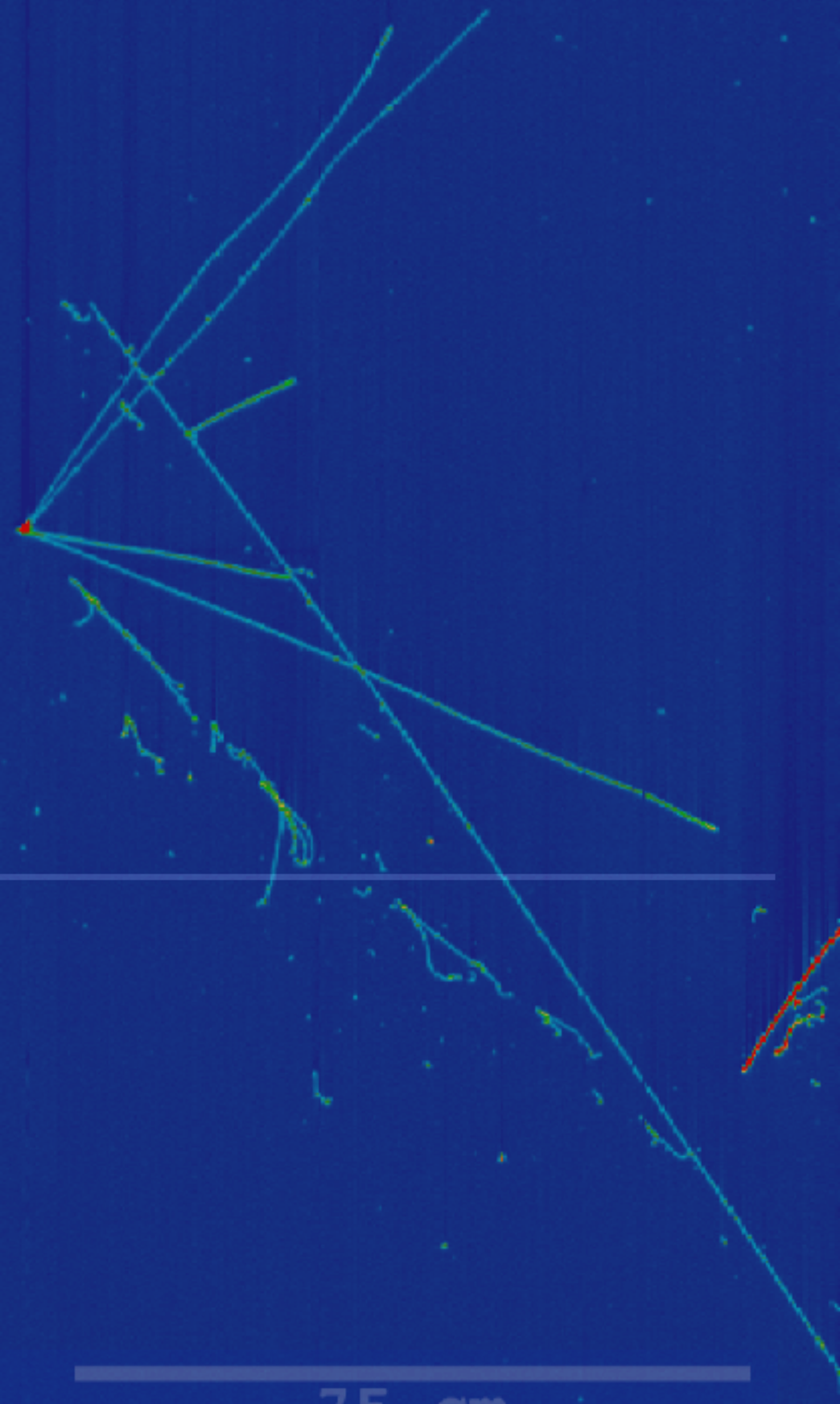
μBooNE

Run 3493 Event 41075, Oct. 23rd, 2015

FINDING NEUTRINOS IN LARTPCS USING CONVOLUTIONAL NEURAL NETWORKS

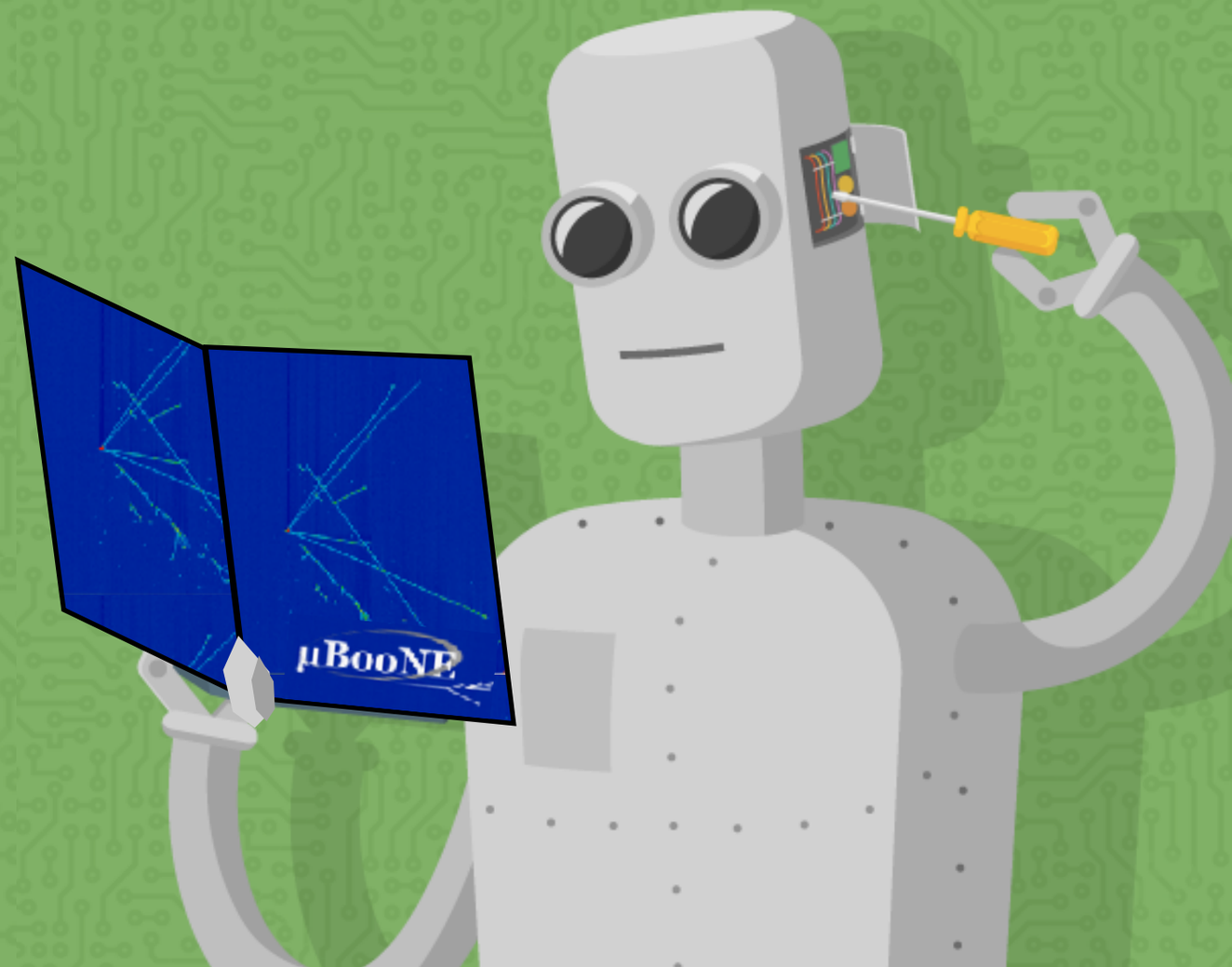
Taritree Wongjirad
Tufts University

DNP 2017



Outline

- Convolutional neural networks (CNNs) are a type of deep, feed-forward neural network that have been successfully applied to a wide range of problems
- Discuss the ways MicroBooNE - a LArTPC detector - has been exploring the use of CNNs
- Three applications
  - Classification
  - Object detection
  - Semantic Segmentation



MICROBOONE (A LARTPC) GOALS

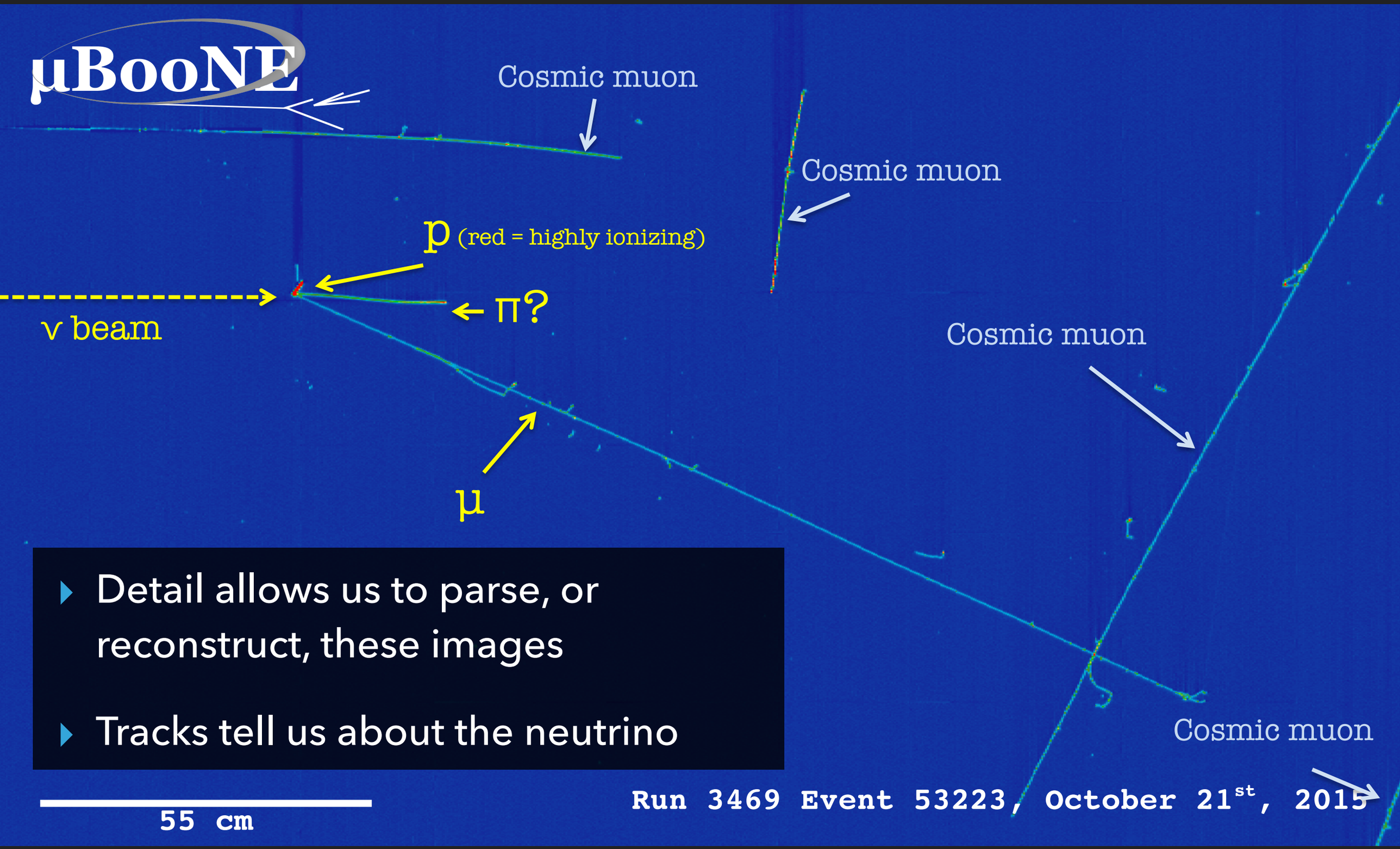
The detector during construction

- MicroBooNE, a LArTPC detector filled with 170 tons of LAr
- Looking for numu to nue oscillations
- Measure neutrino and argon cross sections
- Perform LArTPC R&D

