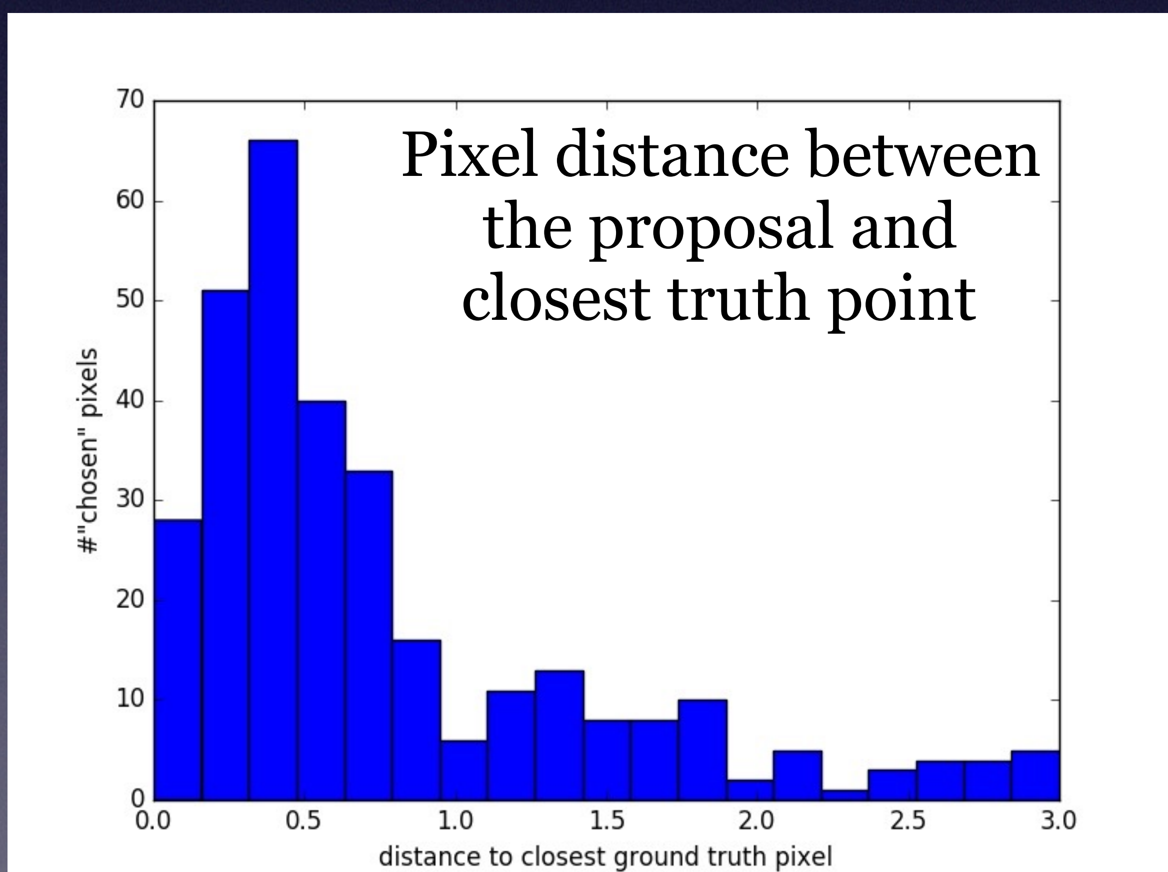
Example: **particle edge points prediction** by **PPN**

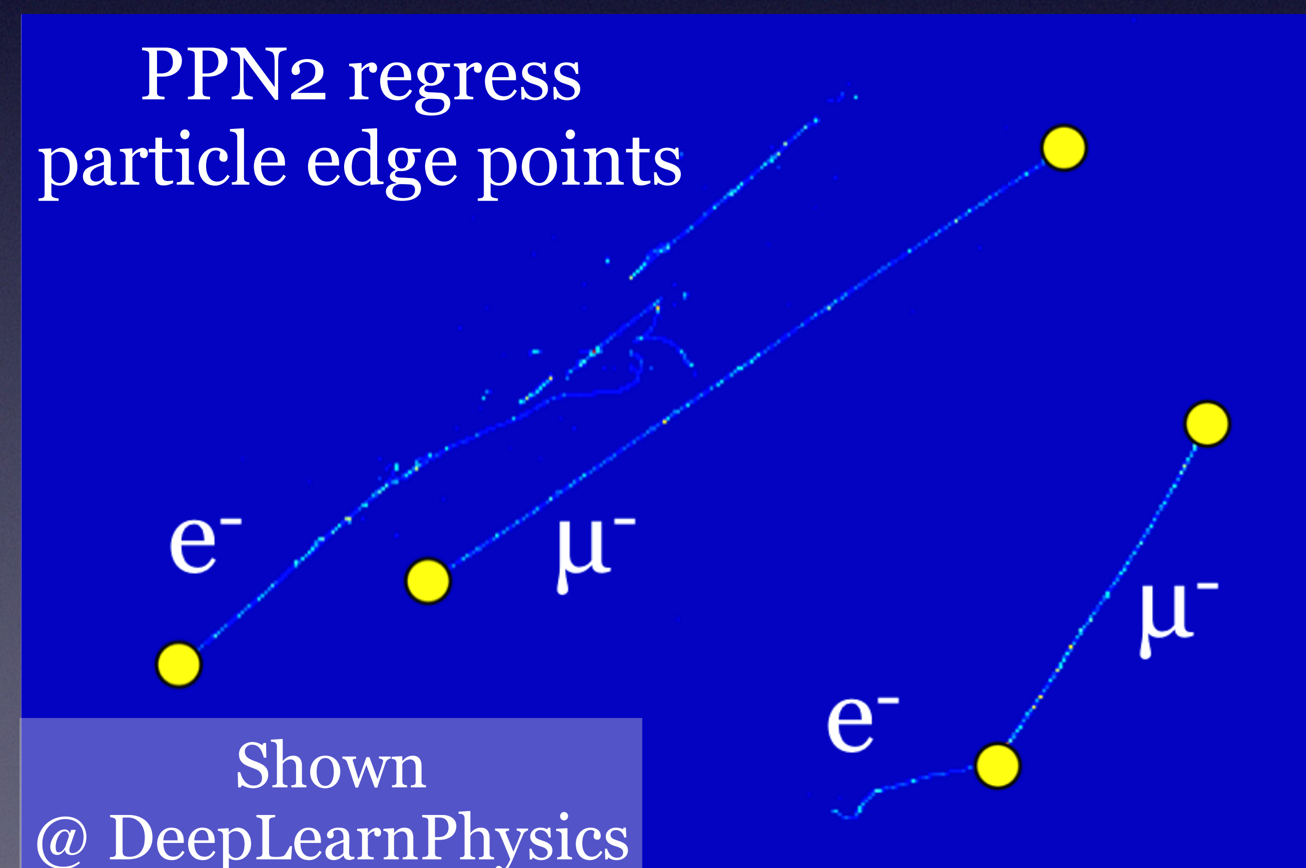
(In particular, piggy-backing on U-ResNet)

Deep Neural Network for Reconstruction

Feature space point finding via regression

- PPN: two piggy-backing subnetworks
  - Proposes "particle trajectory start & end" in 2D/3D
  - Can be attached to an image classification network or U-ResNet
  - Can run in *real time* (≃ 60 FPS for 756 x 756 pixels)





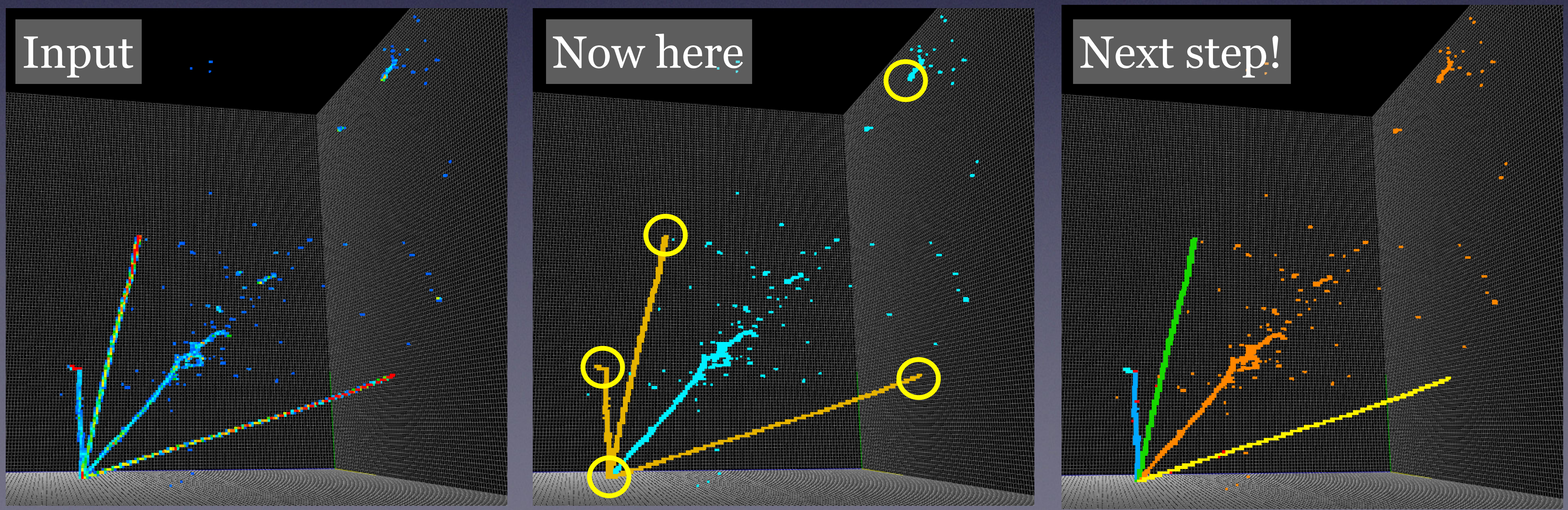
Example: **particle edge points prediction** by **PPN**

(In particular, piggy-backing on U-ResNet)

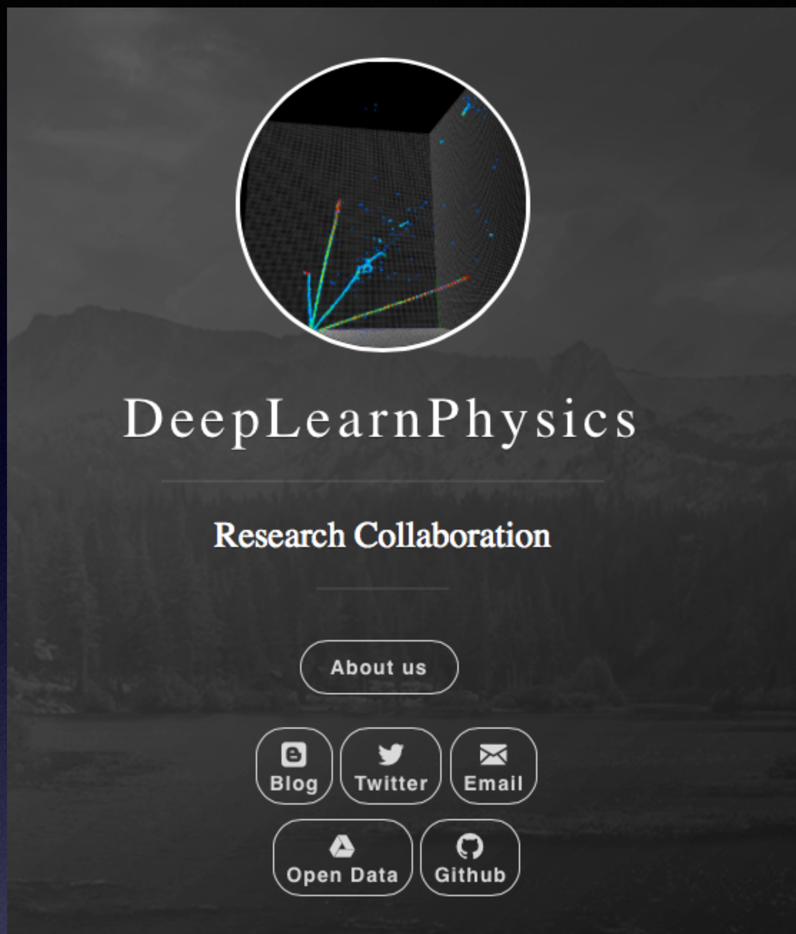
Deep Neural Network for Reconstruction

Where we are heading toward

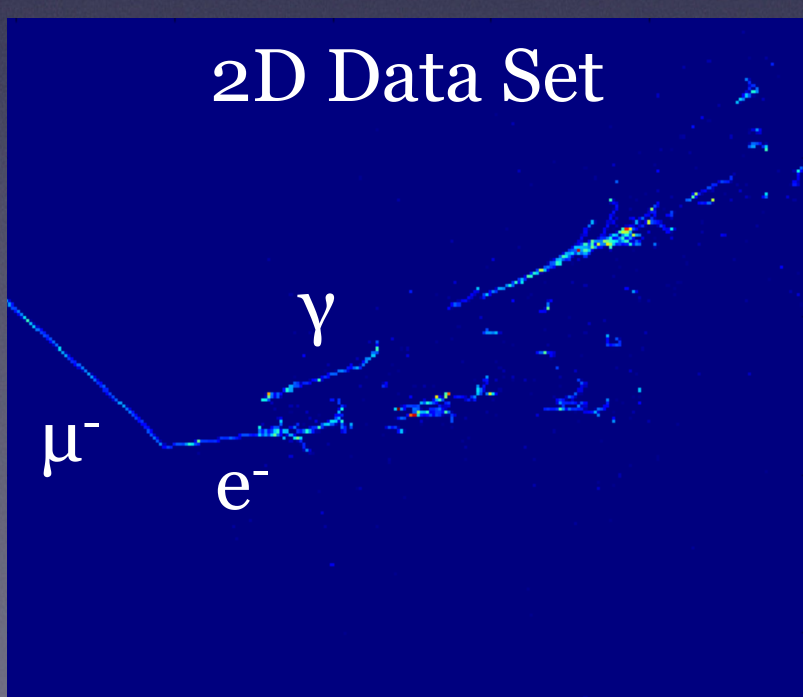
- **Full reconstruction chain**
  - Individual particle clustering & trajectory reconstruction
  - Interaction topology + particle hierarchy reconstruction
  - Energy reconstruction
- **Modular design:** plug & play to choose DNN vs. human-engineered algorithms, keep capability of full chain optimization

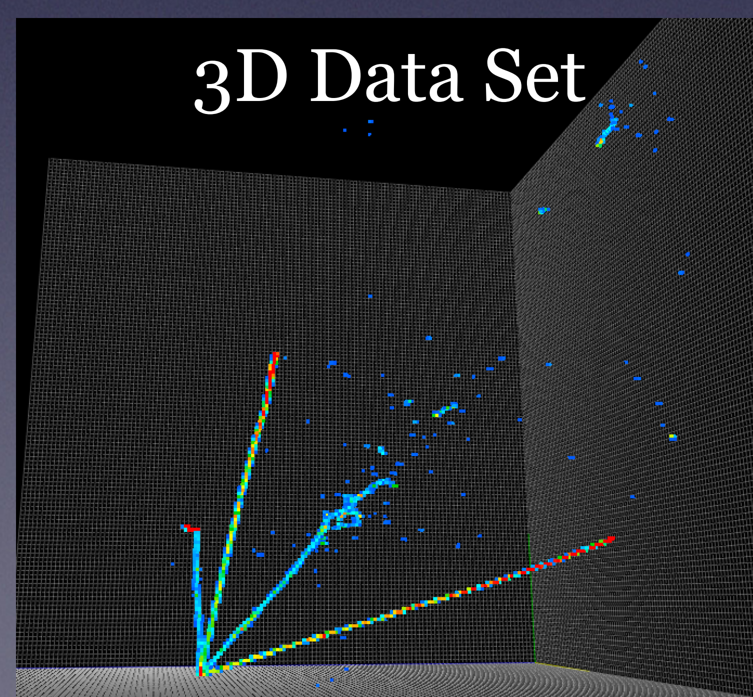


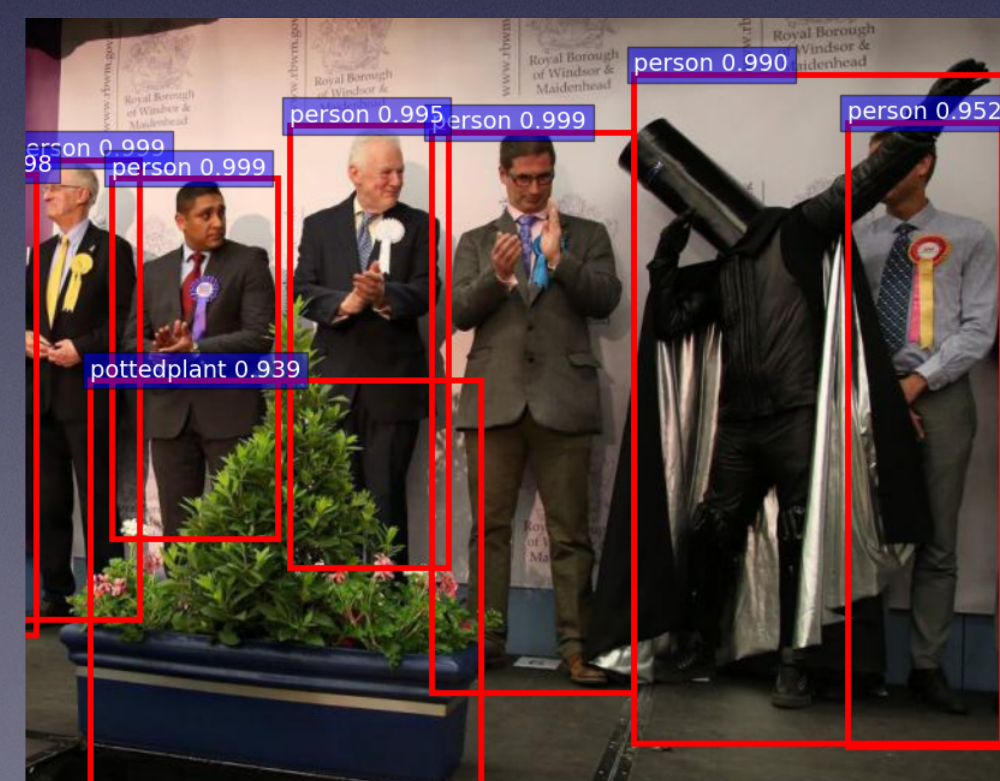
DeepLearnPhysics



- Group of physicists mainly from neutrino TPC experiments
  - http://deeplearnphysics.org
  - MicroBooNE/SBND/ICARUS/DUNE/NEXT/nEXO/non-HEP ...
- Share software tools + open data
- Meetings/Blog posts to share experience, discuss problems, etc.