MICROBOONE EVENT

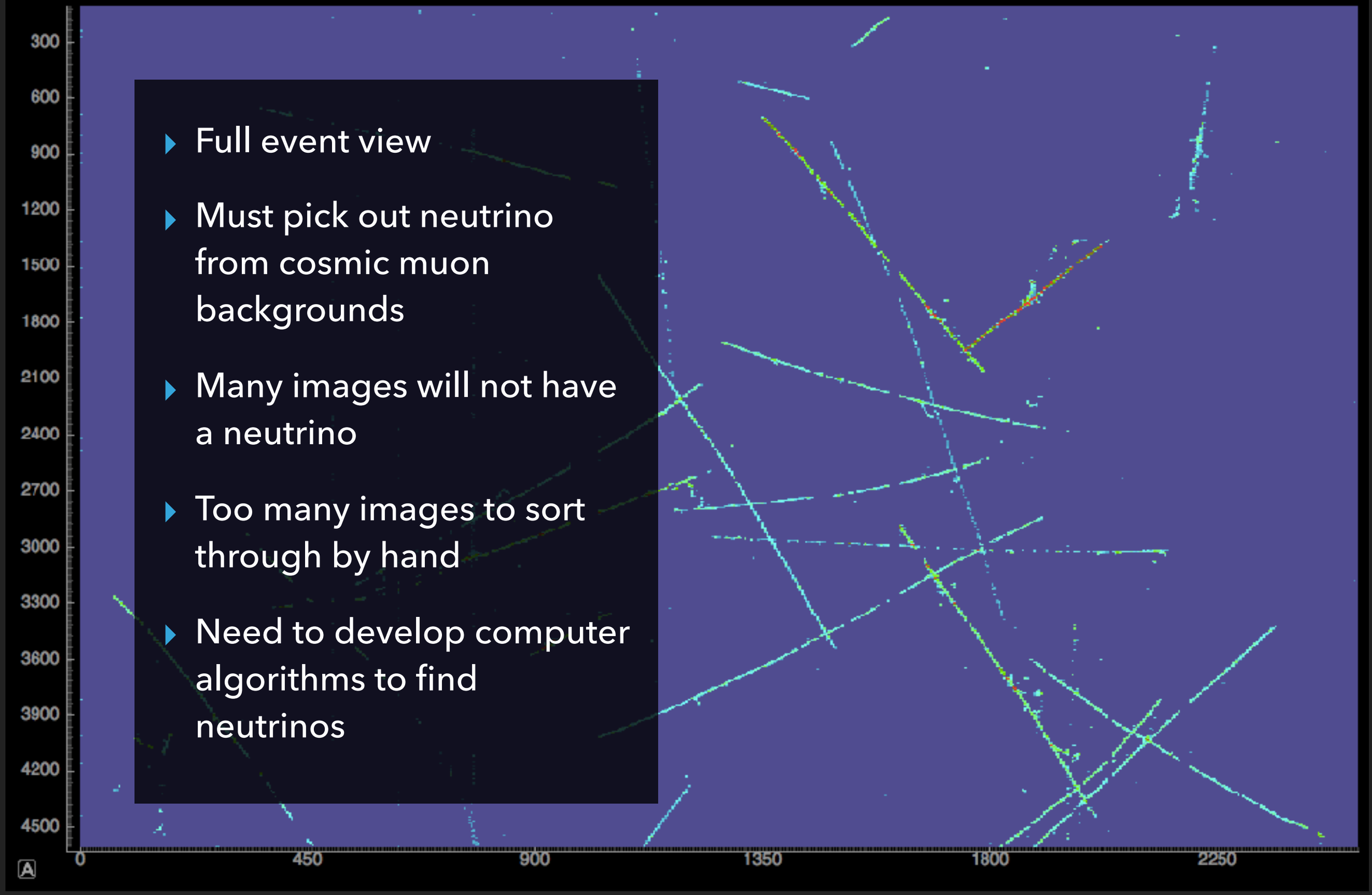
- Example neutrino event from the beam
- Lots of detail on location and amount of charge created in detector

RECONSTRUCTION



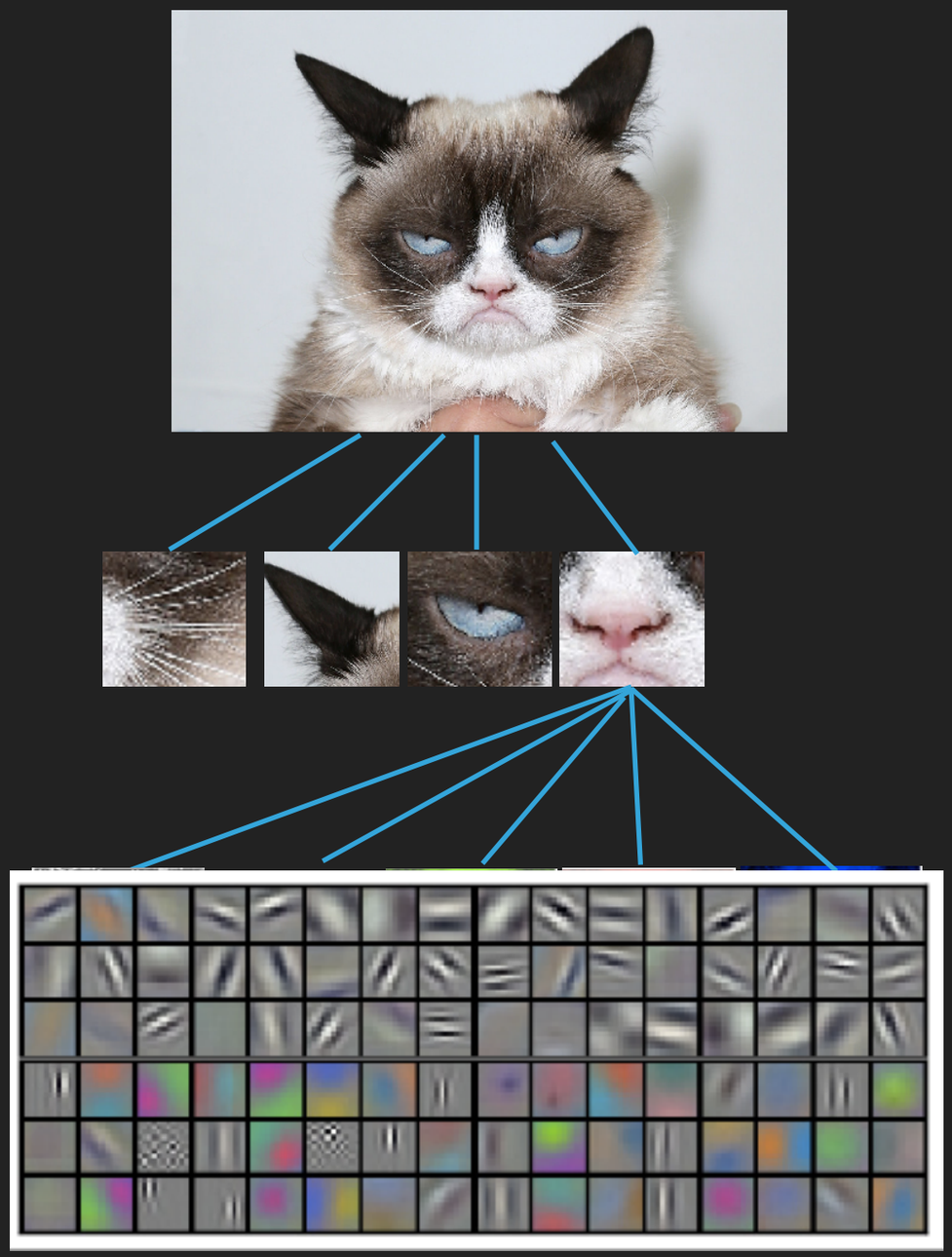
- Detail allows us to parse, or reconstruct, these images
- Tracks tell us about the neutrino

CHALLENGES



- Full event view
- Must pick out neutrino from cosmic muon backgrounds
- Many images will not have a neutrino
- Too many images to sort through by hand
- Need to develop computer algorithms to find neutrinos

IMAGE ANALYSIS

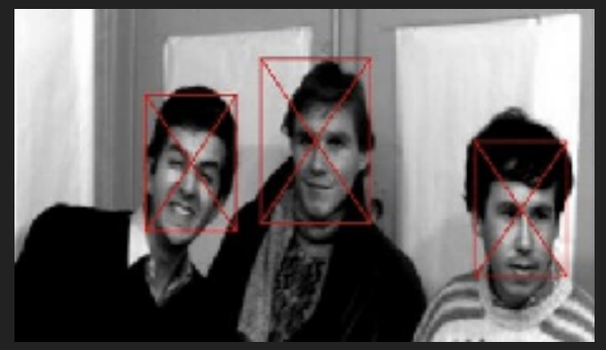


- To analyze an image, e.g. recognize as cat, decompose an object into a collection of small features
- Features composed of different patterns, lines and colors
- The problem
  - What features to define?
  - How to put them together?

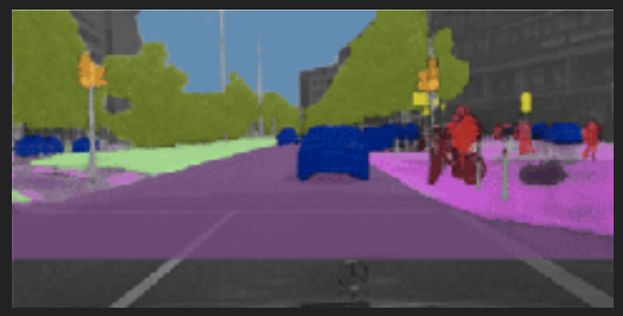
CONVOLUTIONAL NEURAL NETWORKS

- Applying convolutional neural nets (CNN)
- Very adept at image analysis
- Primary advantages: scalable and generalizable technique
- Successfully applied to many different types of problems

Face detection



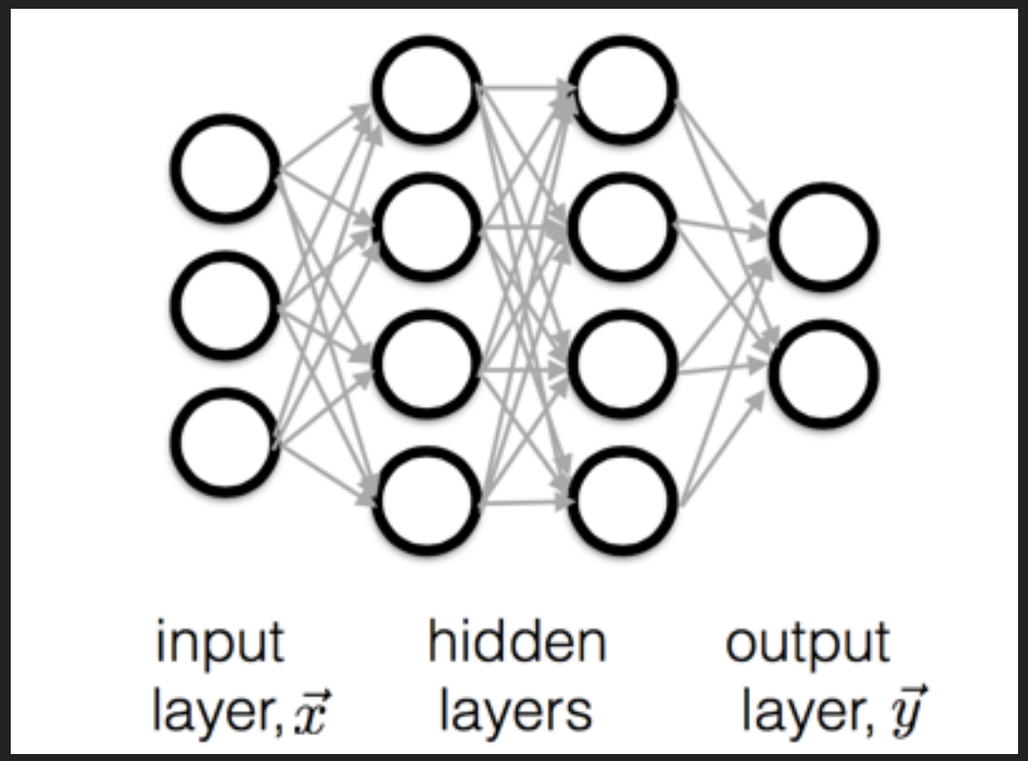
Video analysis for self-driving cars

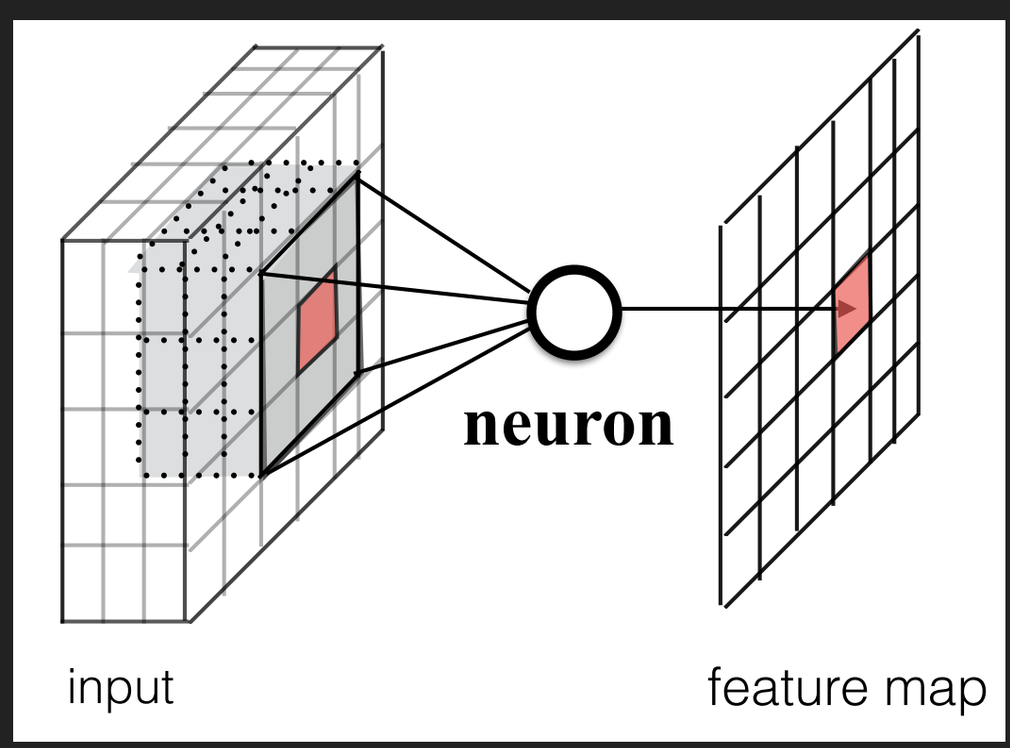


Defeating humans at Go

CONVOLUTIONAL NEURAL NETWORKS

- CNNs differ from “traditional” neural nets in their structure
- CNN “neuron” looks for local, translation-invariant patterns among inputs

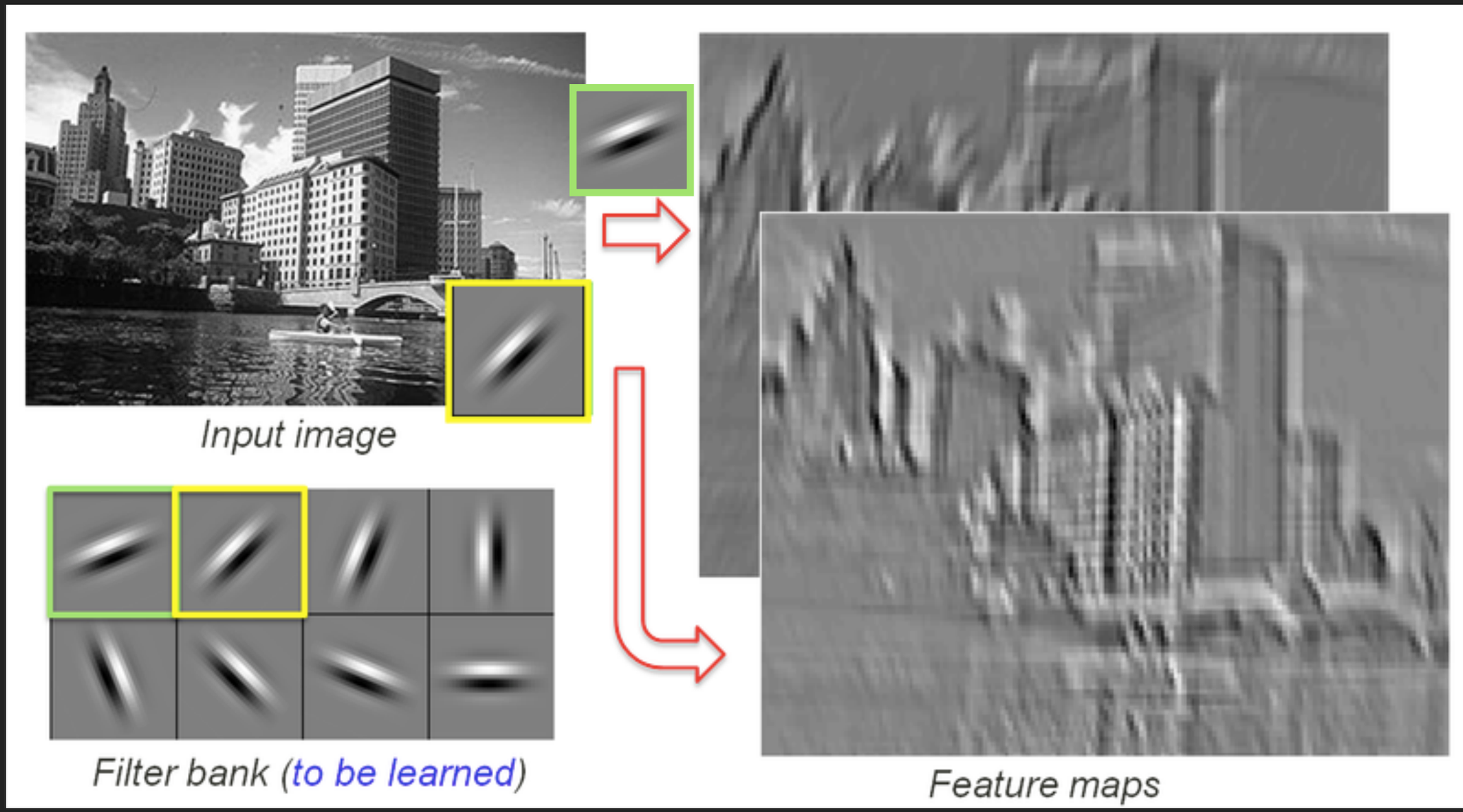




$$f_{i,j}(X) = \sigma\left(W_i \cdot X_j + b_i\right),$$

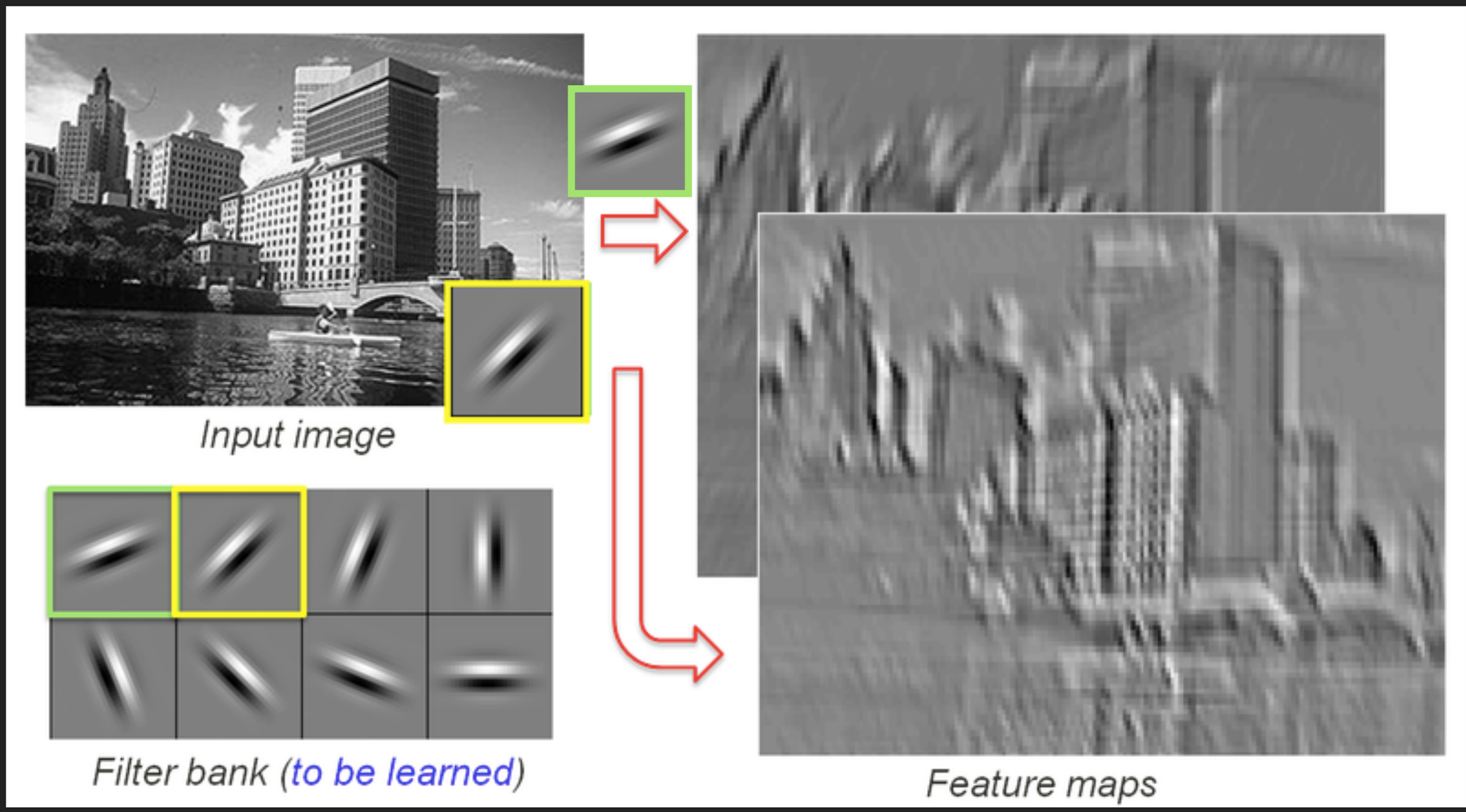
CONVOLUTIONAL FILTER

- Core operation in a CNN is the convolutional filter – identifies the location of patterns in an image
- Here regions of light and dark are where the pattern (or its inverse) matched well within the image

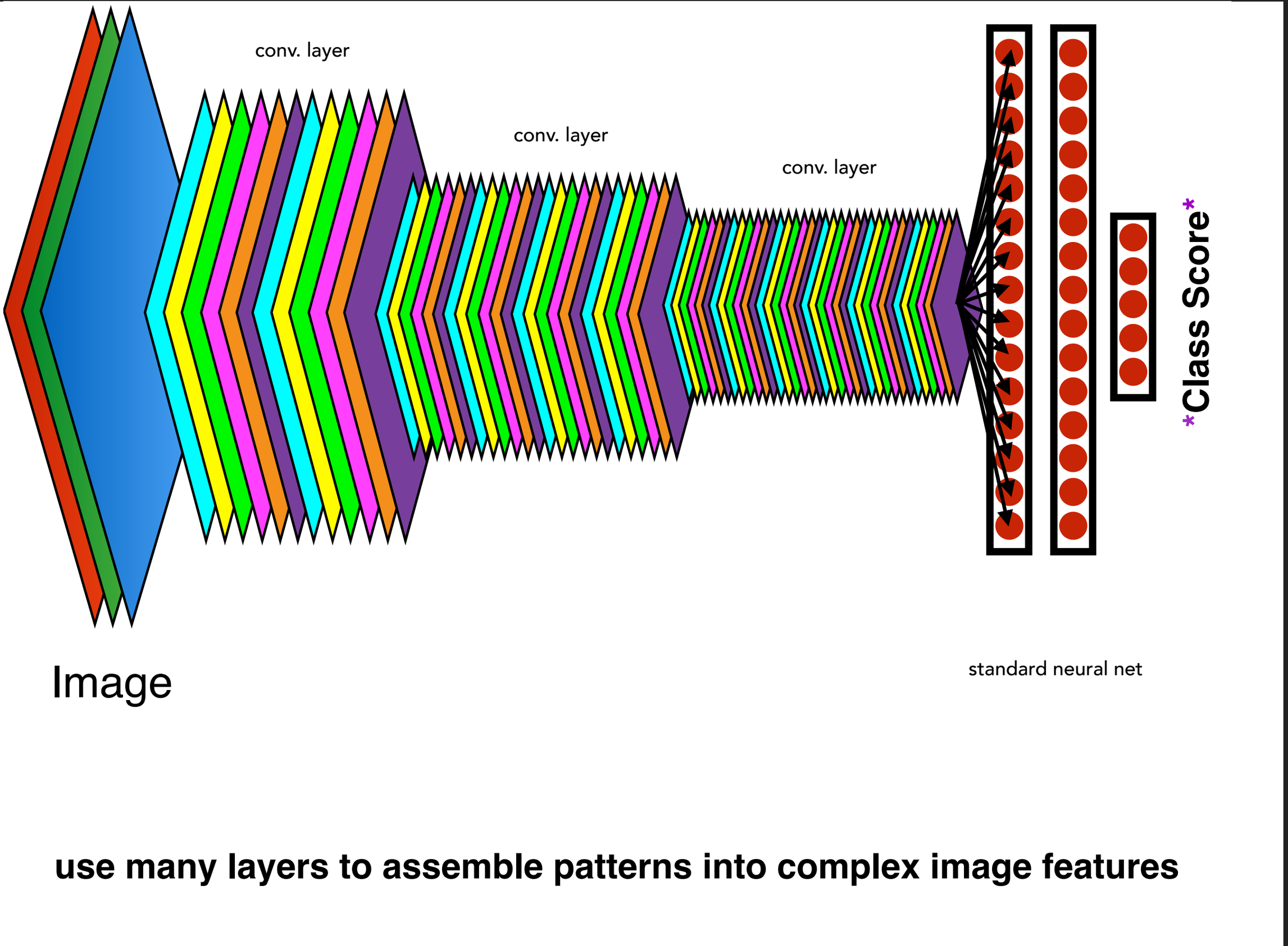


CONVOLUTIONAL FILTER

- one neuron produces one feature map
- operation takes as input an image and outputs an image