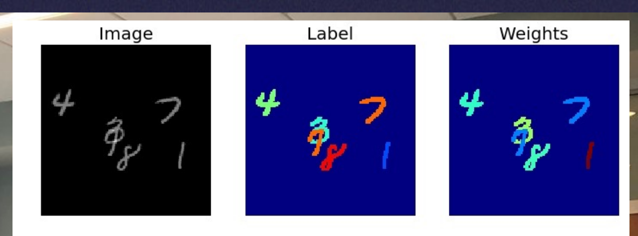


DeepLearnPhysics

Open-Source Software Development @ DLP

- Image/Volumetric data processing framework
  - Experiment agnostic design, Qt/OpenGL based visualization toolkit, C++/CUDA based software with extensive Python APIs
  - Interface to DL frameworks (MXNet, Pytorch, Tensorflow), Singularity container distribution for cloud deployment

Hands-on workshop
@ SLAC/Stanford
@ March 2018

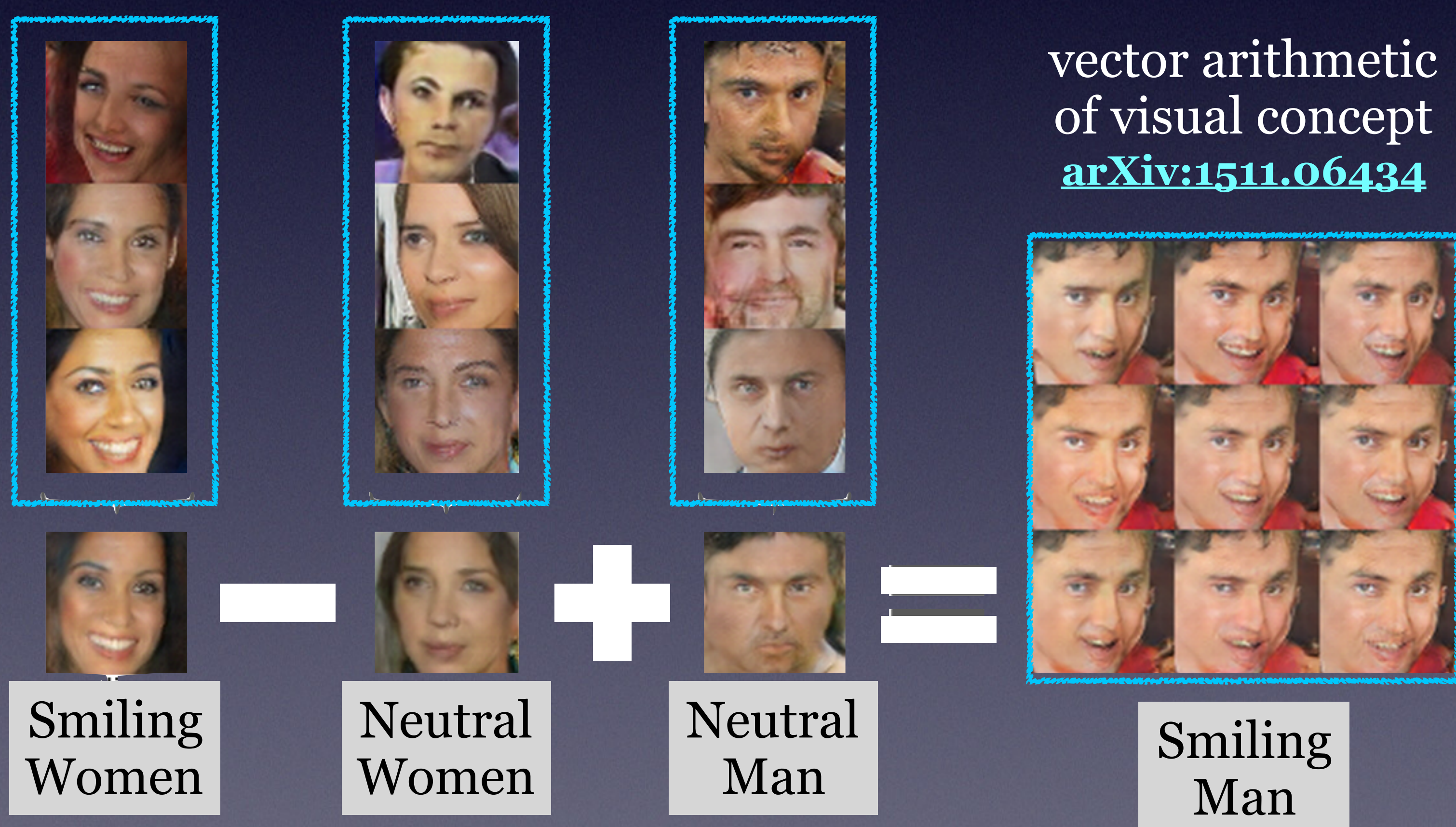


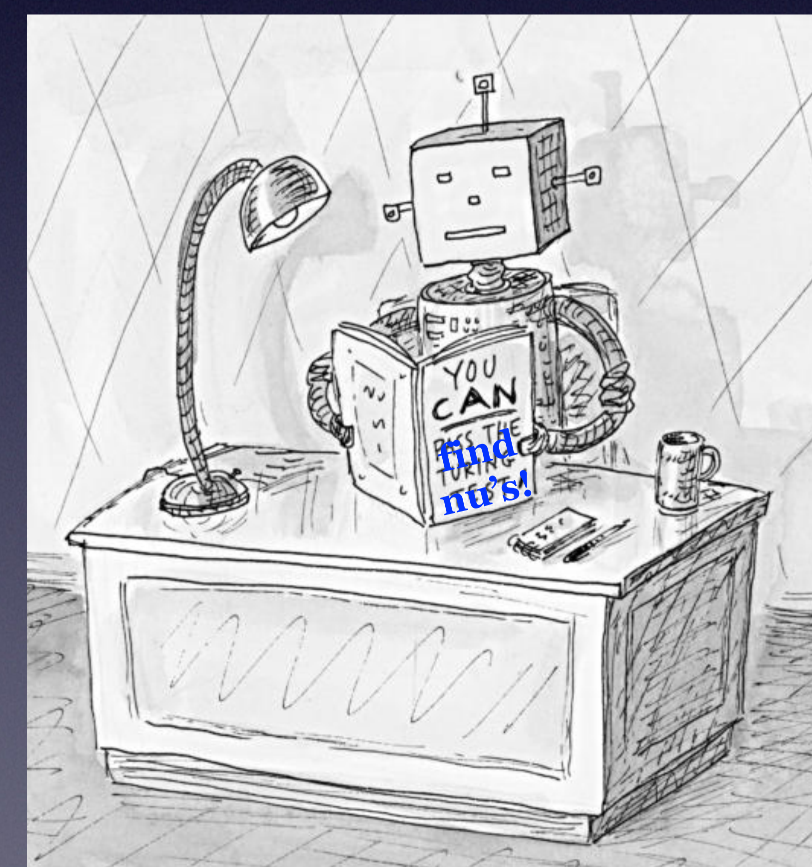
Workshops to share & raise expertise

- "GPU for everyone": using free K80 GPU from Google cloud
- Where we synergy across fields
  - Collaboration with Stanford campus CS/ML, Cryo-EM, accelerator, photon-science,

... more exciting projects ...

vector arithmetic
of visual concept
arXiv:1511.06434





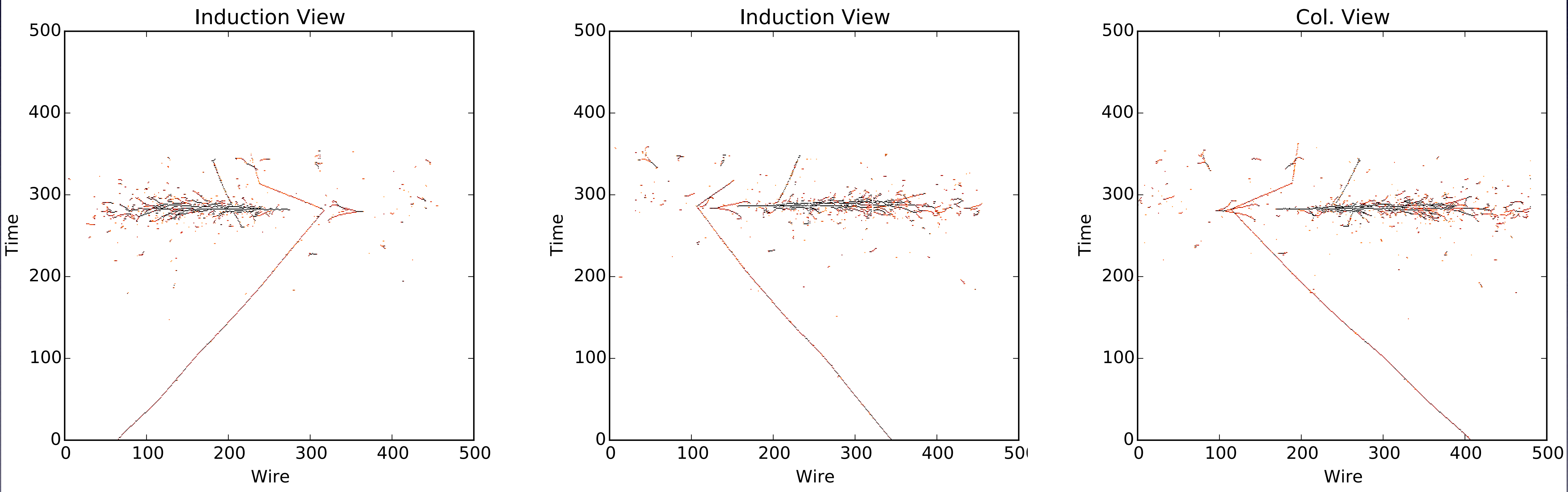
SBND Cosmic Rejection w/ U-ResNet

Collection plane view,
similar performance
on induction planes
(from C. Adams)

Our Input

DL @ DUNE FD Analysis

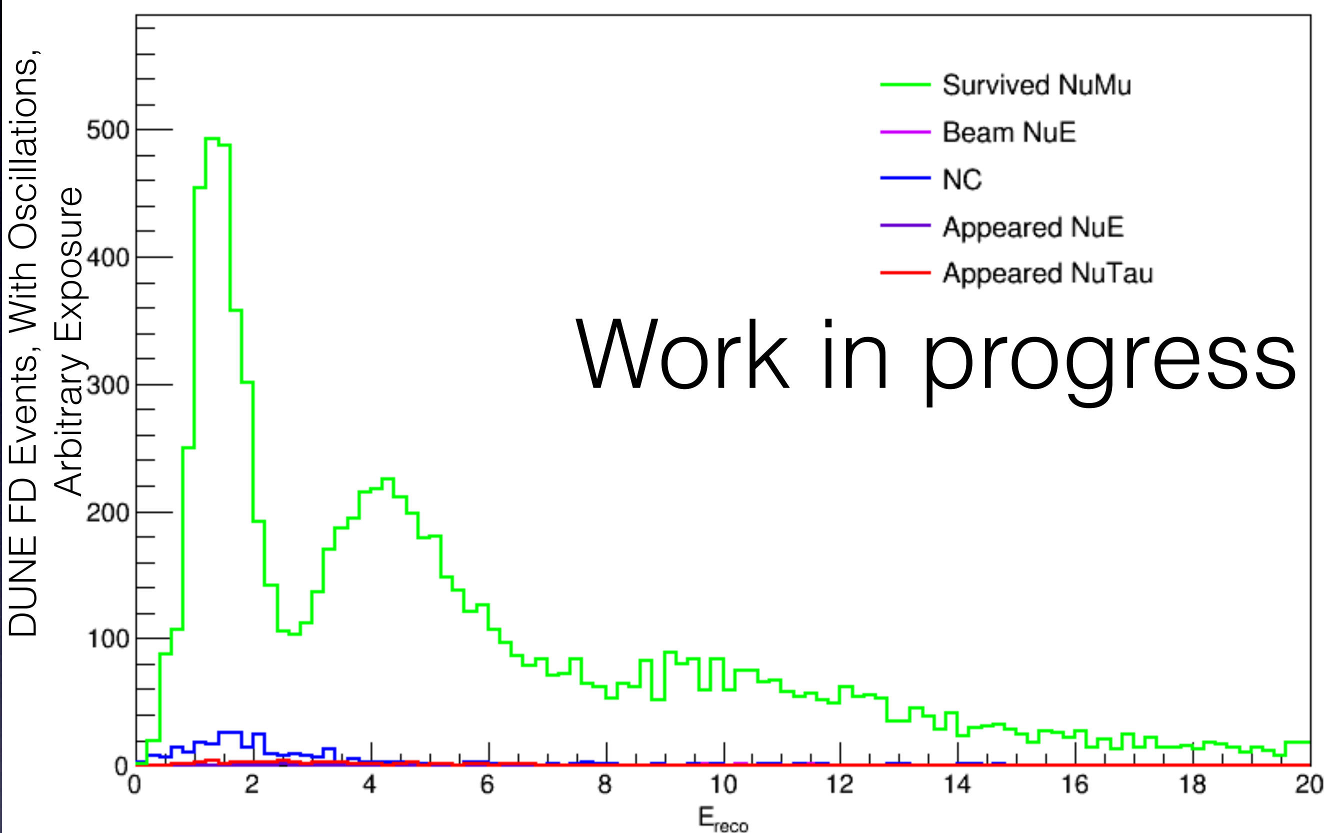
Each "pixel" is the integrated ADC response in that time/space slice. These maps are chosen to be 500 wires long and 1.2ms wide (split into 500 time chunks).