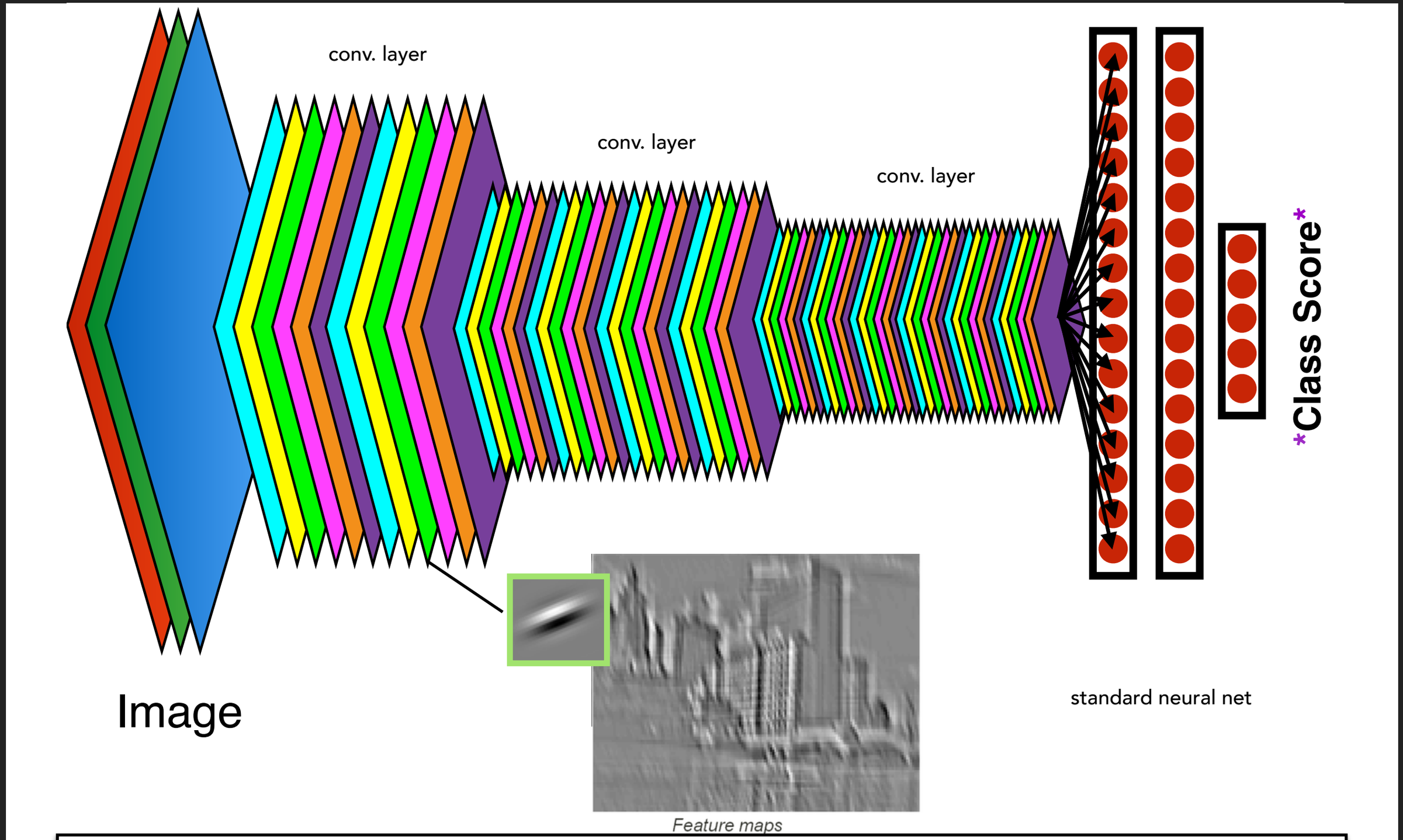


CNN NETWORK ARCHITECTURE



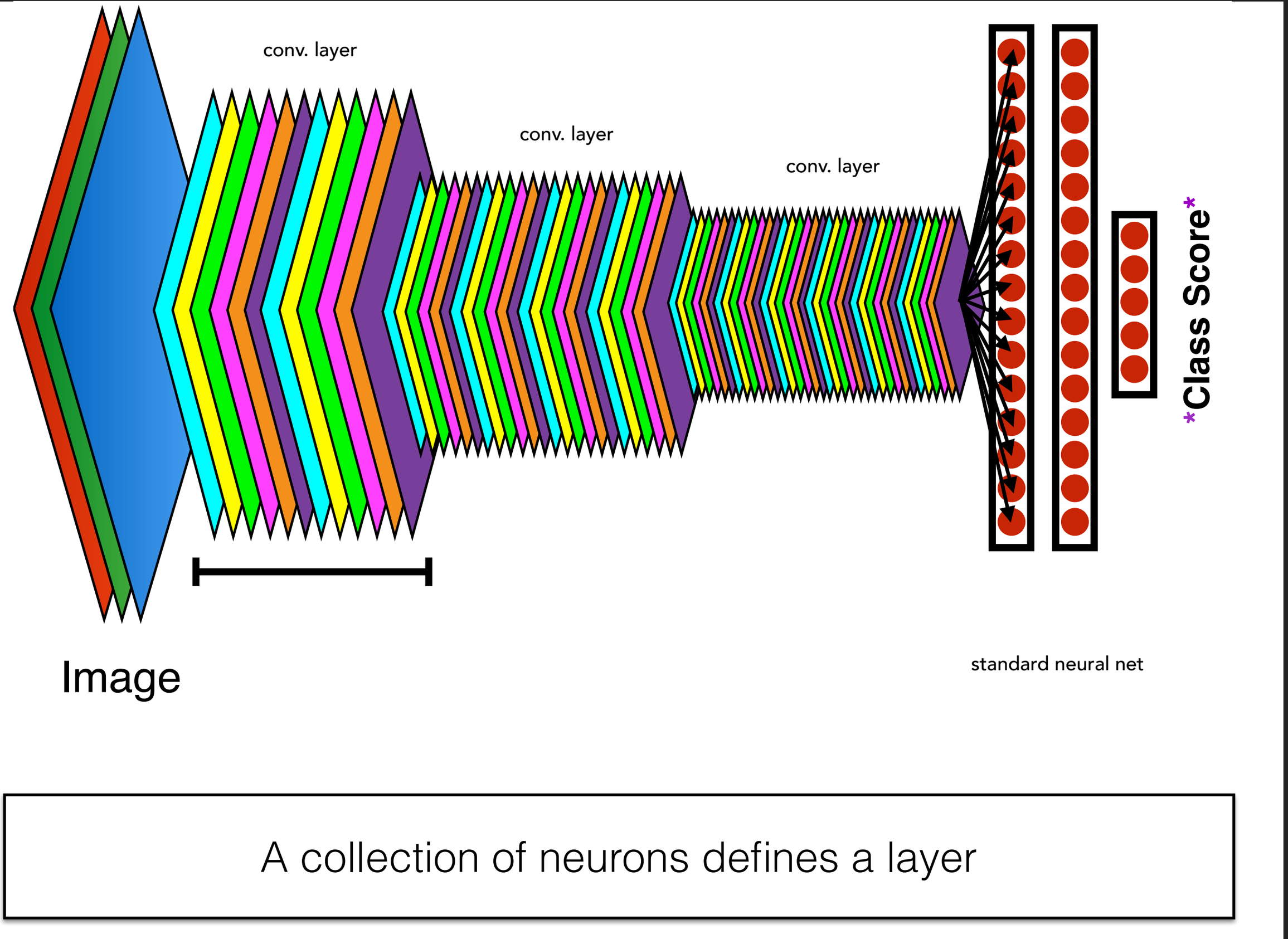
use many layers to assemble patterns into complex image features

CNN NETWORK ARCHITECTURE



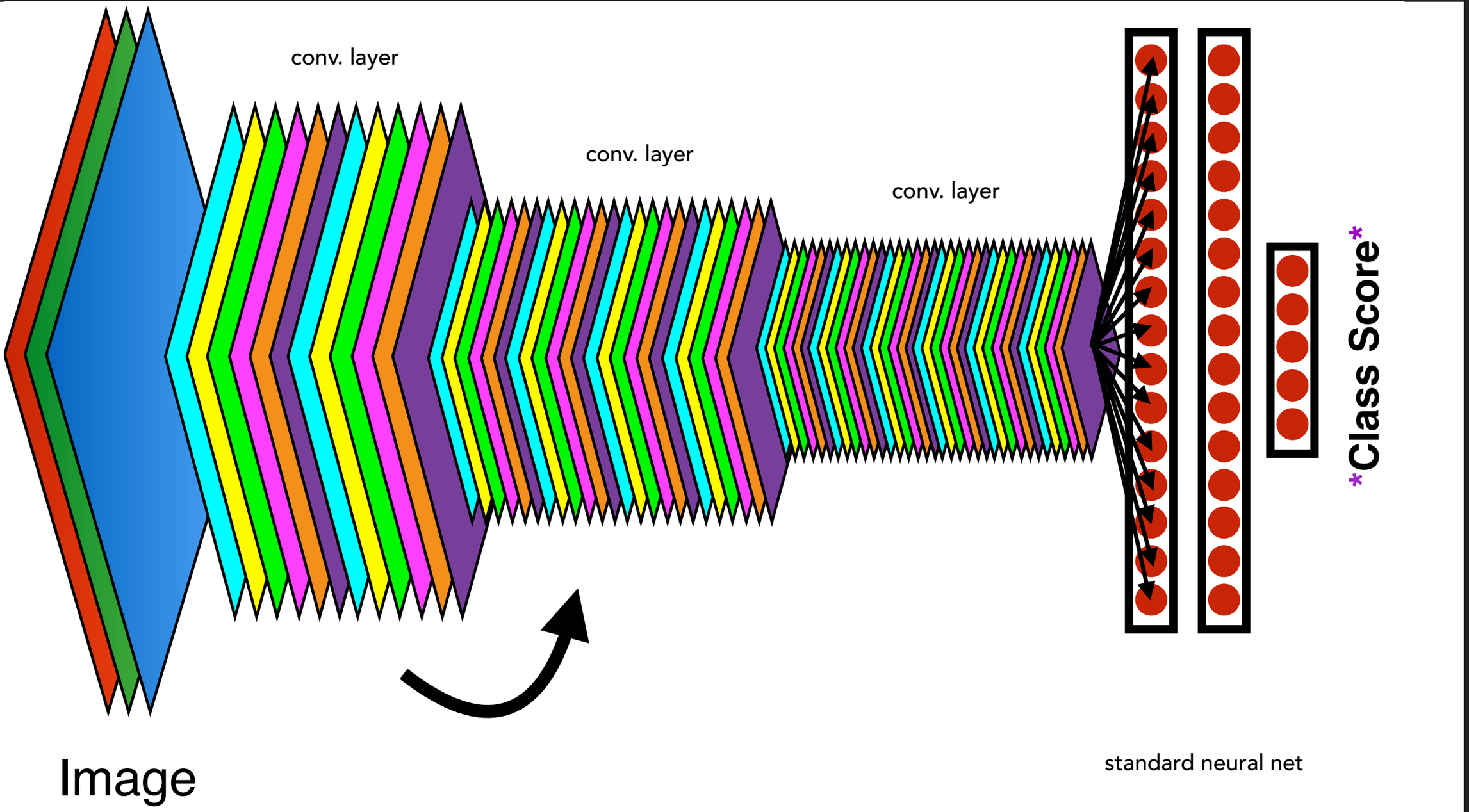
In this cartoon, each sheet represents a feature map — the output of one neuron

CNN NETWORK ARCHITECTURE



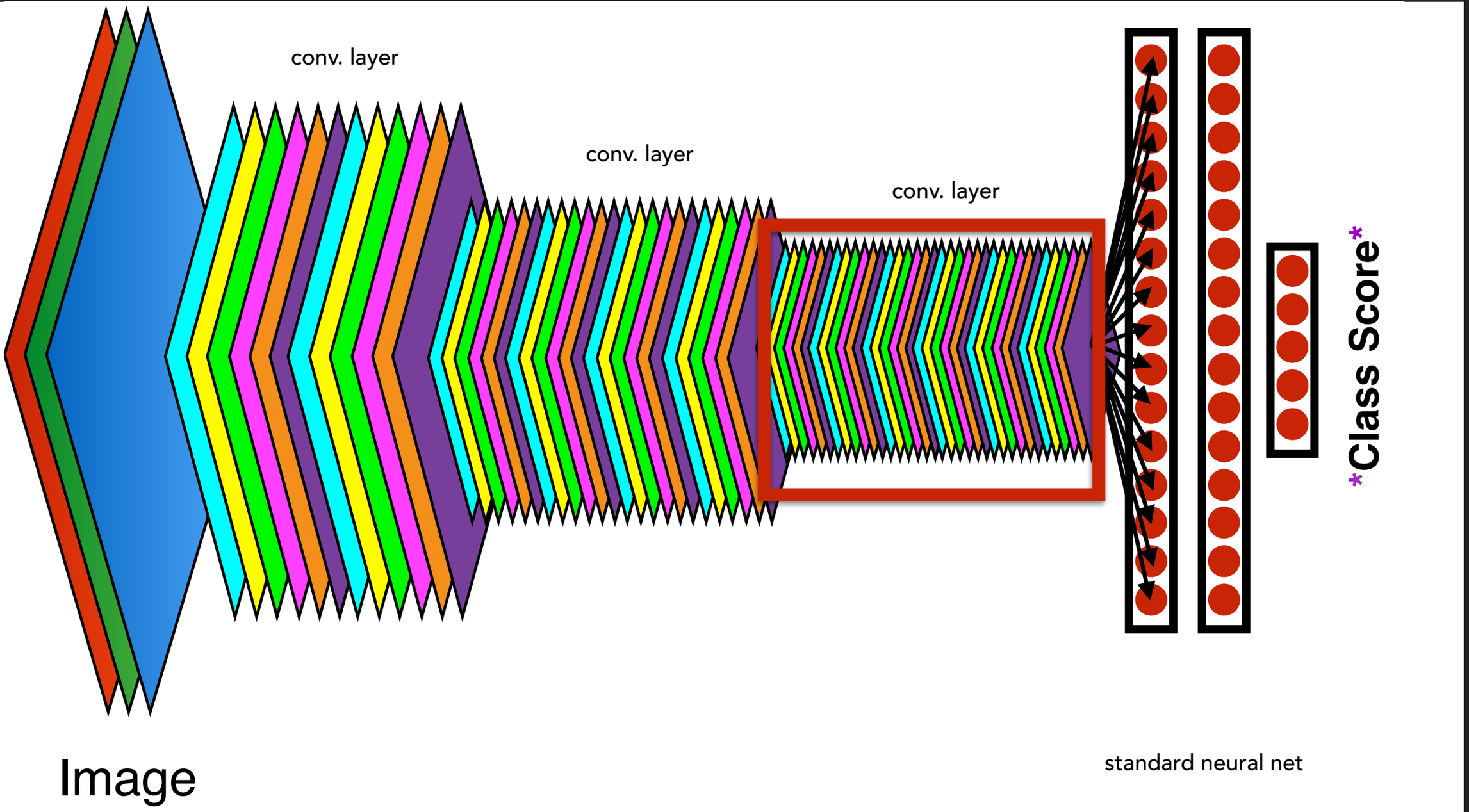
A collection of neurons defines a layer

CNN NETWORK ARCHITECTURE



After each layer, a pooling operation (e.g. maximum score between a neighborhood of pixels) is usually performed