NuMu Selected Events Reconstructed Energy Spectr

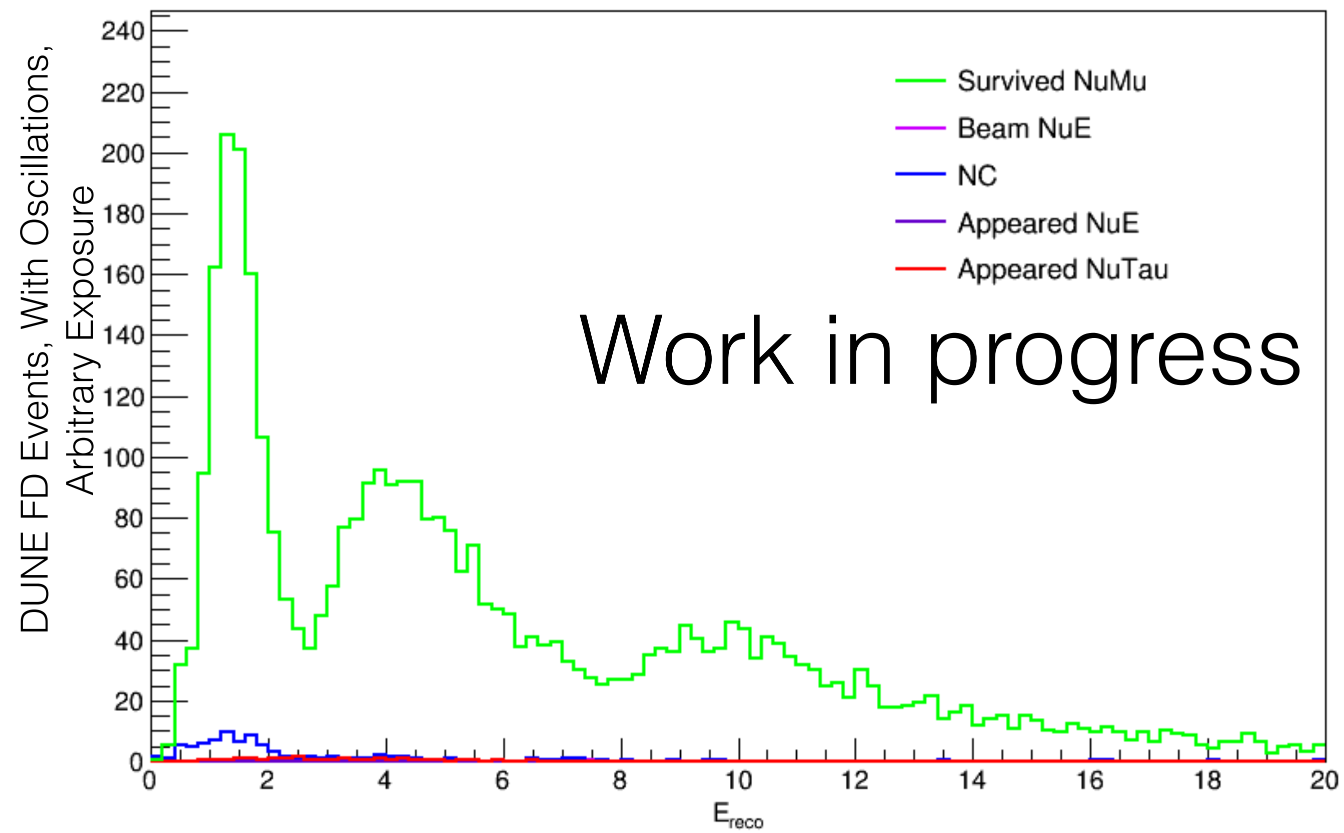
DL @ DUNE FD Analysis

Neutrino Beam



	NuMu	Appeared NuE	Beam NuE	NC	NuTau
Efficiency	**80.6**				
Rejection		99.0	98.7	**97.6**	81.5

Anti-Neutrino Beam

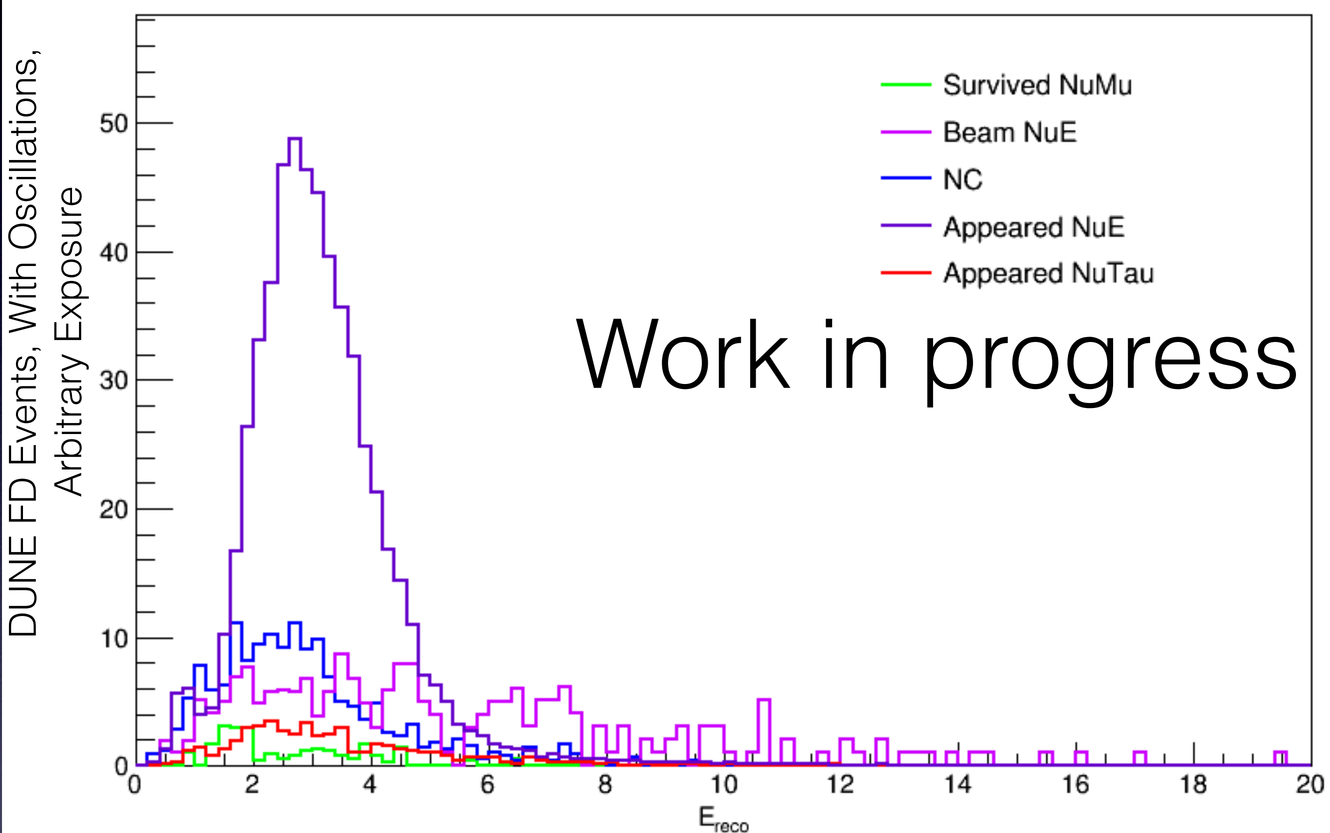


	NuMu	Appeared NuE	Beam NuE	NC	NuTau
Efficiency	**87.7**				
Rejection		99.6	99.3	**98.3**	81.4

NuE Selected Events, Reconstructed Energy Spectra

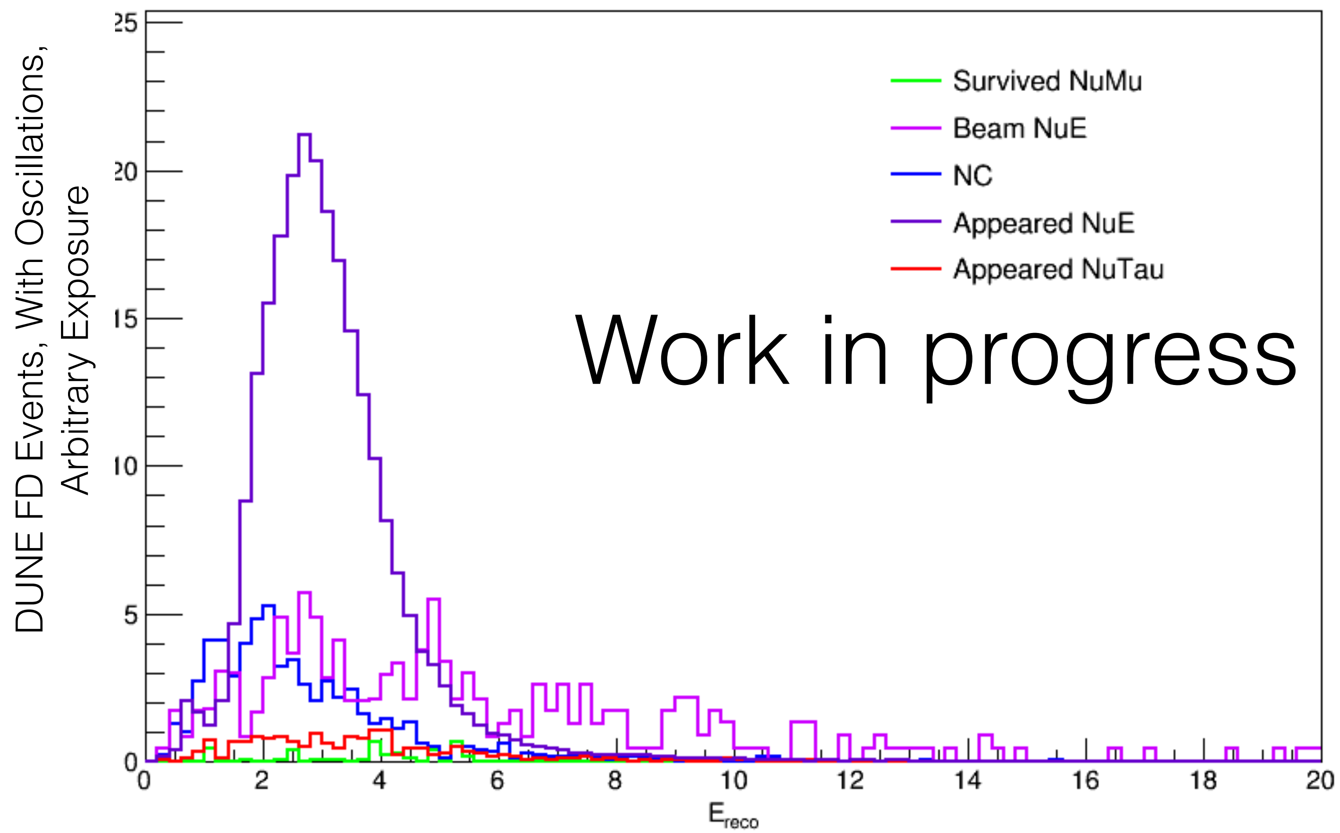
DL @ DUNE FD Analysis

Neutrino Beam



	Appeared NuE	NuMu	Beam NuE	NC	NuTau
Efficiency	**67.5**				
Rejection		99.8	52.1	**98.6**	85.8

Anti-Neutrino Beam



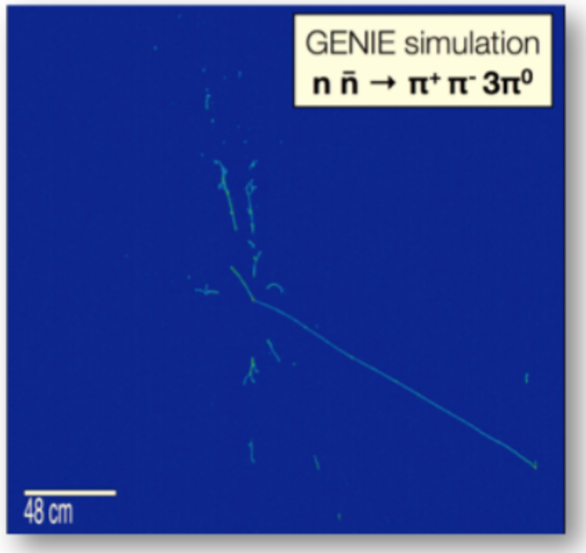
	Appeared NuE	NuMu	Beam NuE	NC	NuTau
Efficiency	**79.3**				
Rejection		99.9	48.2	**98.8**	87.6

n-nbar Search in DUNE FD

Deep Learning application for rare event searches (and more) in DUNE

Group: Georgia Karargiorgi (Columbia U/U Manchester), Jeremy Hewes (formerly U Manchester), Yuyang Zhou (Columbia U)

CNN application in DUNE: originally developed as a DL-based analysis for a search for **rare neutron-antineutron oscillation events** (B-violating signature) in DUNE.



Simulated n-nbar event in DUNE; striking ("star event") topology

MANCHESTER 1824 The University of Manchester

NEVIS LABORATORIES COLUMBIA UNIVERSITY

n-nbar Search in DUNE FD

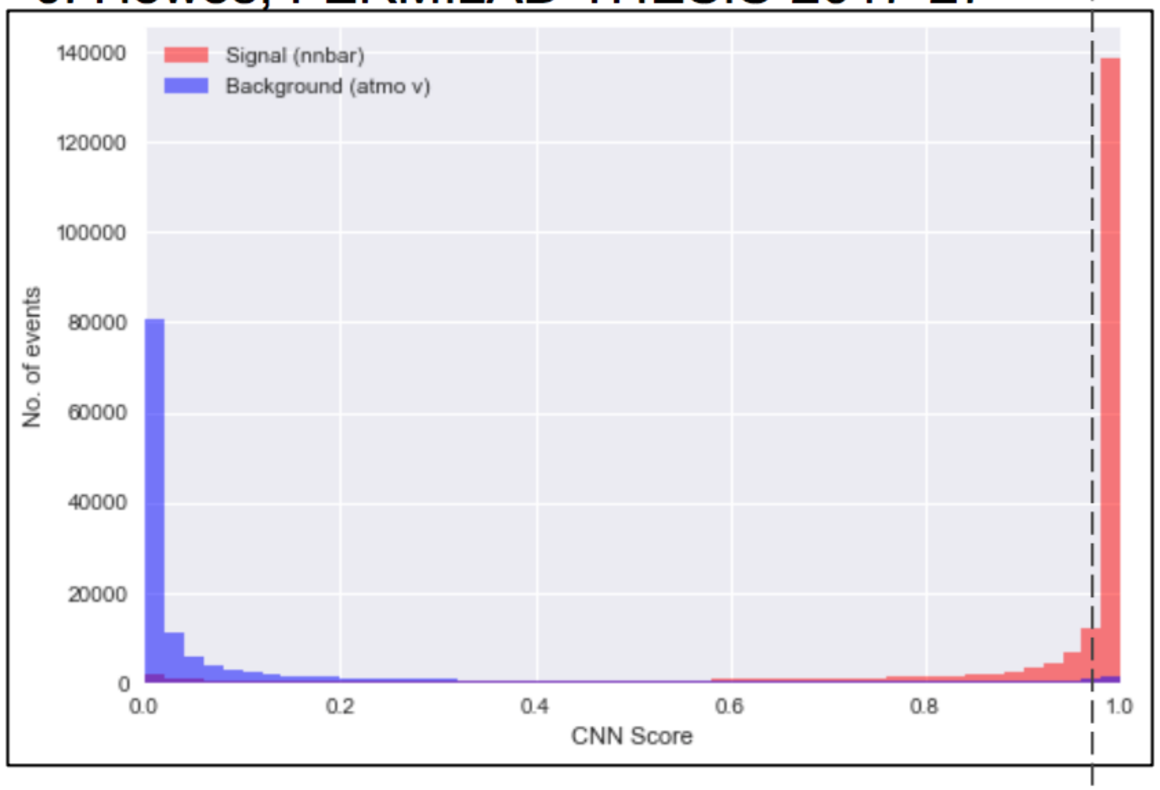
CNN-based search for n-nbar in DUNE

vgg16 network

Trained to differentiate n-nbar events from atmospheric neutrino events* (training samples of 50k events), and tested (samples of 200k events).

*atmospheric neutrino events expected to be the dominant background in DUNE

J. Hewes, FERMILAB-THESIS-2017-27



An optimized cut on CNN score yields signal efficiency of 14% background mis-ID rate of 0.003%

MANCHESTER 1824 The University of Manchester

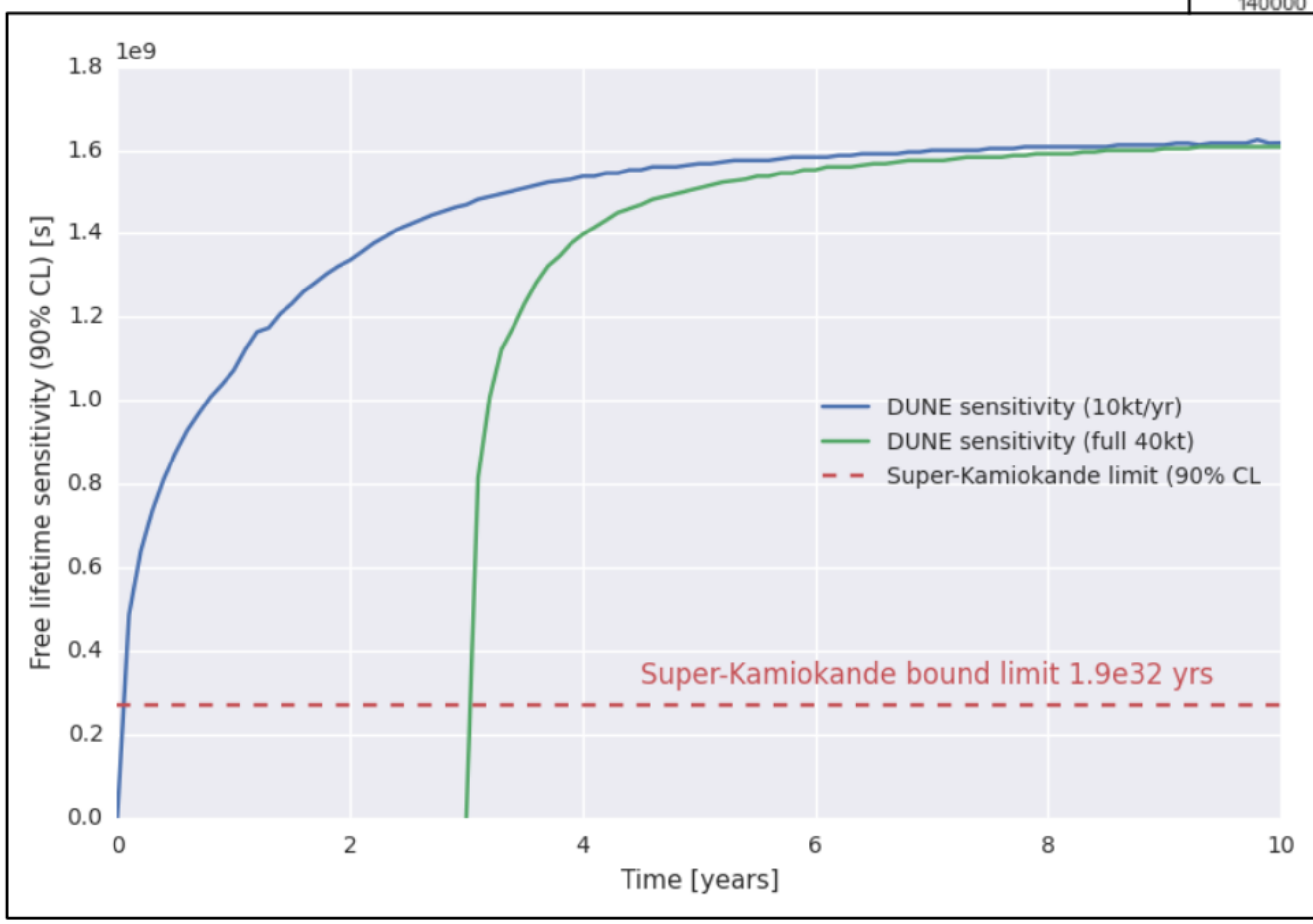
NEVIS LABORATORIES COLUMBIA UNIVERSITY

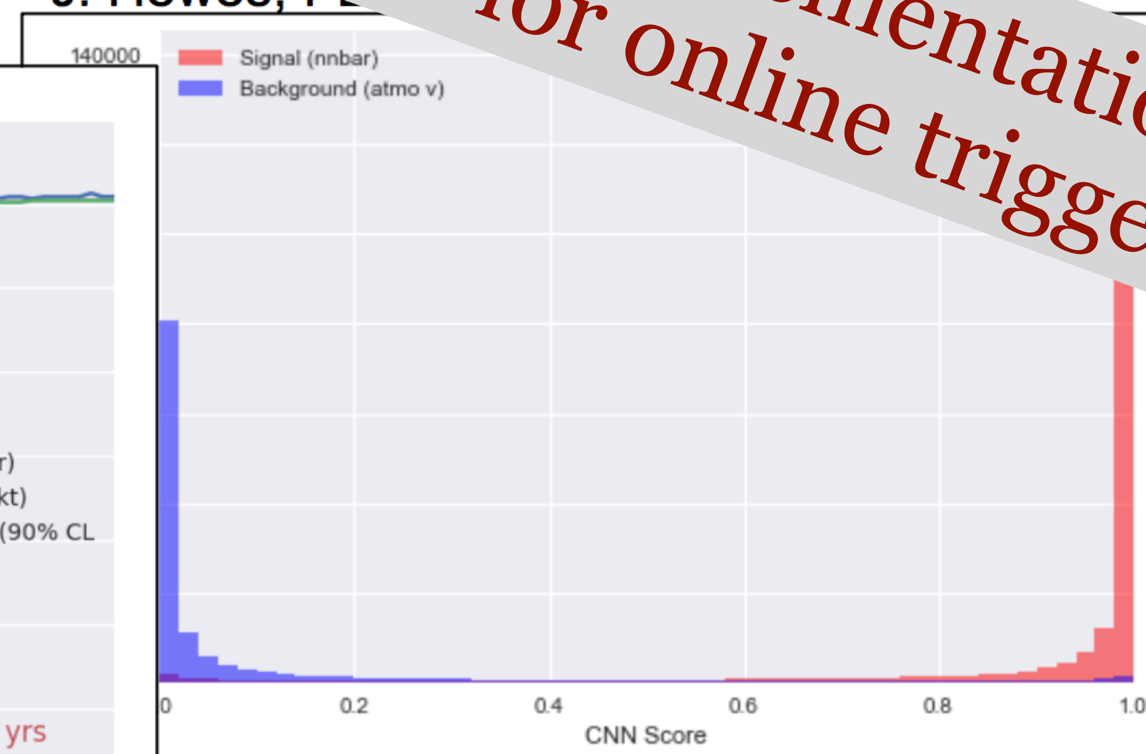
n-nbar Search in DUNE FD

CNN-based search for n-nbar in

FPGA Implementation R&D for online trigger

J. Hewes,





Resulting projected sensitivity of DUNE for given efficiency and mis-ID rate, as a function of run time.
Sensitivity shows 5x improvement over current Super-K limit.

MANCHESTER 1824
The University of Manchester

Nevis Laboratories
Columbia University

Distributed CNN Training at PNNL

E. Church, J. Daily, C. Siegel, M. Schram, J. Strube, K. Wierman

MaTEx

Pacific Northwest NATIONAL LABORATORY

Proudly Operated by Battelle Since 1965

- Full event image: **3600 wires x 3600 time bins x 3 planes x 4 Bytes**
  - MicroBooNE simulated single particle events
  - **~150 MB / event**
- Even a moderately small network only leaves room for a mini-batch size of 1-2 events on a modern GPU, for full event fidelity
  - This is smaller than required given the latent space of the CNN → slow development. Distributed scaling of compute resources will help significantly.
  - Scaling allows increase in network depth too (if required)
- For deep learning, one wants large training samples.
  - Training may become quickly I/O bound and hence prohibitively slow
  - Even a dedicated ”large-mem” node cannot fit more than a few thousand samples into memory, at best.

→ We are studying PNNL’s MaTEx for distributed training

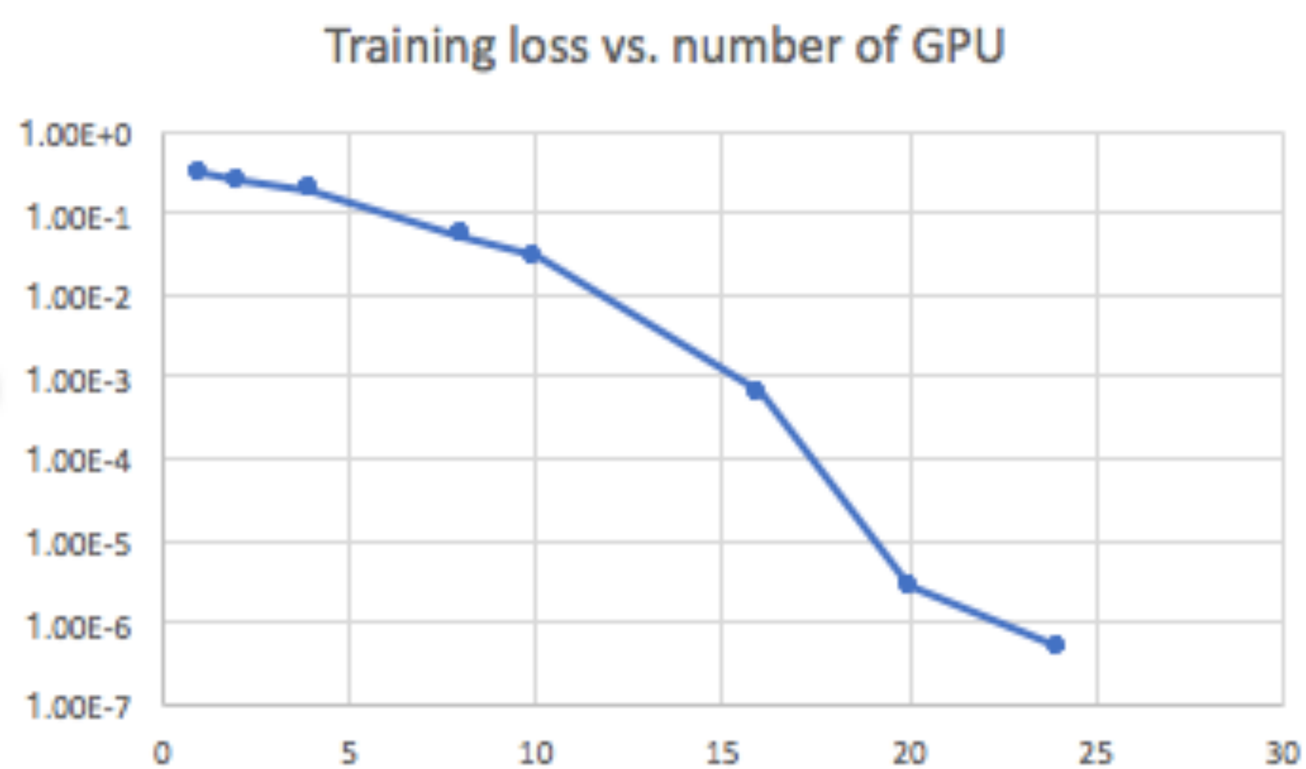
Easier to ”drop in” than say the uber solution, and locally supported!

→ And using in-memory loss-less image compression

Slide 1/2 from E. Church

Current status (preliminary)

Training time: mini-batch size = 2, 10000 steps per GPU … 10 epochs
Identical networks, loss functions, optimizers and input data
➞ MaTEx does not currently introduce noticeable overhead at this scale



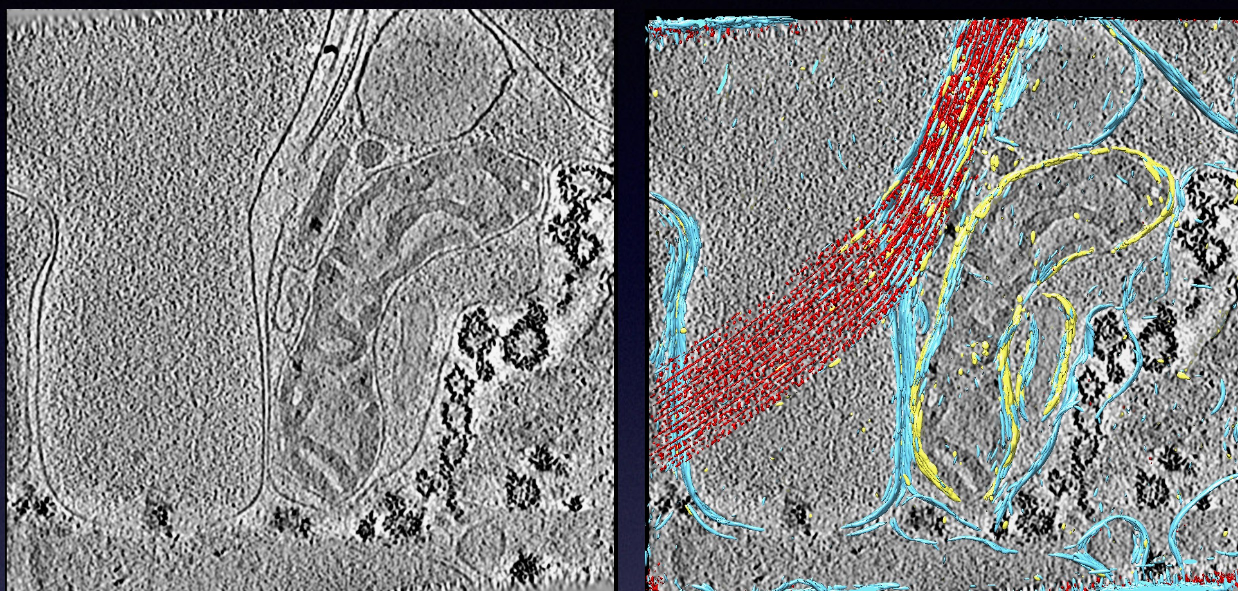
For the same wall time, training improves with number of GPUs
➞ Studies ongoing, significant updates planned for CHEP2018

Slide 2/2 from E. Church

DL Software @PNNL
Framework Development

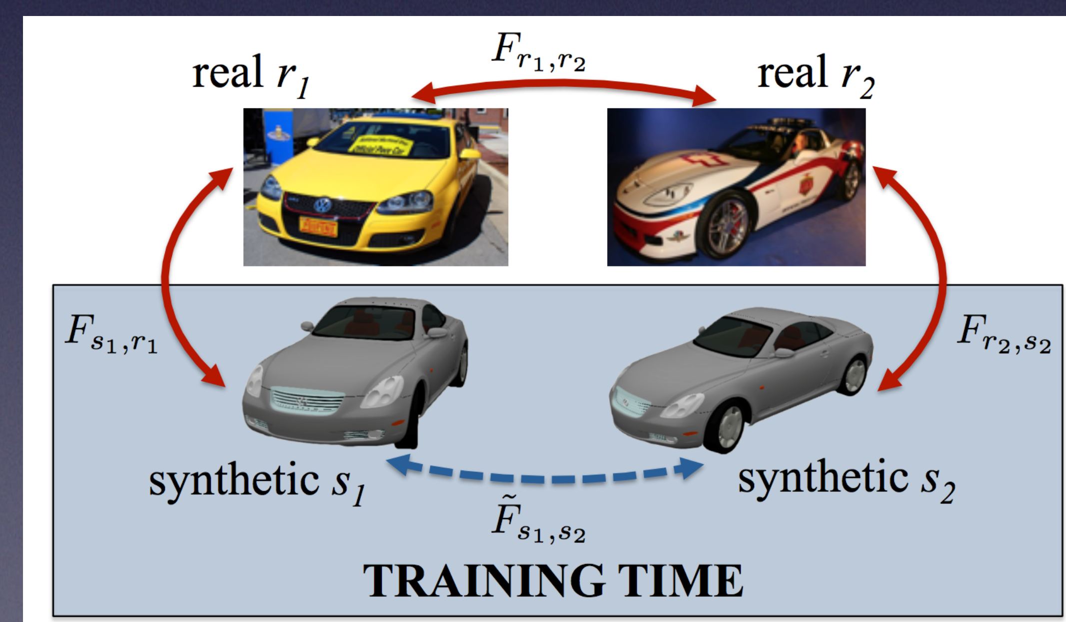
More Exciting Stuffs … come chat w/ me :)

3D voxel labeling of Cryo-EM image
(below: mitochondrion detection)



Multi-network Training
Techniques R&D

Detection + Clustering (Mask R-CNN)
of ATLAS jet images
(w/ SLAC ATLAS group)



Pixel-Flow network for 3D track reco
(via cross-plane pixel correlation)

... wrapping up ...