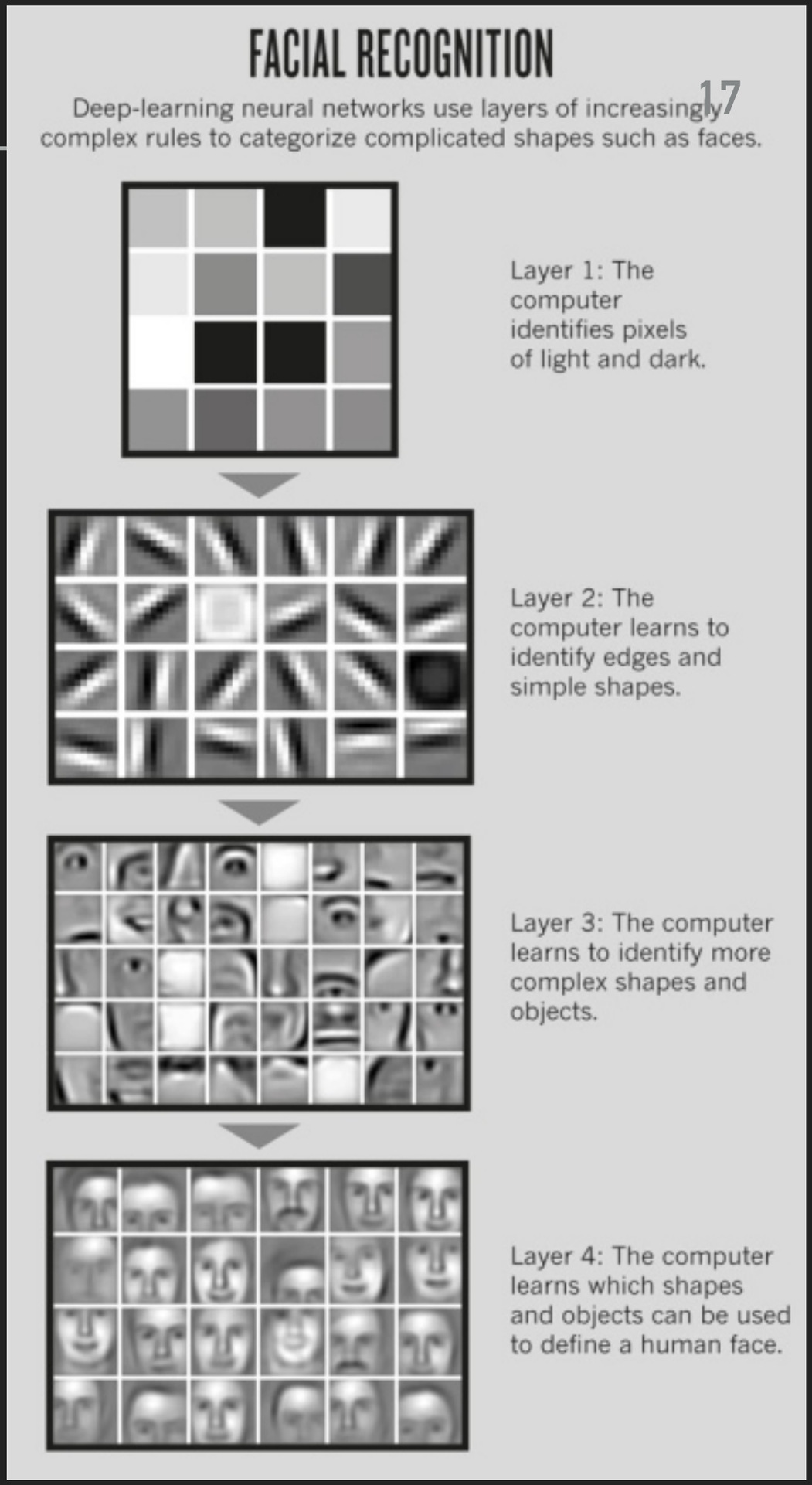
CNN NETWORK ARCHITECTURE



Features from last layer are passed into a fully-connected neural network, which associates the collection of features to a class

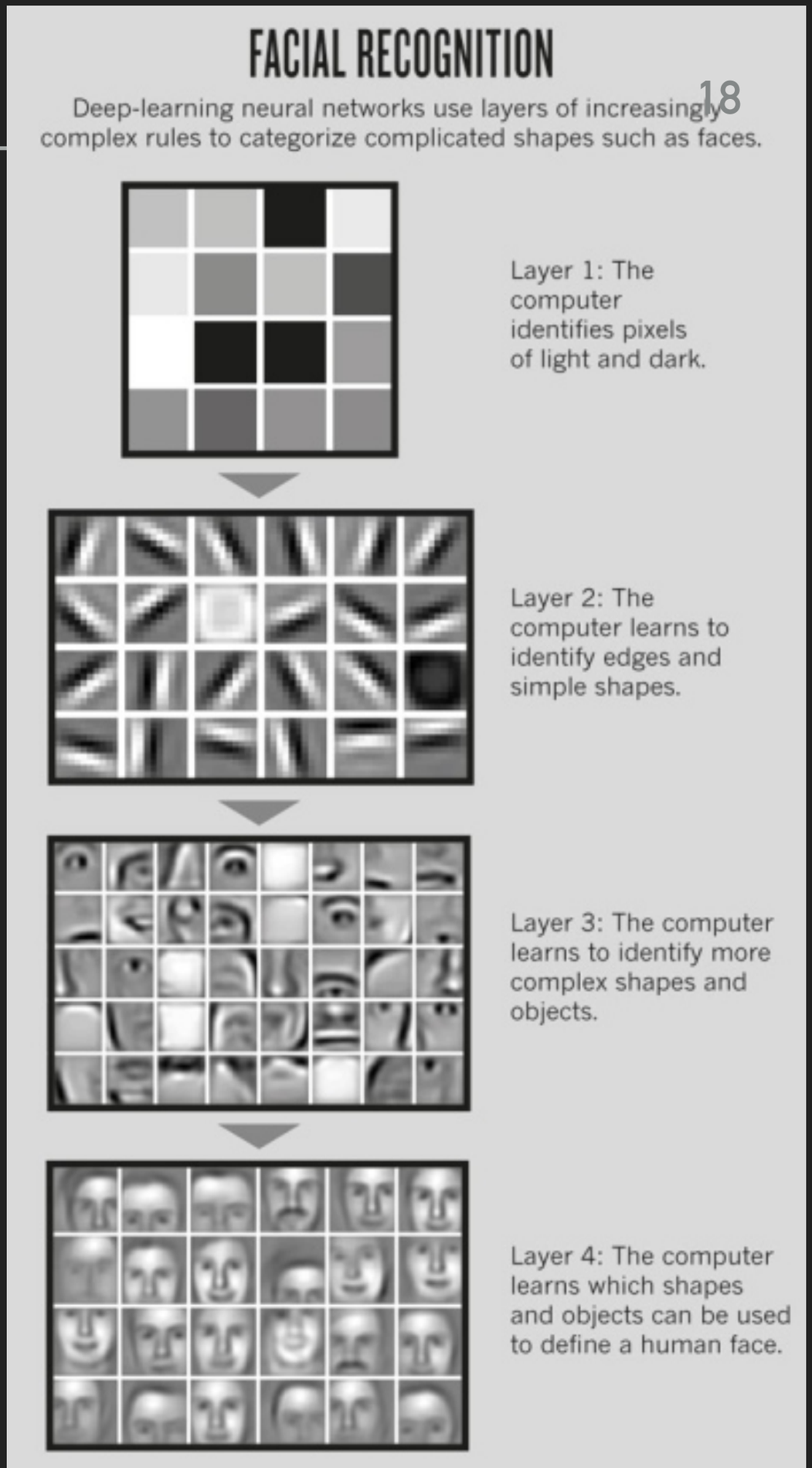
CONVOLUTIONAL NETWORKS

- Consider the task of recognizing faces
- Begin with image pixels (layer 1)
- Start by applying convolutions of simple patterns (layer 2)
- Find groups of patterns by applying convolution on feature maps (layer 3)
- Repeat
- Eventually patterns of patterns can be identified as faces (layer 4)



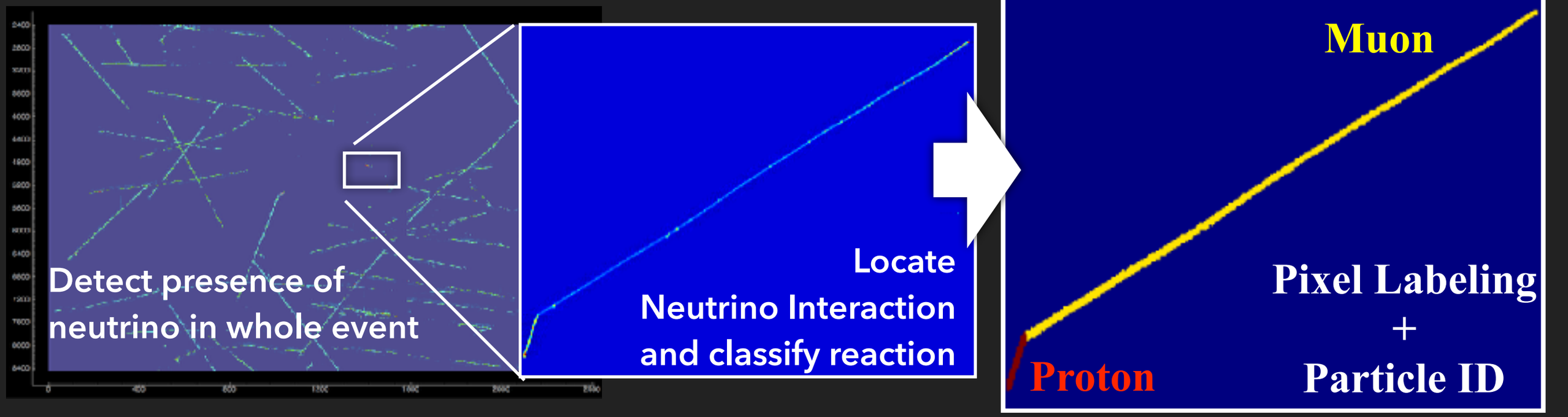
CONVOLUTIONAL NETWORKS

- CNNs learn these patterns (or convolutional filters) by themselves
- That's why CNNs are effective for many different tasks



CNNS IN MICROBOONE (AND LARTPCS)

- Explored several CNN algorithms that perform tasks directly applicable to our problem
  - Image classification
  - Object detection
  - Pixel labeling

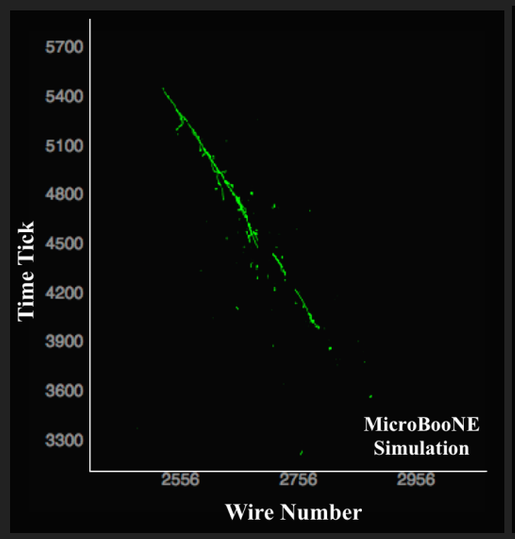


$$\nu_\mu + n \rightarrow \mu + p$$

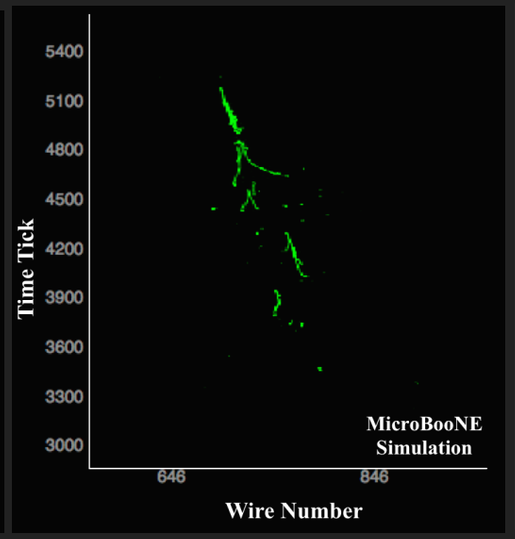
PROOF OF PRINCIPLE STUDY

- Study with images from simulation
- To start: can network tell these four particles apart?
- Important particles in analyses