Outline

1. Introduction
2. Neutrino oscillation experiments
3. MicroBooNE: first large-scale LArTPC in U.S.
4. DeepLearning for LArTPC image analysis
5. **Summary**

Wrap Up

- Very active **DNN techniques R&D** for LArTPC
  - MicroBooNE/SBND/ICARUS/DUNE (ND+FD)

- **DNN for data reconstruction**
  - Modular algorithms for a full reconstruction chain
  - In-depth computer-vision application development using deep neural networks

- **Cross-disciplinary effort** including non-HEP
  - Energy frontier (LHC), cosmic frontier (LSST/EXO/NEXT), photon science (LCLS), Cryo-EM, QIS…
  - **DeepLearnPhysics** group for sharing tools/expertise

Back-Up
Slides

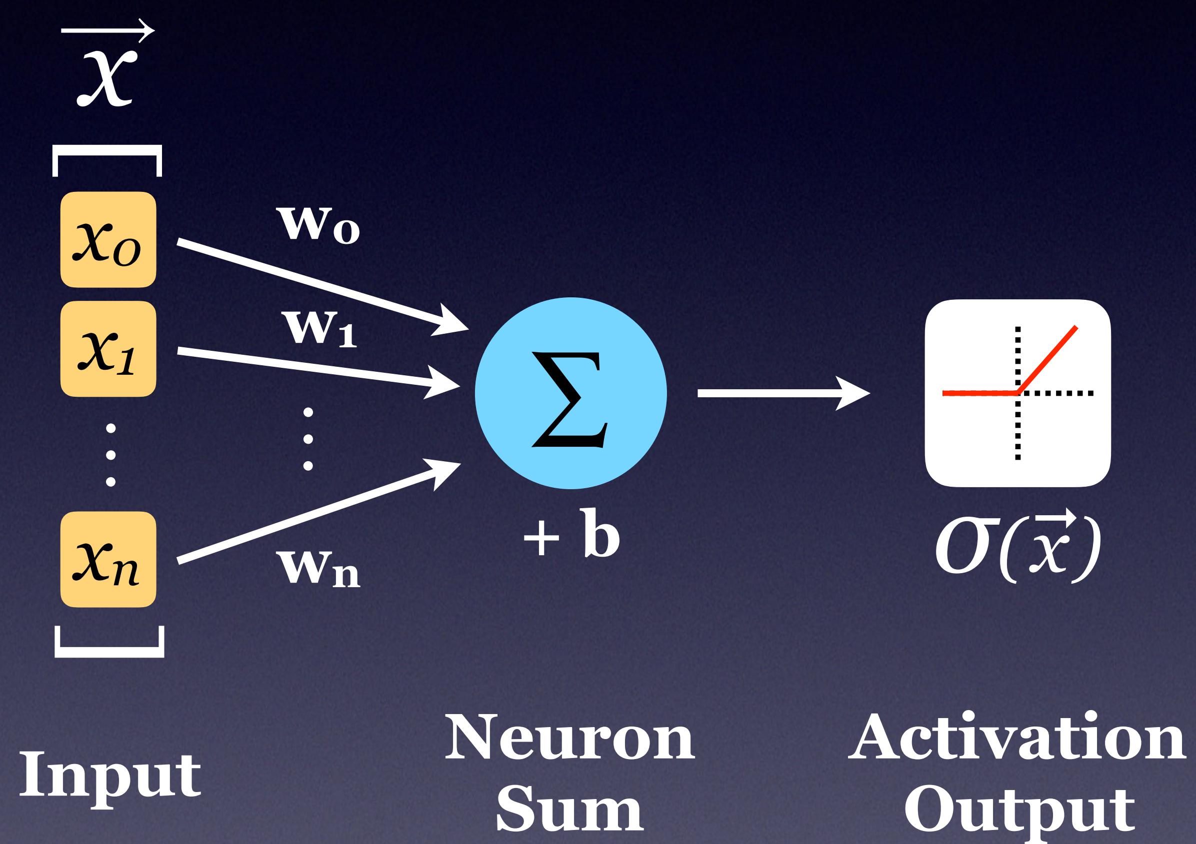
NN & CNN
Basics
~ How Does It Work? ~

How a Simple Perceptron Works

Background: Neural Net

The basic unit of a neural net is the ***perceptron*** (loosely based on a real neuron)

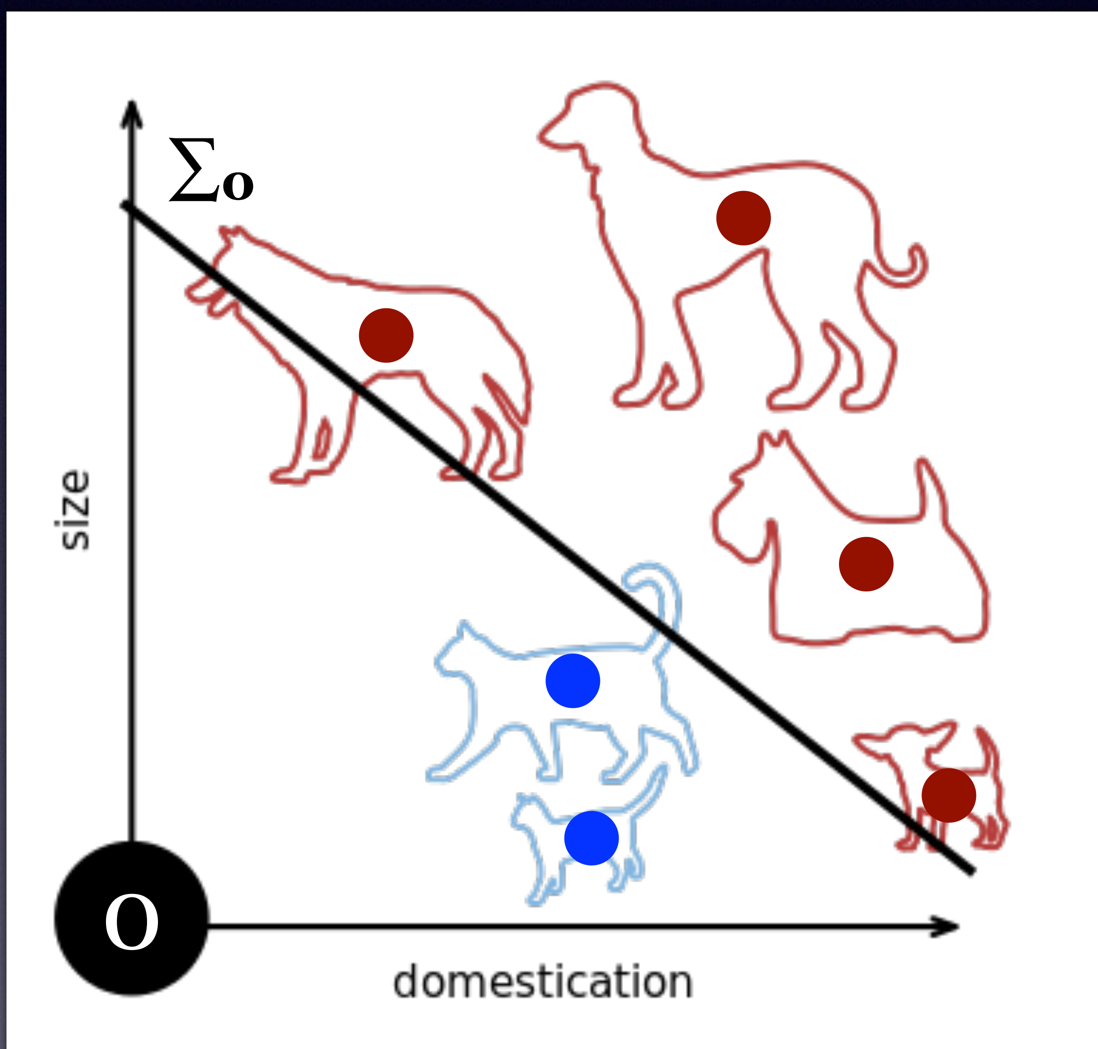
Takes in a vector of inputs (*x*). Commonly inputs are summed with weights (*w*) and offset (*b*) then run through activation.



$$\sigma(\vec{x}) = \begin{cases} \vec{w}_i \cdot \vec{x} + b_i & \vec{w}_i \cdot \vec{x} + b_i \geq 0 \\ 0 & \vec{w}_i \cdot \vec{x} + b_i < 0. \end{cases}$$

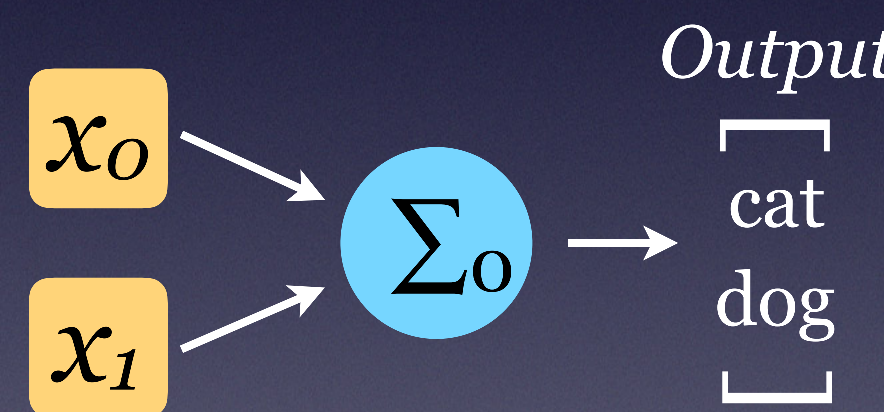
How a Simple Perceptron Works
Perceptron 2D Classification

Imagine using two features to separate cats and dogs



from [wikipedia]()

$$\sigma(\vec{x}) = \begin{cases} \vec{w}_i \cdot \vec{x} + b_i & \vec{w}_i \cdot \vec{x} + b_i \geq 0 \\ 0 & \vec{w}_i \cdot \vec{x} + b_i < 0. \end{cases}$$

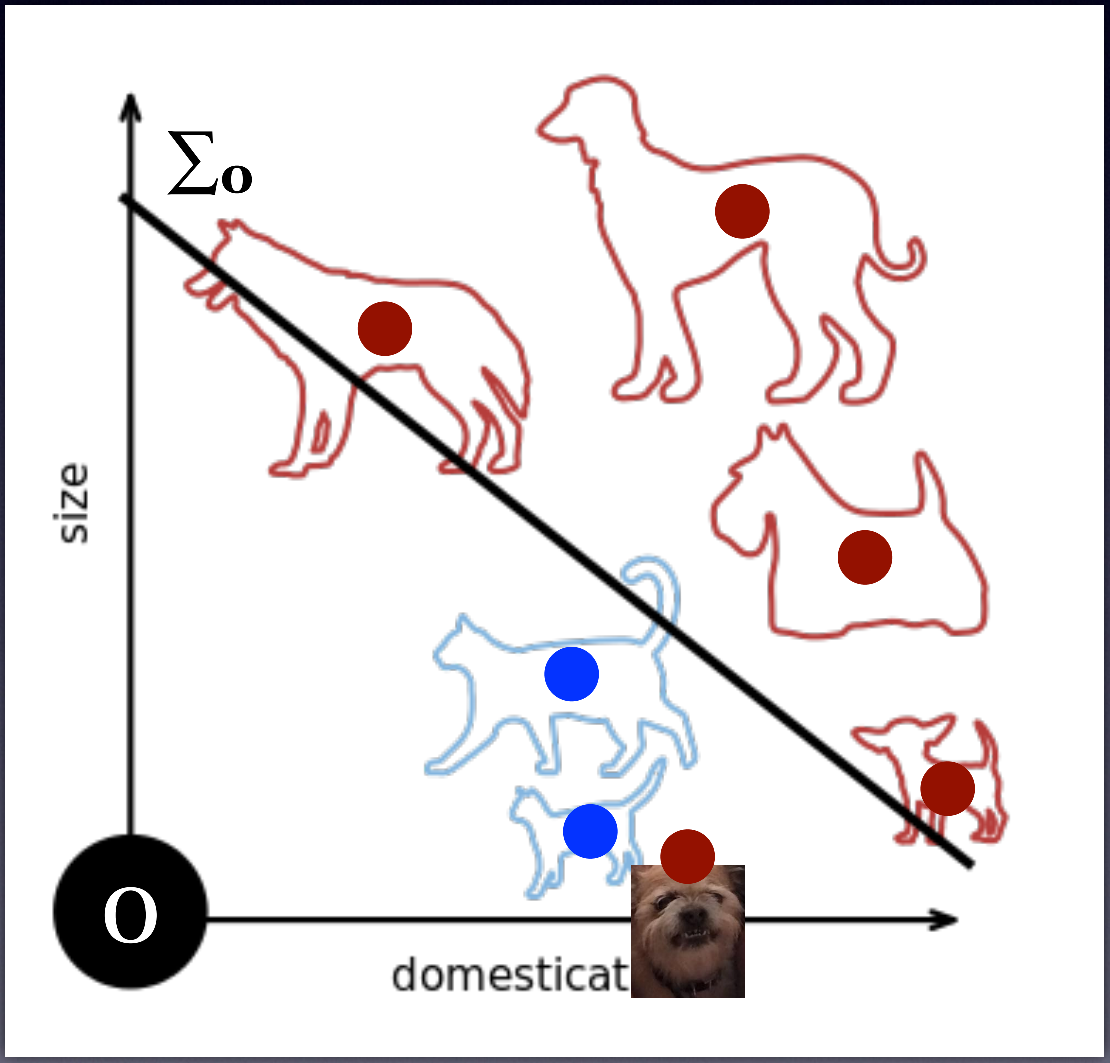


By picking a value for w and b, we define a boundary between the two sets of data

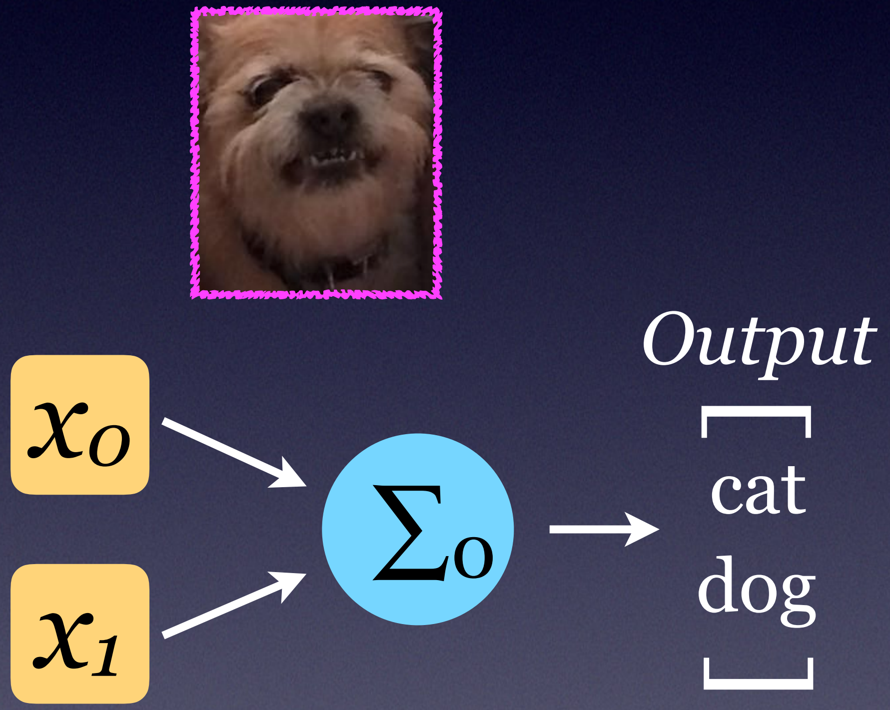
How a Simple Perceptron Works

Perceptron 2D Classification

Maybe we need to do better: assume a new data point (small but not as well behaved)

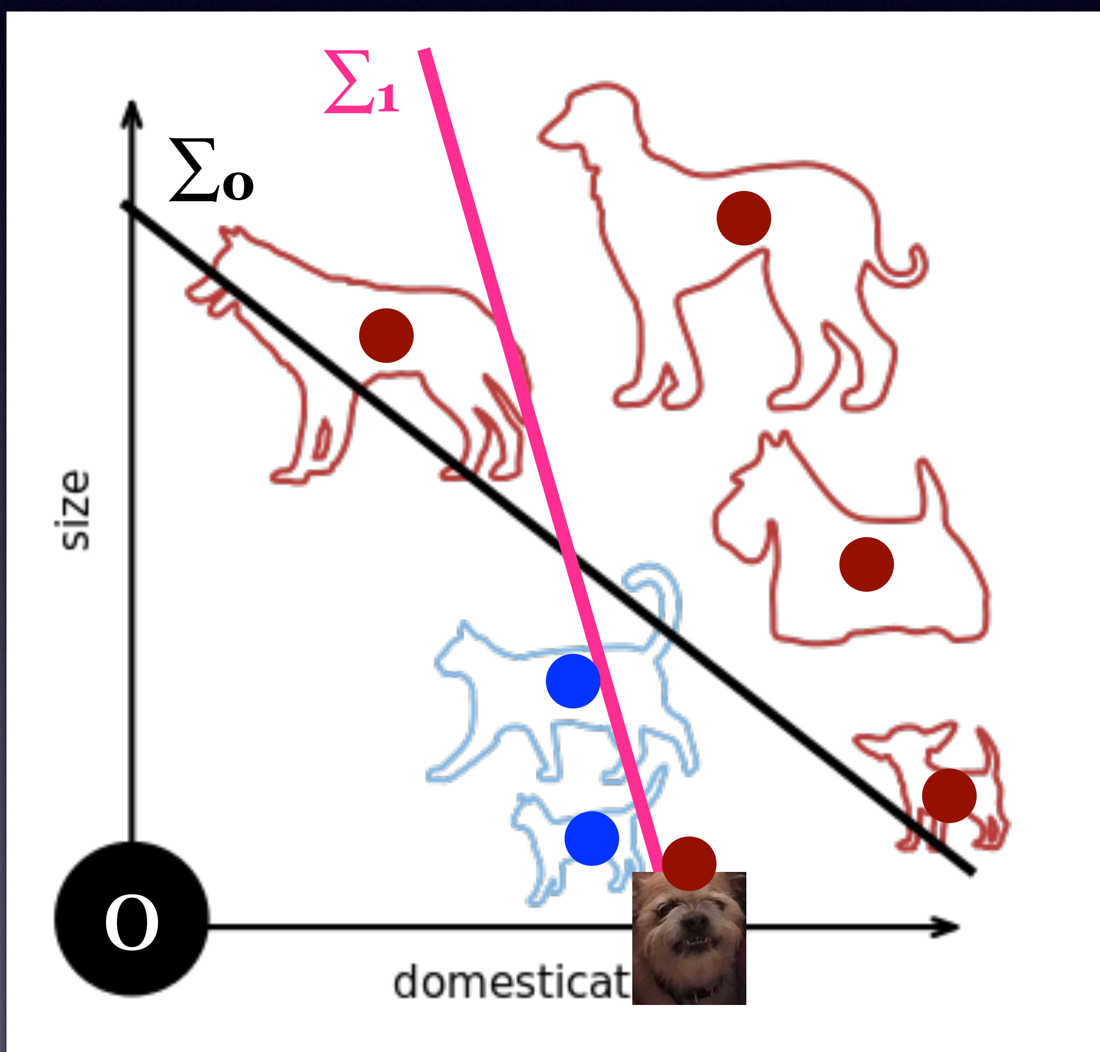


from wikipedia

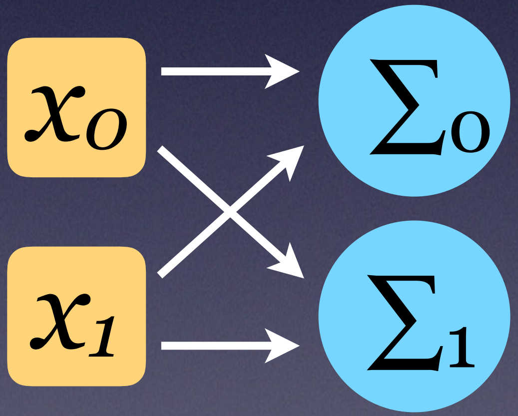


How a Simple Perceptron Works
Perceptron 2D Classification

Maybe we need to do better: assume a new data point (small but not as well behaved)