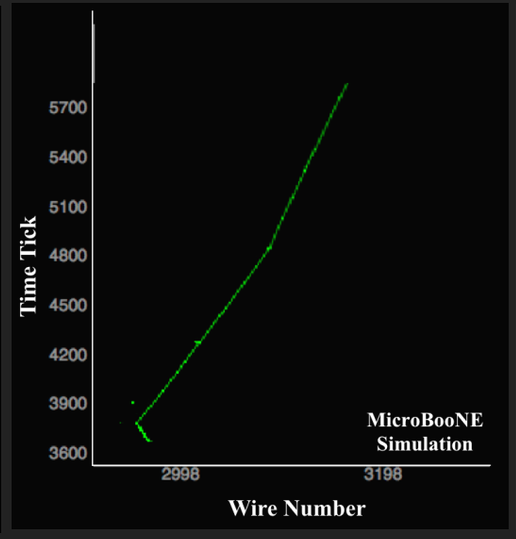
Electron



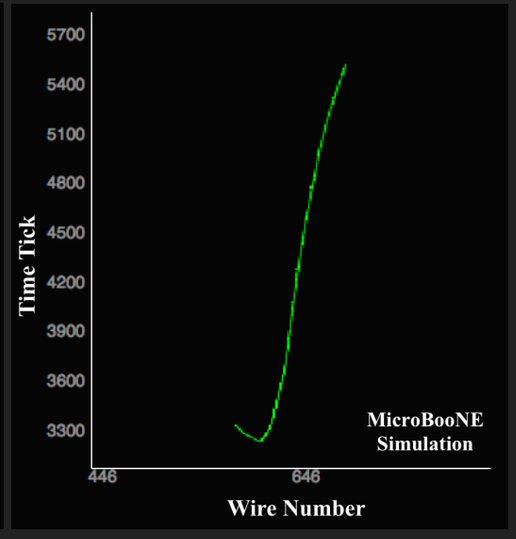
Photon



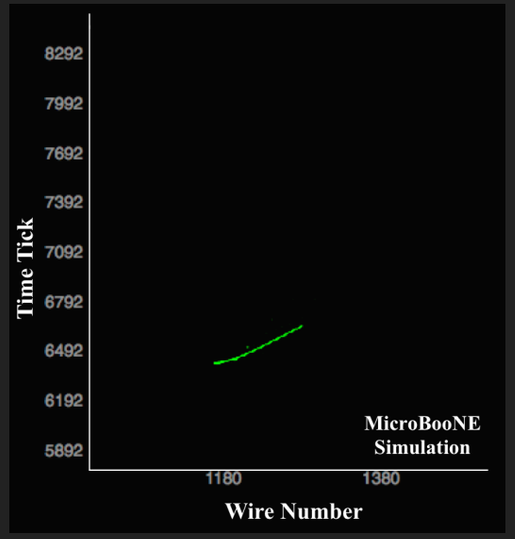
Charged Pion



Muon



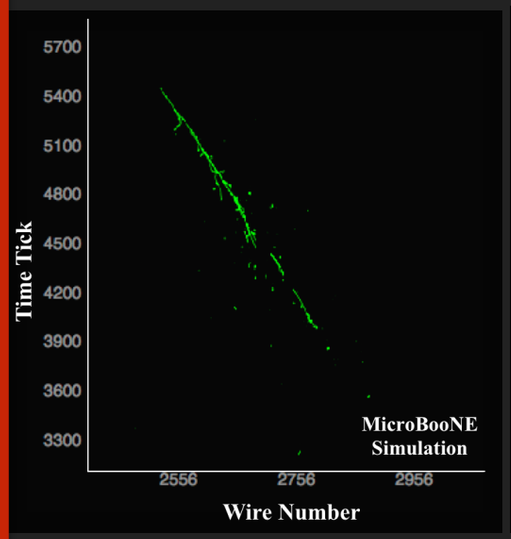
Proton



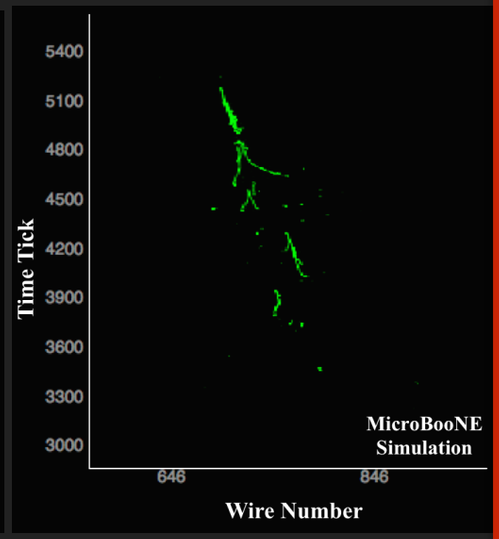
PROOF OF PRINCIPLE STUDY

- Note that there are two broad classes of topologies
  - Shower: electrons and photons produce electromagnetic cascade
  - Tracks: heavier particles travel mainly in straight line
- Mostly use amount of charge deposited per unit traveled to distinguish between particles of similar topology (also decay products useful as well)

Electron

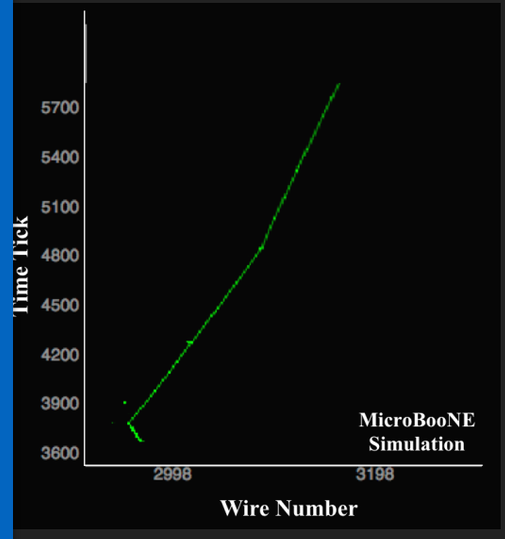


Photon

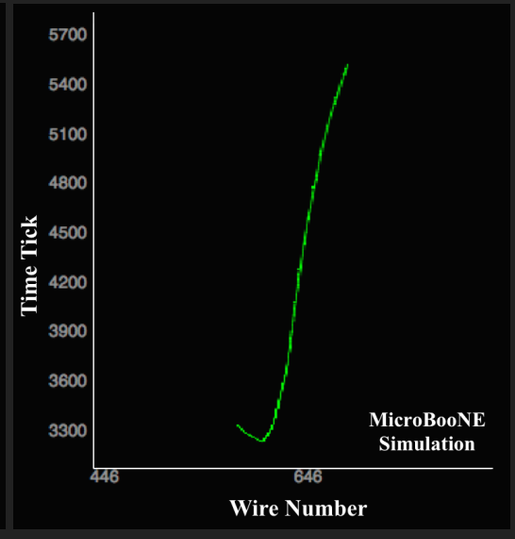


Showers

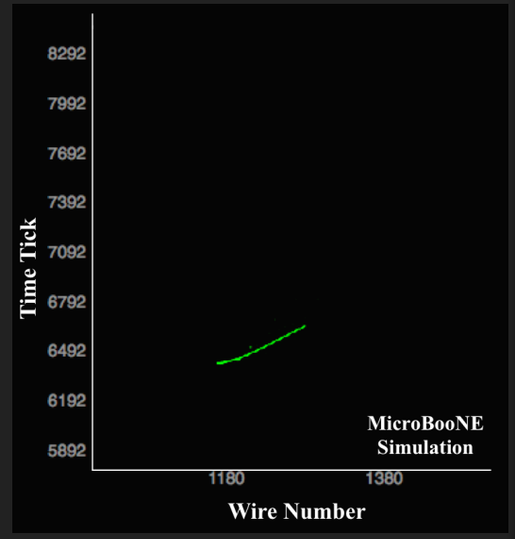
Charged Pion



Muon



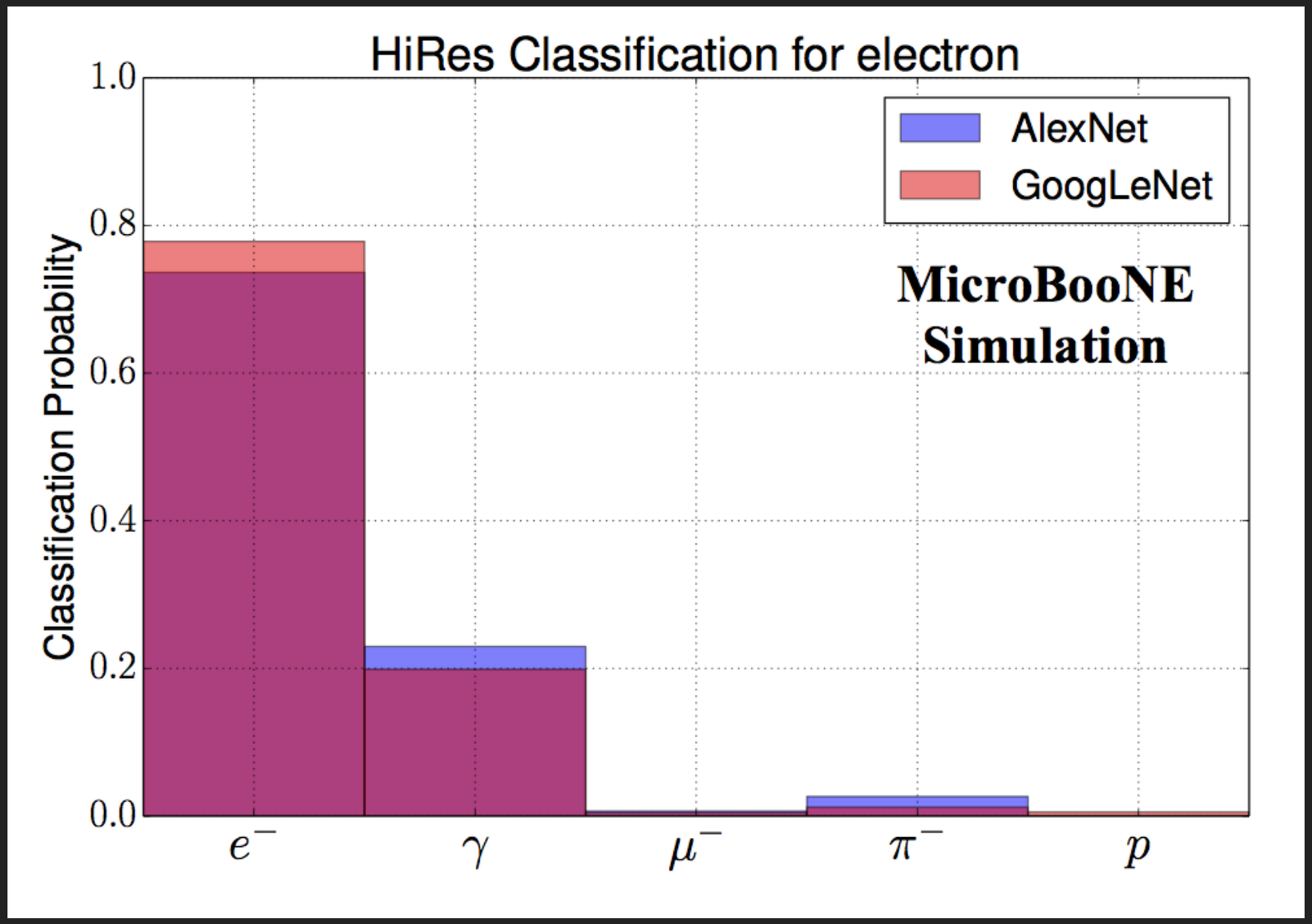
Proton



Tracks

PROOF OF PRINCIPLE STUDY

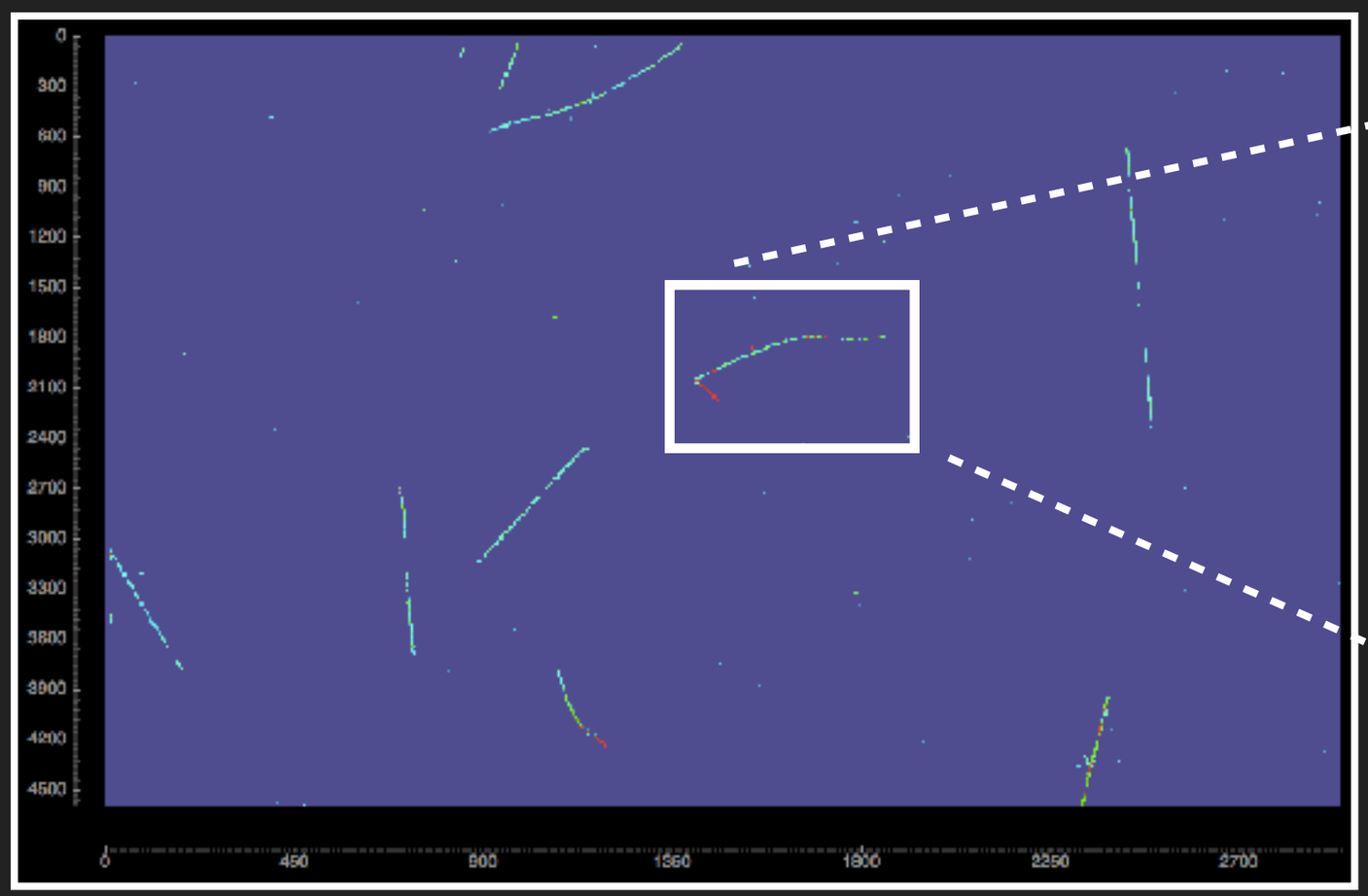
- Study with images from simulation
- High-lighting electron ID: important for finding signal interactions in current/future LArTPCs $\nu_e + n \rightarrow e + p$

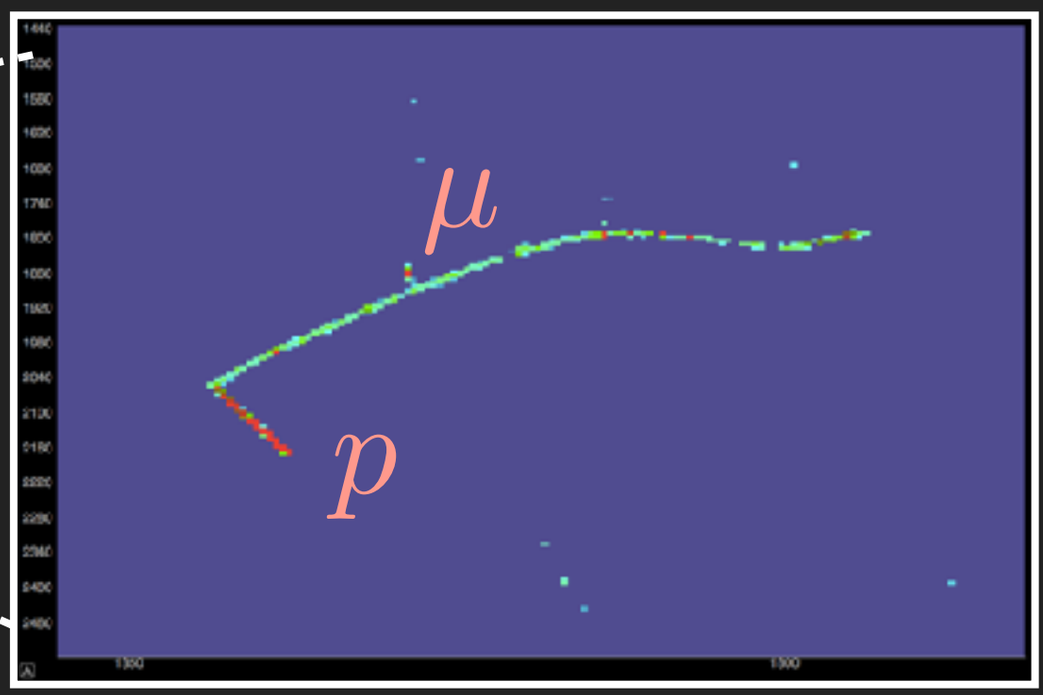


NEUTRINO INTERACTION DETECTION

- Explored class of problems known as objet detection for LArTPCs
- For surface near the detectors, could be used to locate regions of interest in the detector

$$\nu_\mu + n \rightarrow \mu + p$$

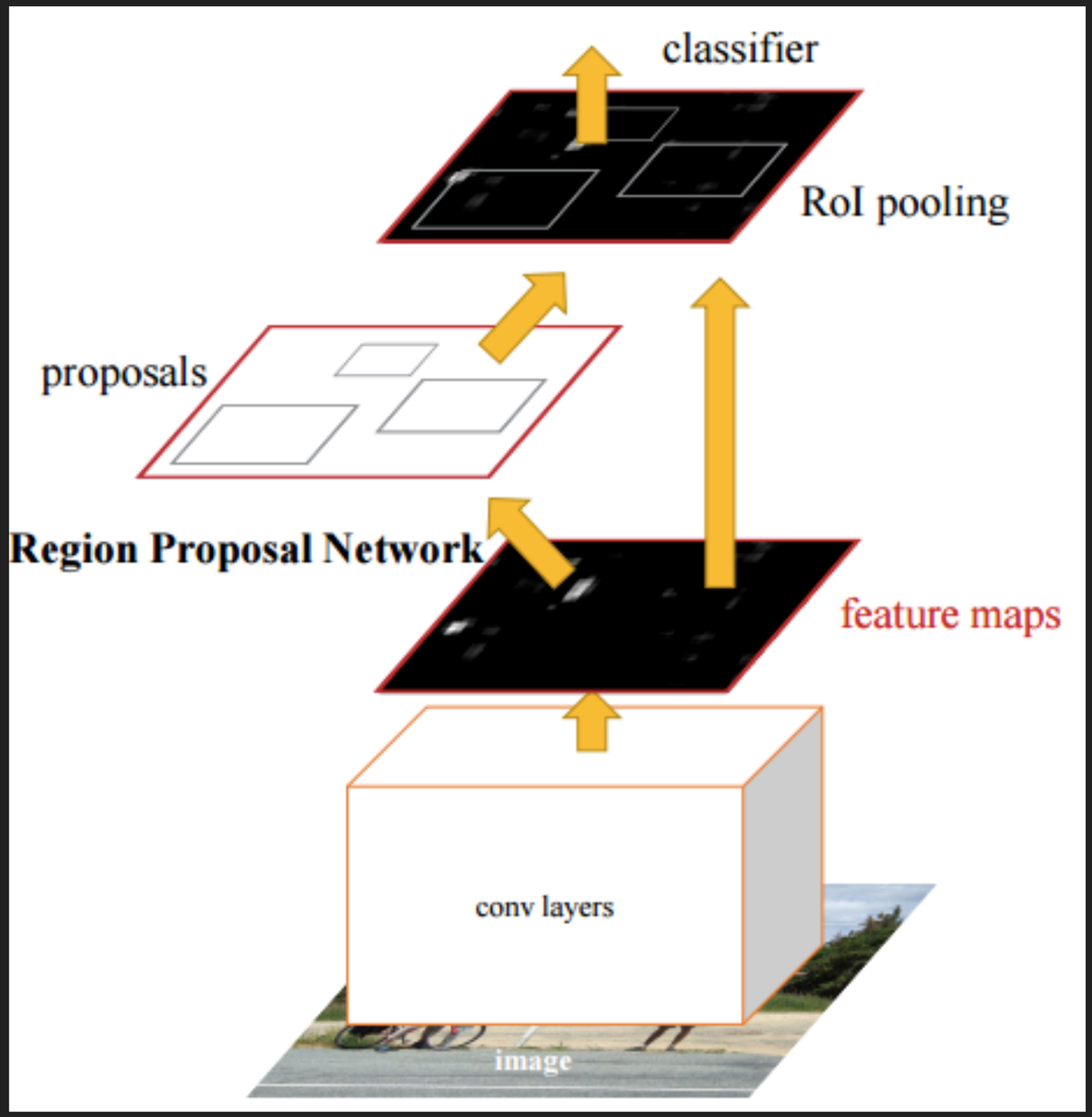




Note: had use reduce resolution image for network

RESULT: NEUTRINO DETECTION

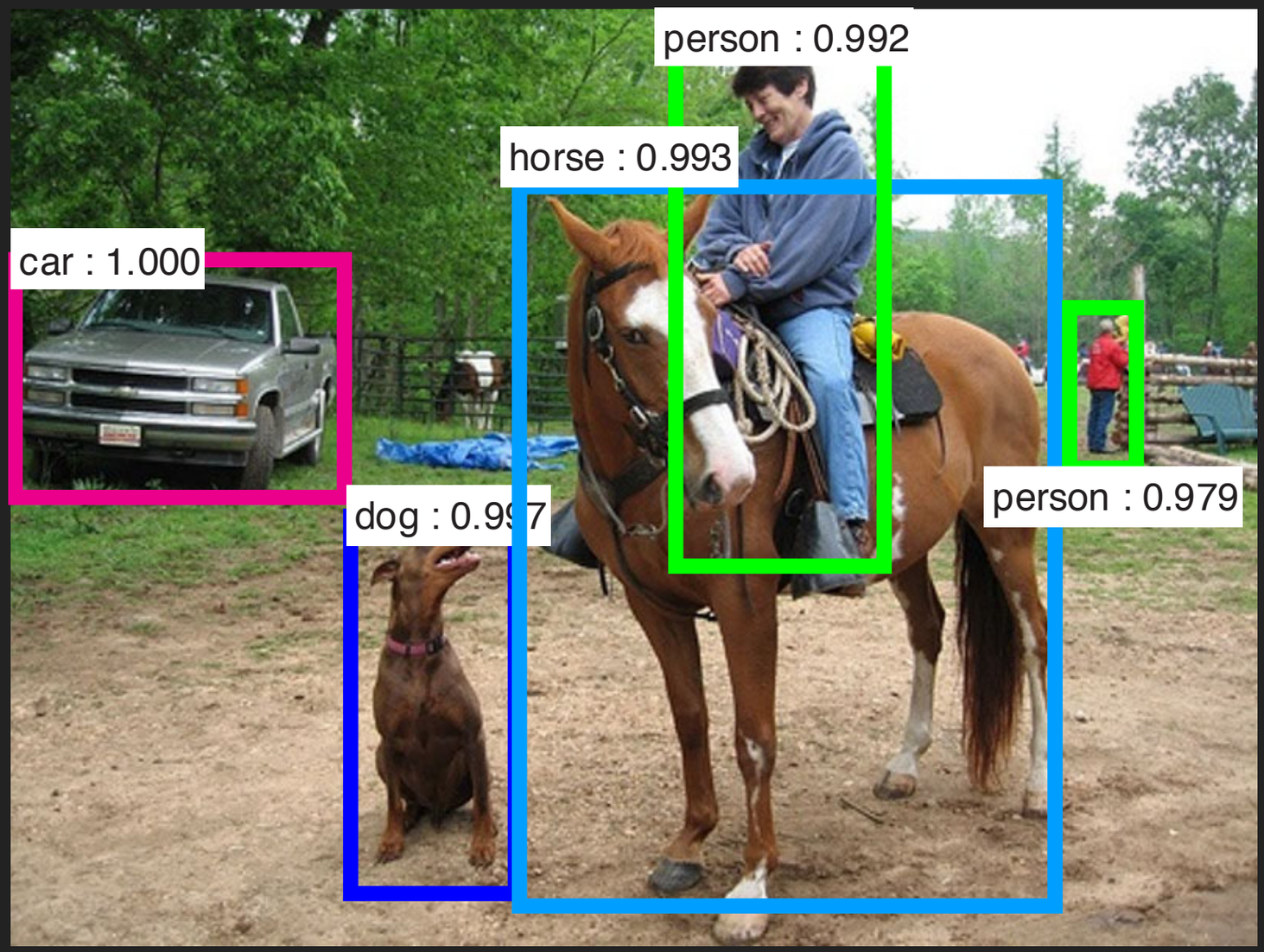
- Key element in faster-RCNN is the Region Proposal Network
- Takes image features and determines if a given location contains an “object”
- Top regions with objects are passed to next stage, a typical classifier