from wikipedia

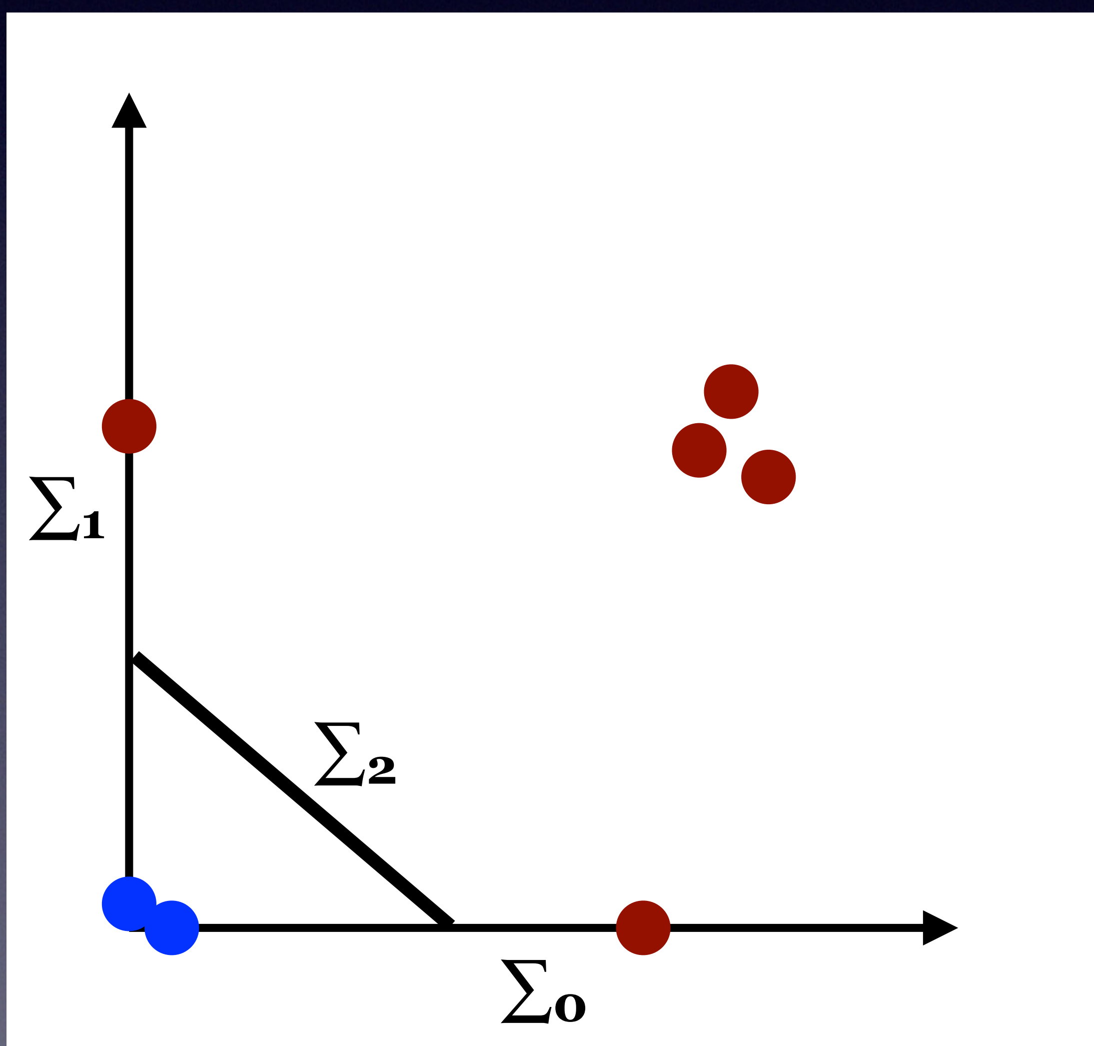


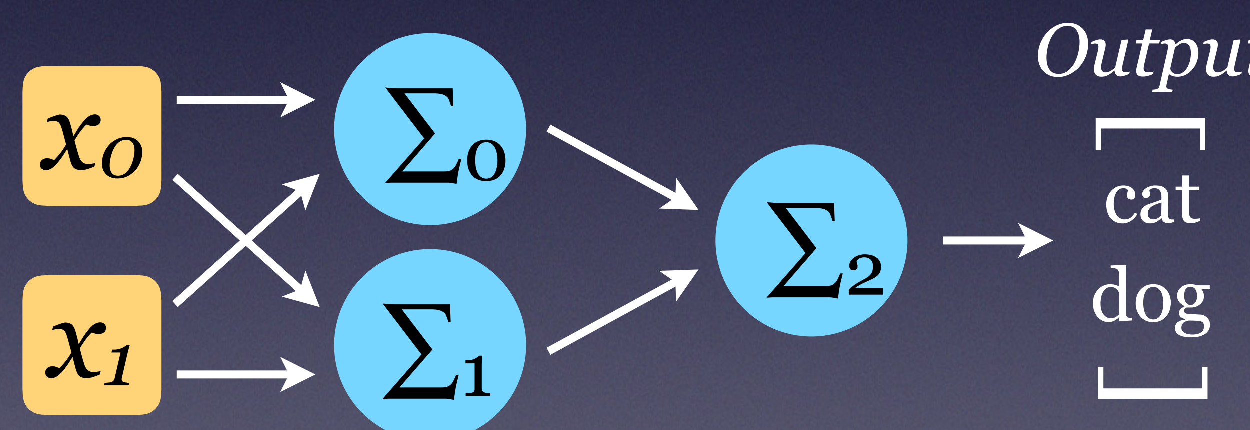
We can add another perceptron to help (but does not yet solve the problem)

How a Simple Perceptron Works

Perceptron 2D Classification

Maybe we need to do better: assume a new data point (small but not as well behaved)

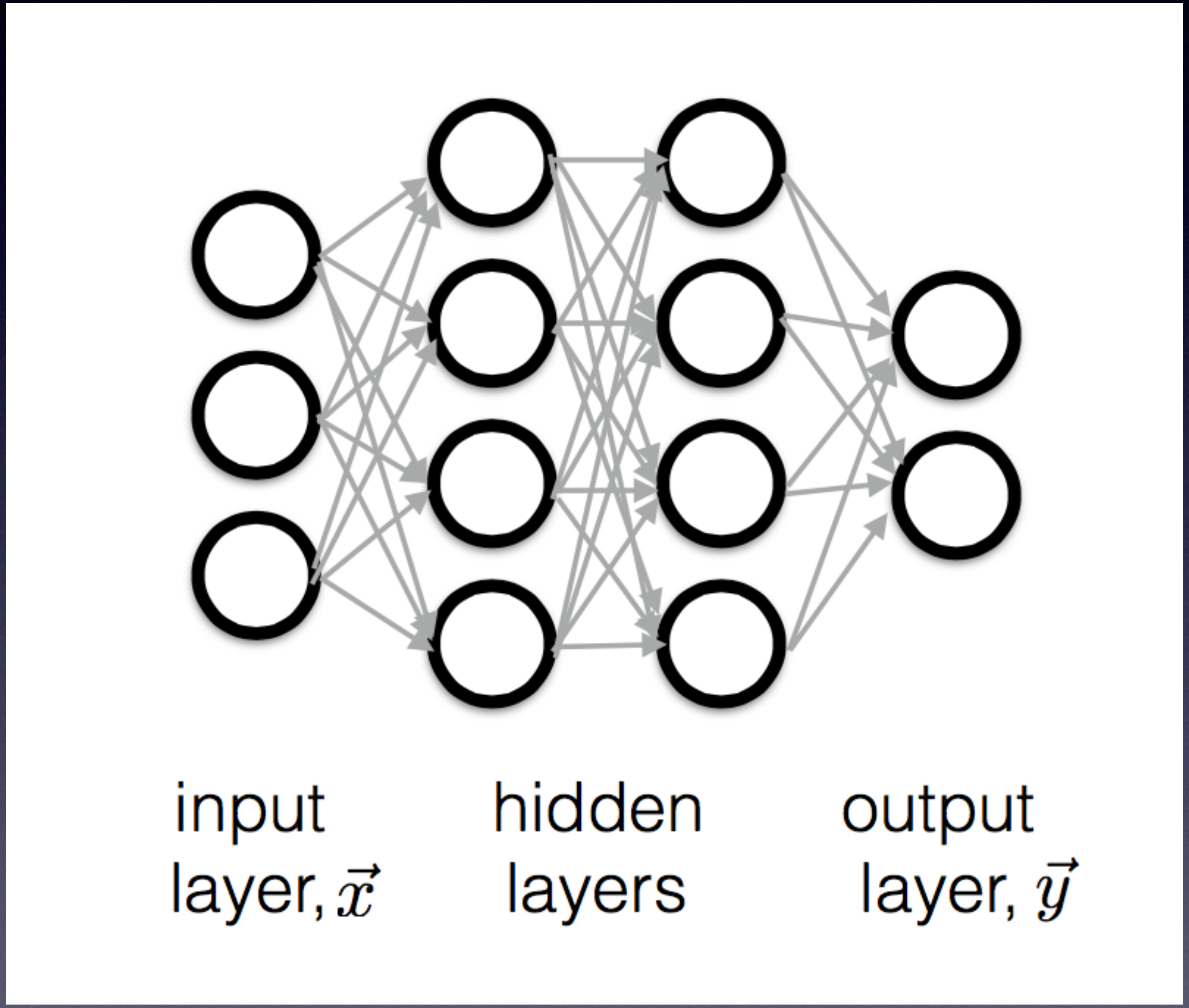




Another layer can classify based on preceding feature layer output

“Classical” Neural Net

Fully-Connected, Feed-forward, Multi-Layer Perceptrons



A traditional neural network consists of a stack of layers of such neurons where each neuron is ***fully connected*** to other neurons of the neighbor layers

"Classical" Neural Net

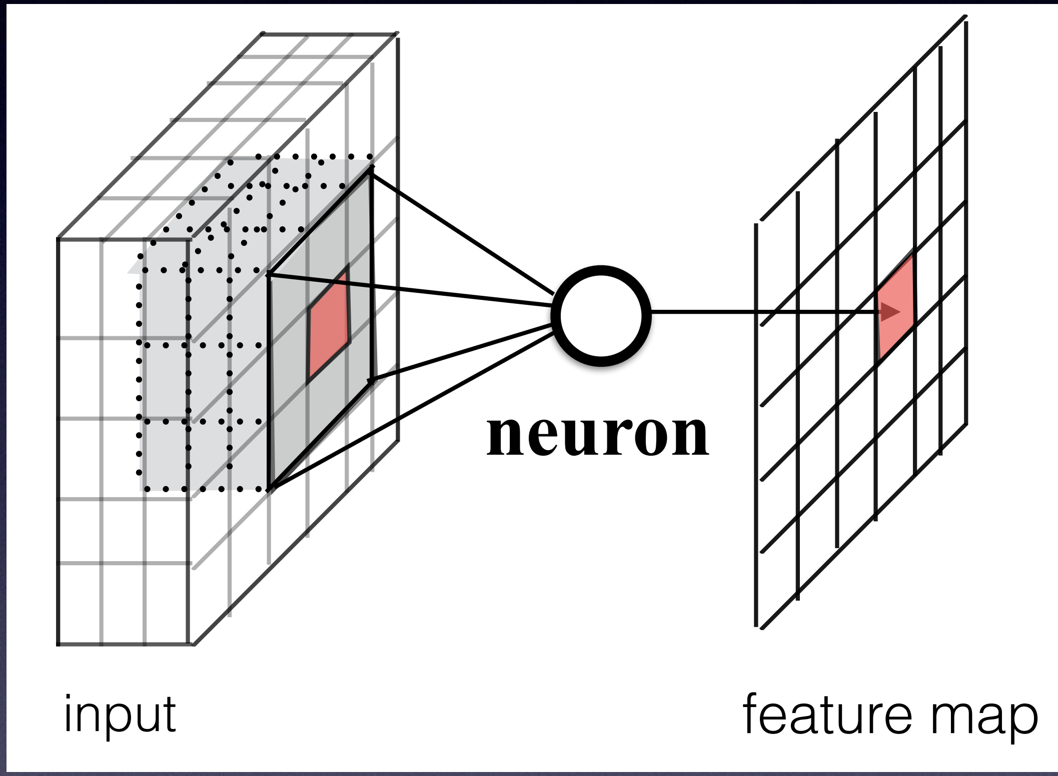
... is not ideal for image classification ...

Image classification

- **What is input neurons?**
  - Every pixel value
- **How many weights?**
  - # of pixels in an image!
- **Fully connected?**
  - translation variant!

Convolutional Neural Networks

CNN introduce a ***limitation*** by forcing the network to look at only local, translation invariant features

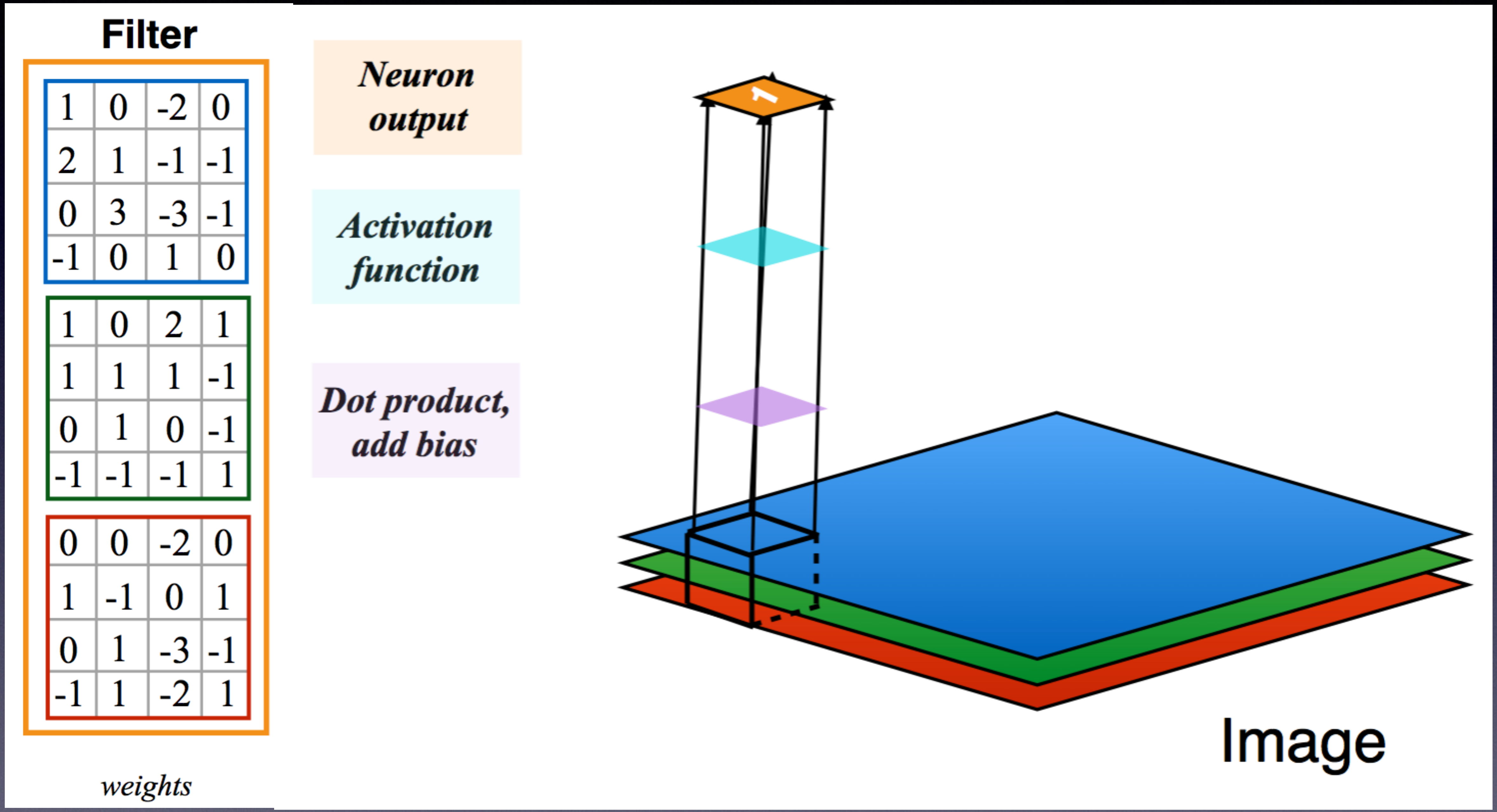


$$f_{i,j}(X) = \sigma\left(W_i \cdot X_j + b_i\right),$$

Activation of a neuron depends on the element-wise product of 3D weight tensor with 3D input data and a bias term

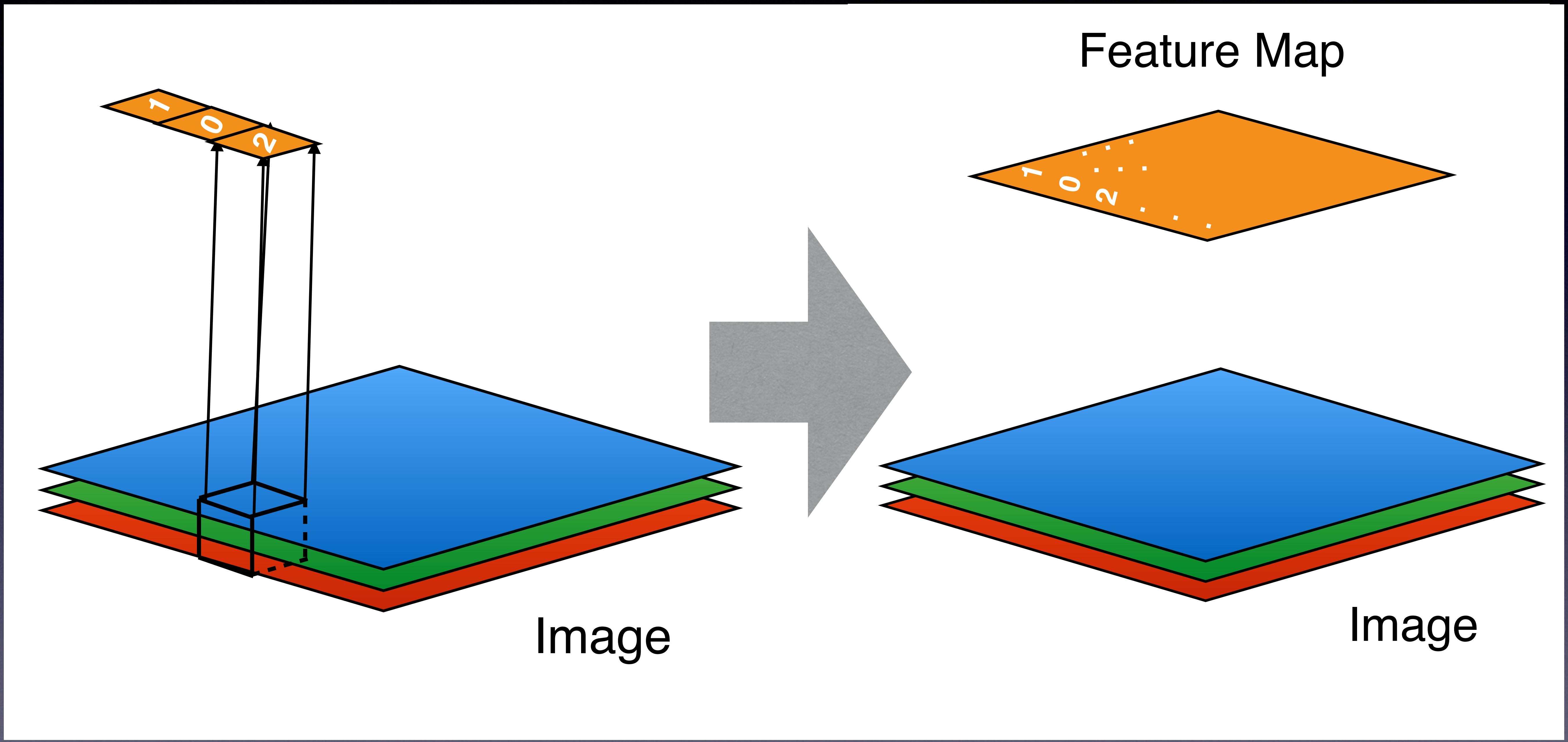
- Translate over 2D space to process the whole input
- Neuron **learns translation-invariant features**
  - Suited for a "***homogeneous***" detector like LArTPC
- **Output**: a "feature-enhanced" image (***feature map***)

Convolutional Neural Networks



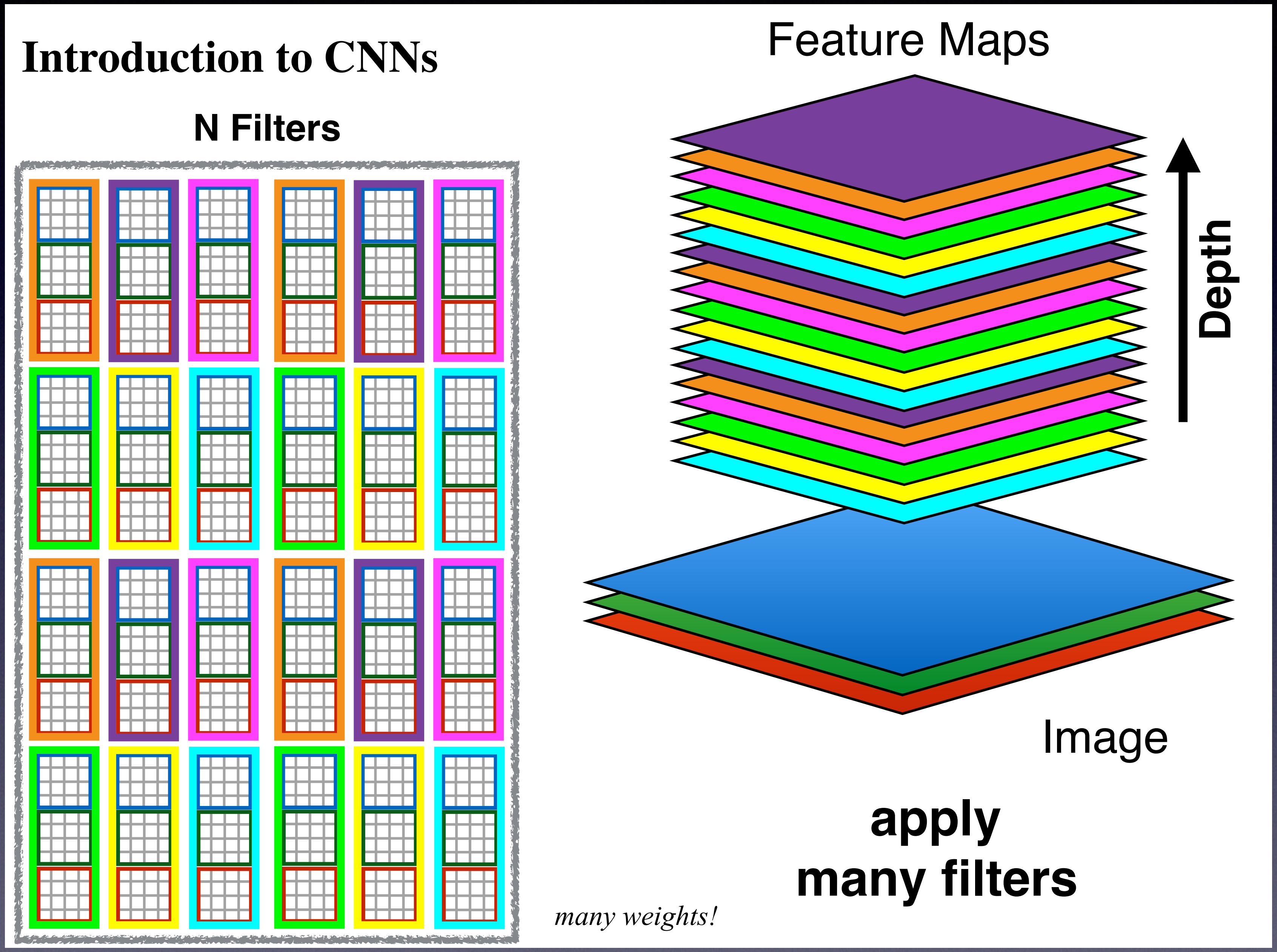
Toy visualization of the CNN operation

Convolutional Neural Networks



Toy visualization of the CNN operation

Convolutional Neural Networks

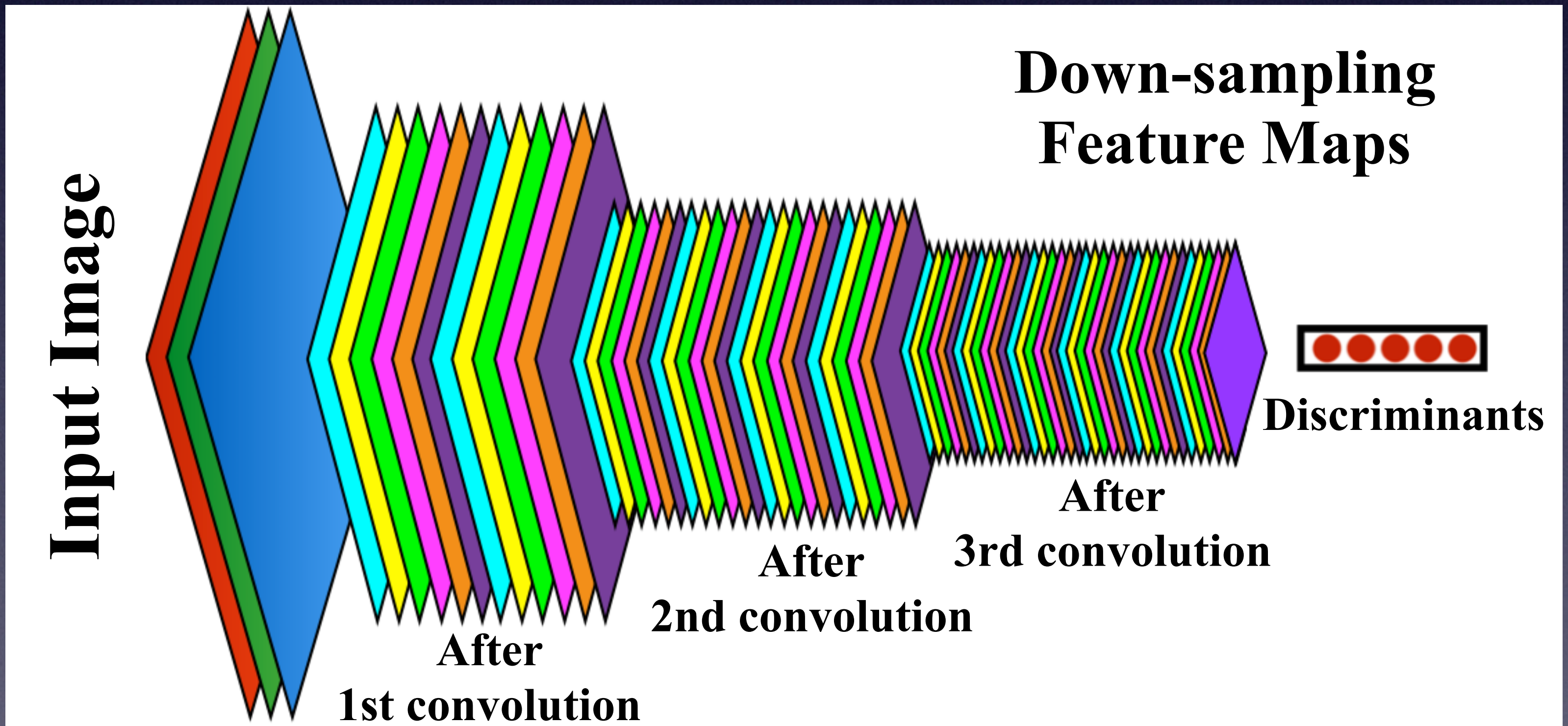


Toy visualization of the CNN operation

How Image Classification Networks Work

Goal: extract features to give "single label" to an image

1. **Convolution operation**
2. **Down-sampling**

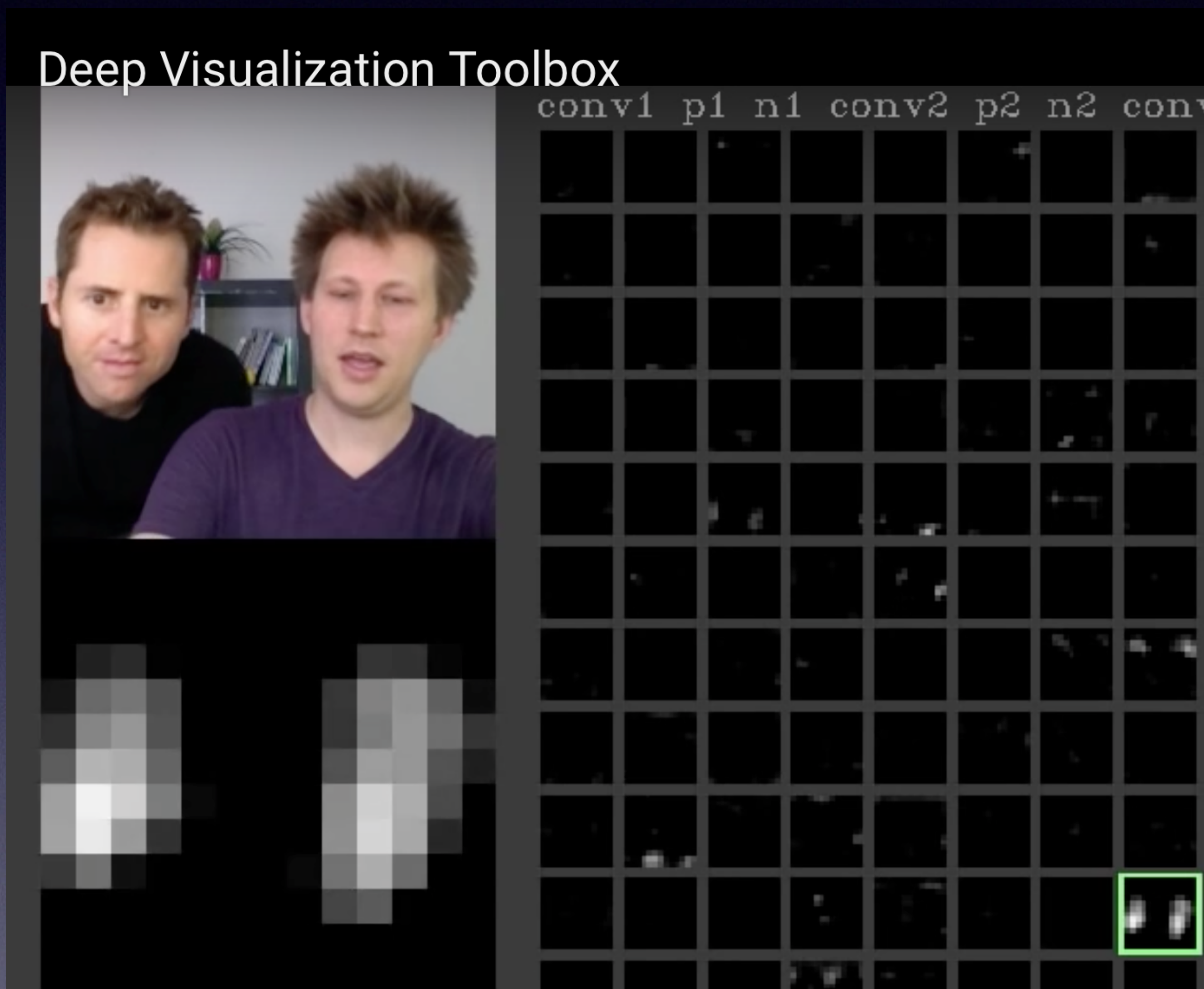


Series of convolutions
+ down-sampling

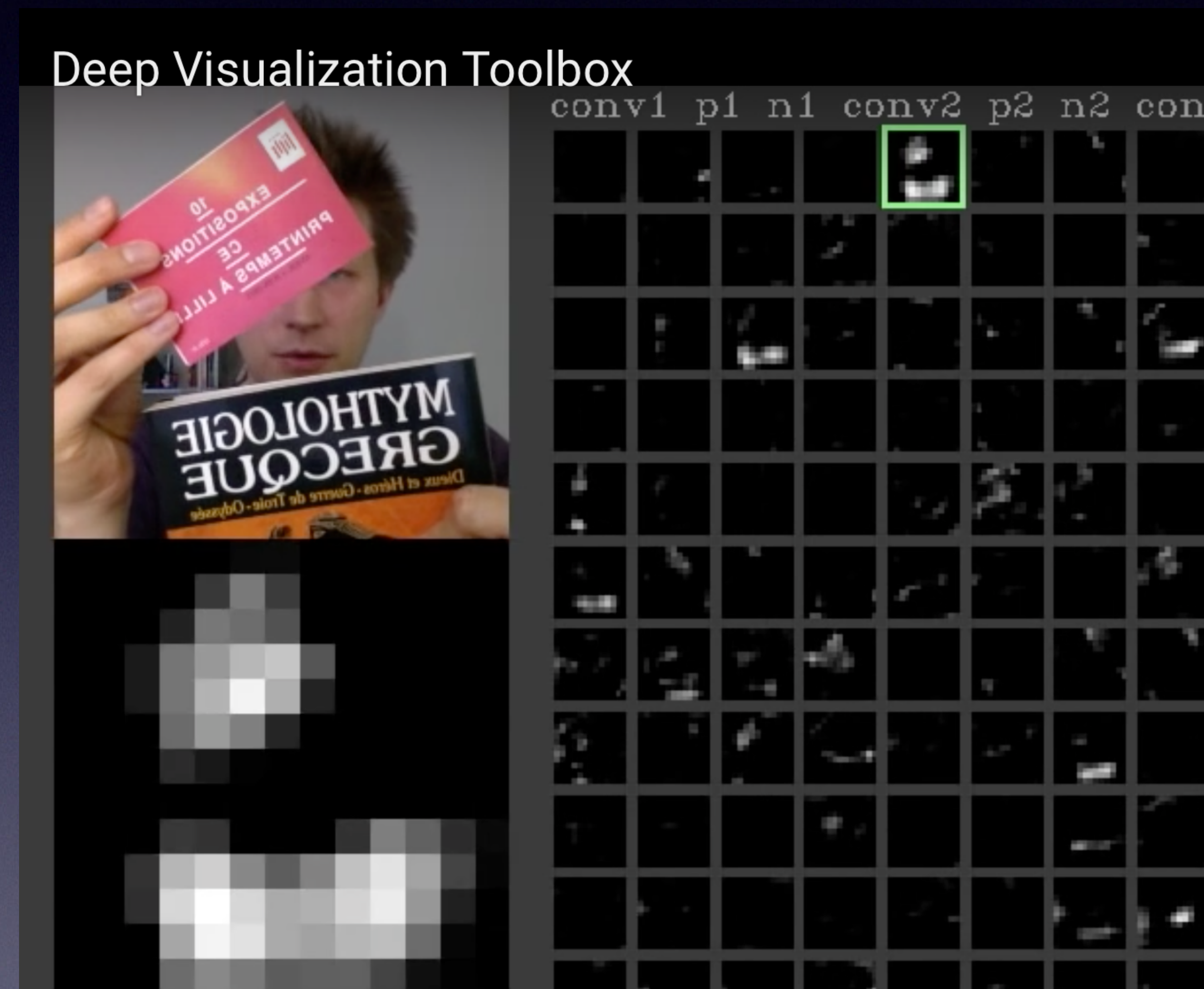
How Image Classification Networks Work

Feature map visualization example

- https://www.youtube.com/watch?v=AgkfIQ4IGaM



Neuron concerning face

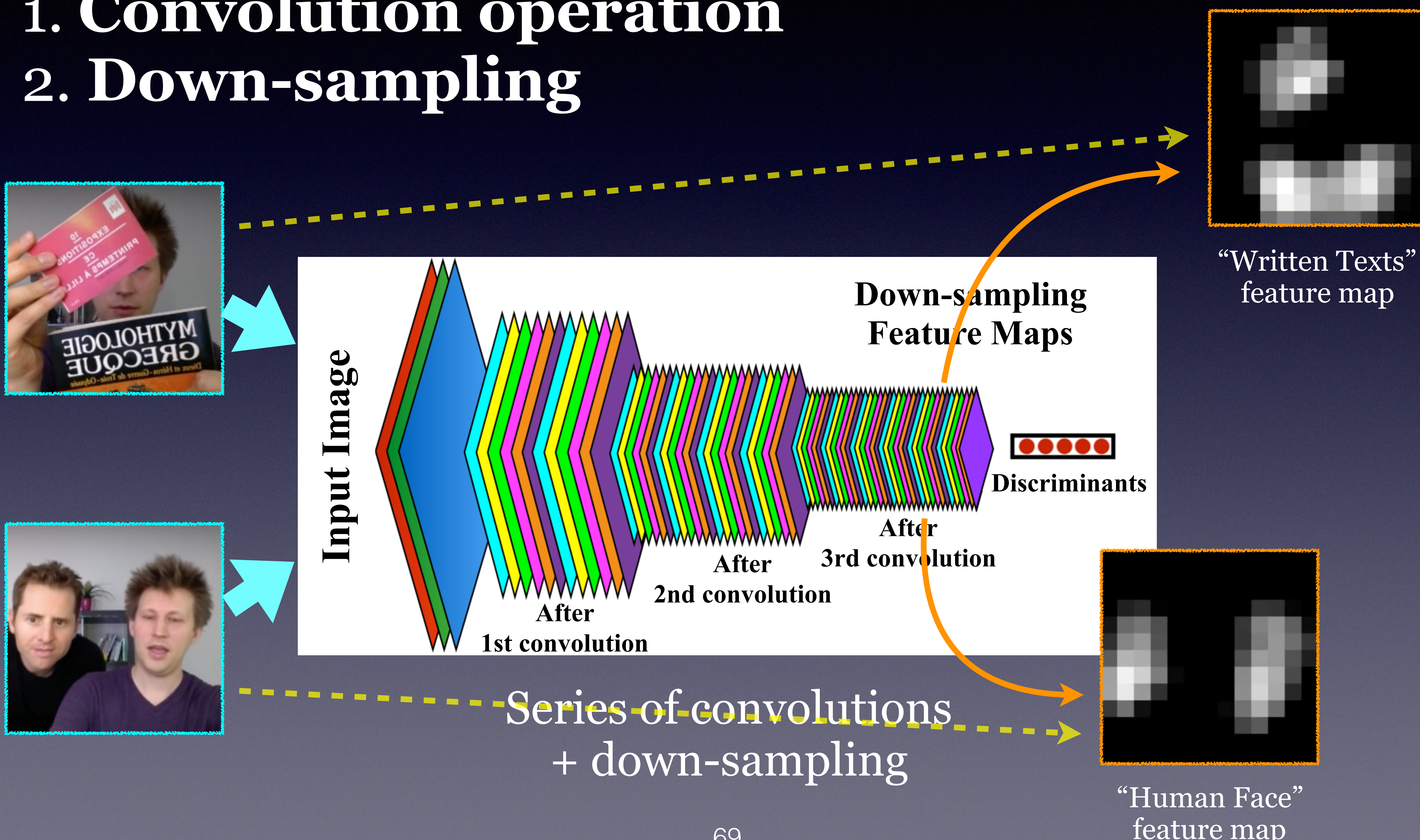


Neuron loving texts
(and don't care about your face)

How Image Classification Networks Work

Goal: extract features to give “single label” to an image

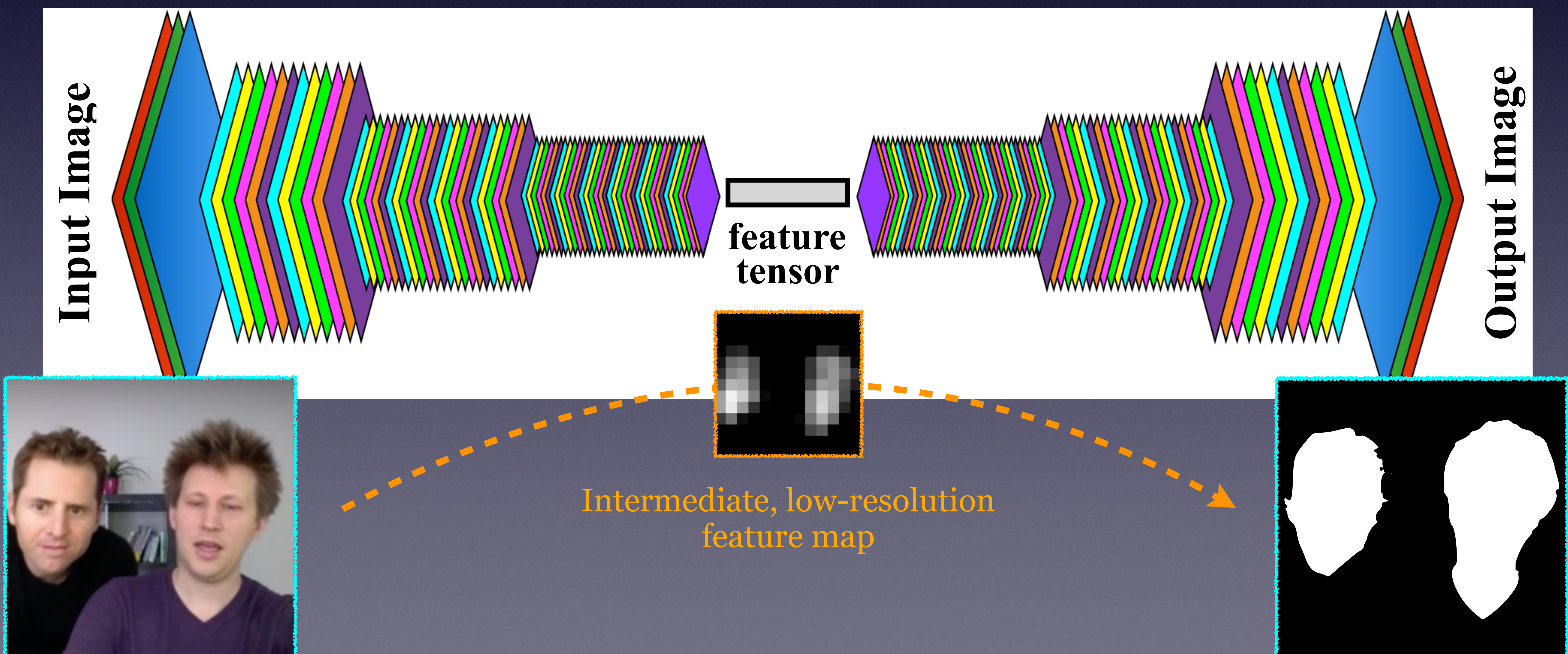
1. **Convolution operation**
2. **Down-sampling**



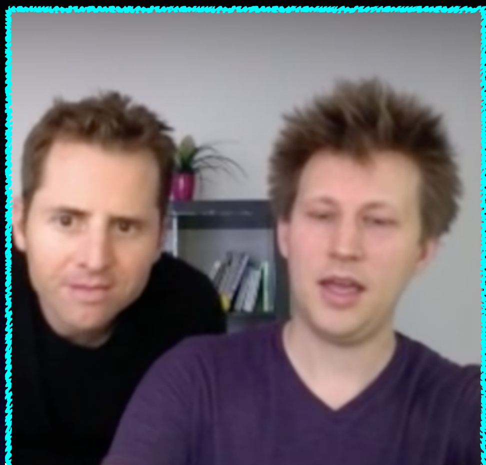
How SSNet Works

Goal: recover precise, pixel-level location of objects

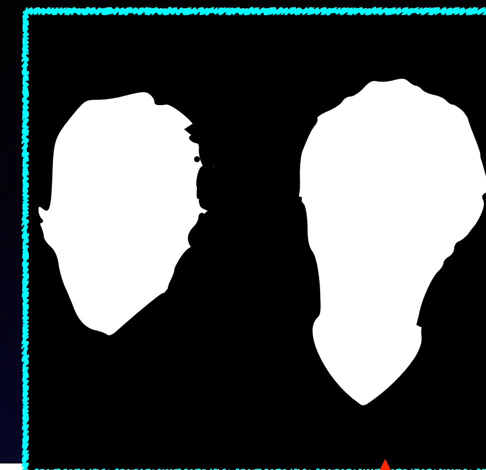
1. **Up-sampling**
   - Expand spatial dimensions of feature maps
2. **Convolution**
   - Smoothing (interpolation) of up-sampled feature maps

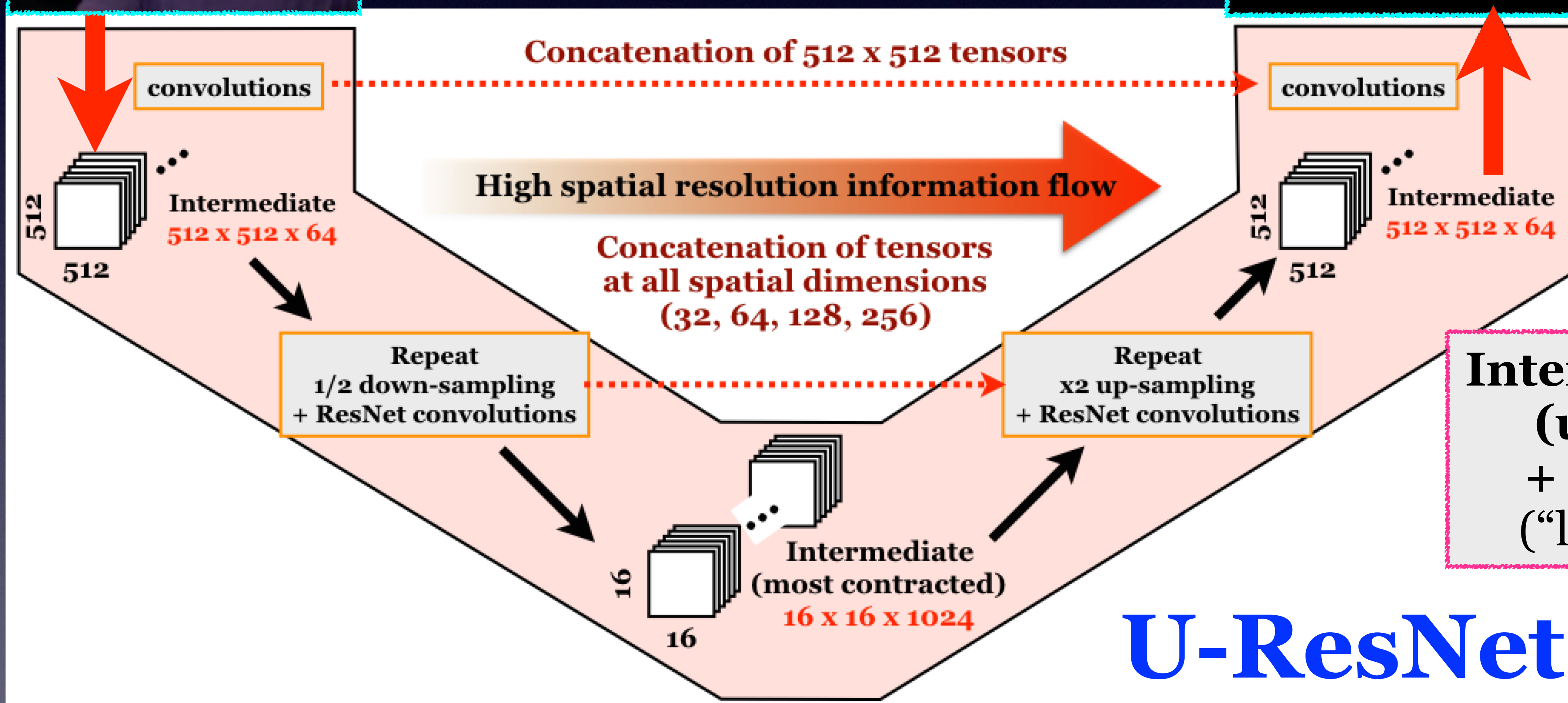


DNN for LArTPC Data Reconstruction

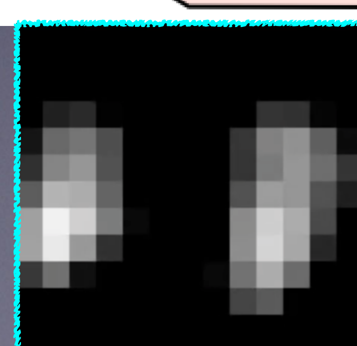


How does U-ResNet Work?





Interpolation filters
(up-sampling)
+ Convolutions
("learnable" filter)



Down sampling + Convolutions to identify
highly abstract features (e.g. "human face")