faster RCNN arXiv:1504.08083

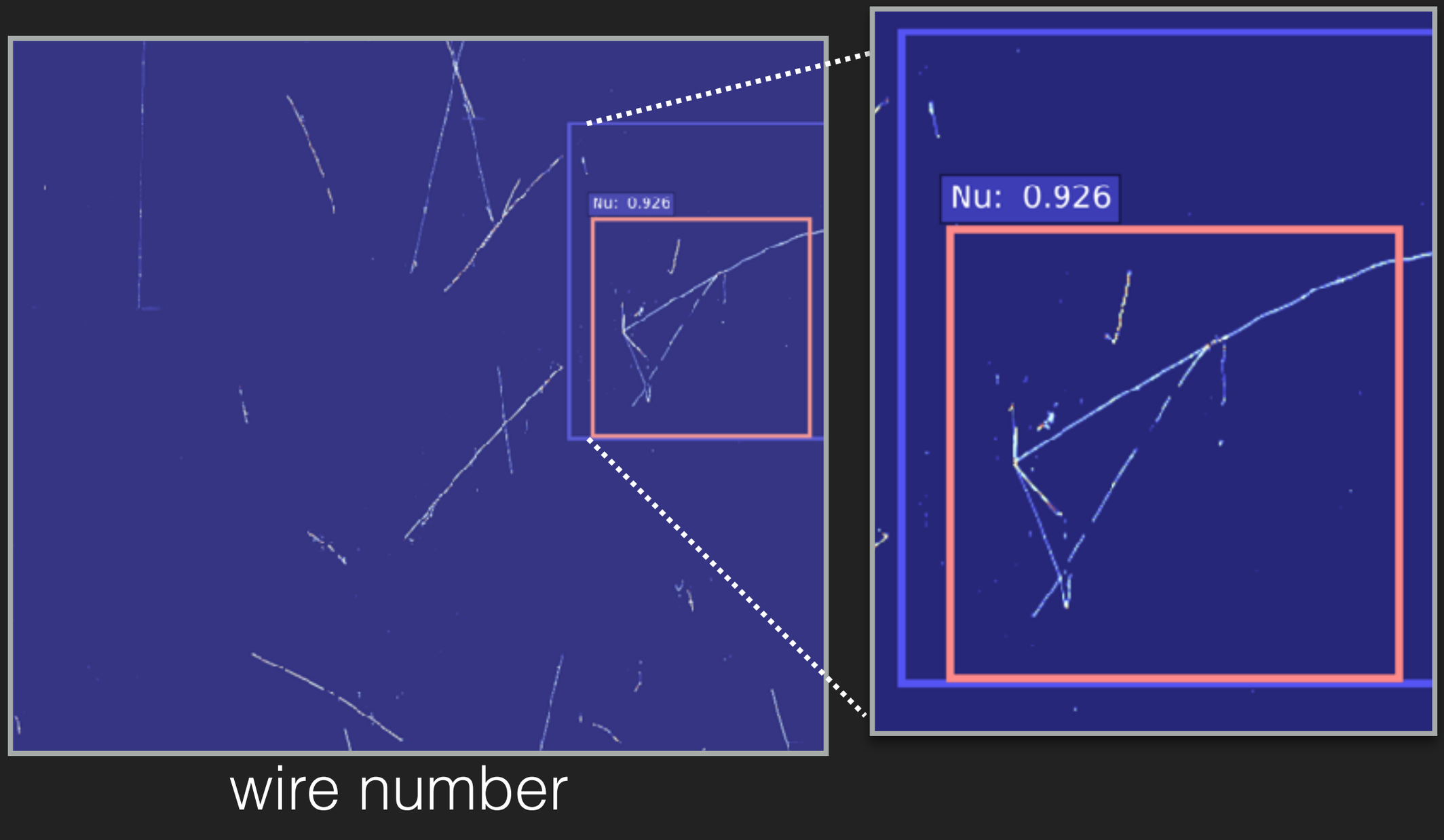
FASTER R-CNN

- Network output are classified regions of the image



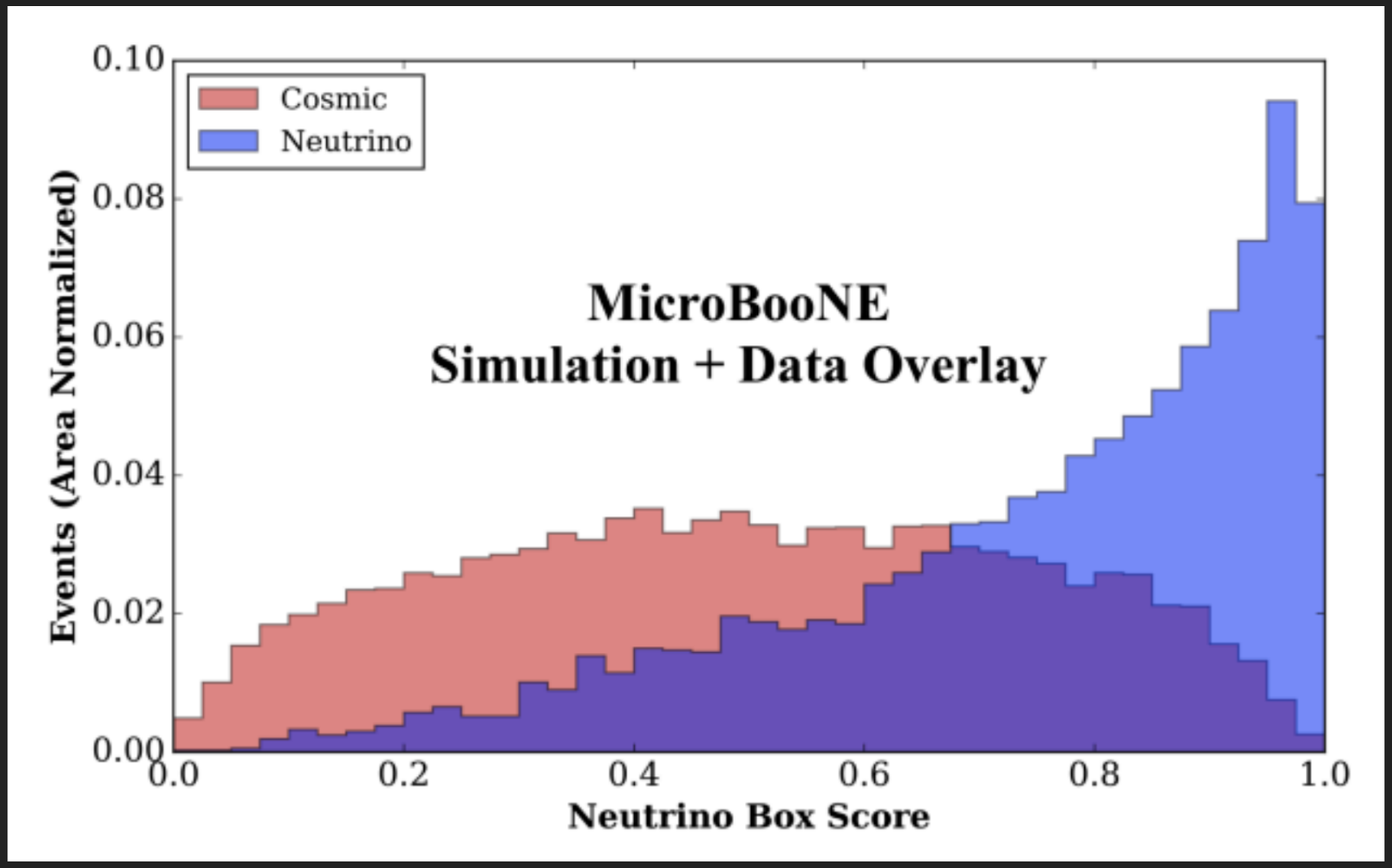
RESULT: NEUTRINO DETECTION

- Trained a network to place a bounding box around a neutrino interaction within a whole event view



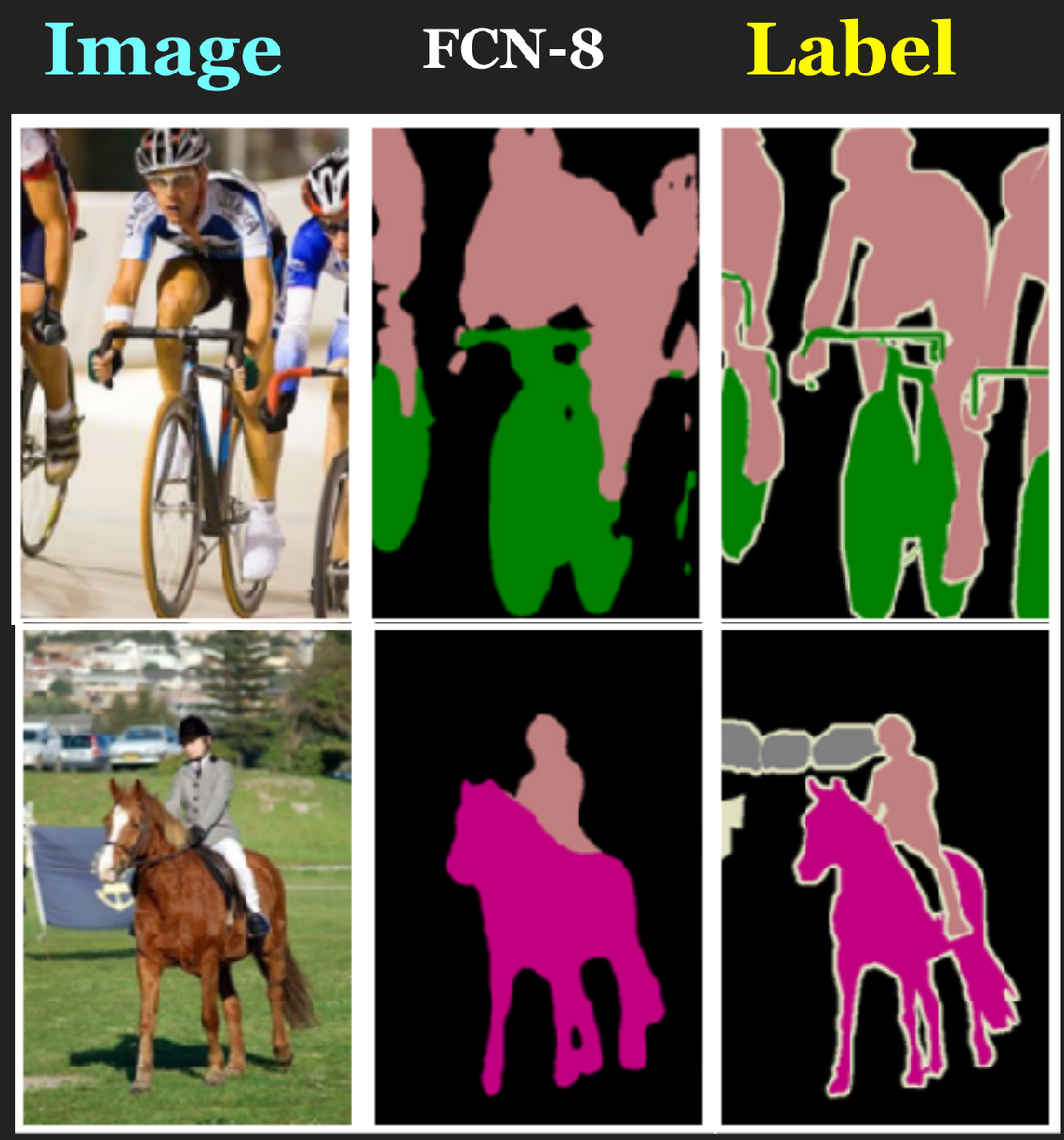
RESULT: NEUTRINO DETECTION

- Distribution of scores for regions overlapping with neutrinos (blue) versus background (red)



SEMANTIC SEGMENTATION

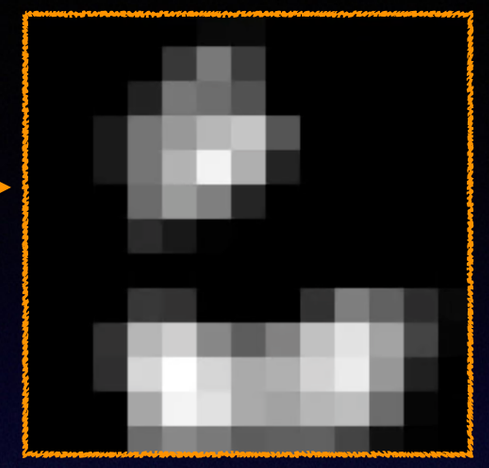
- This task asks the network to label the individual pixels as belong to some class



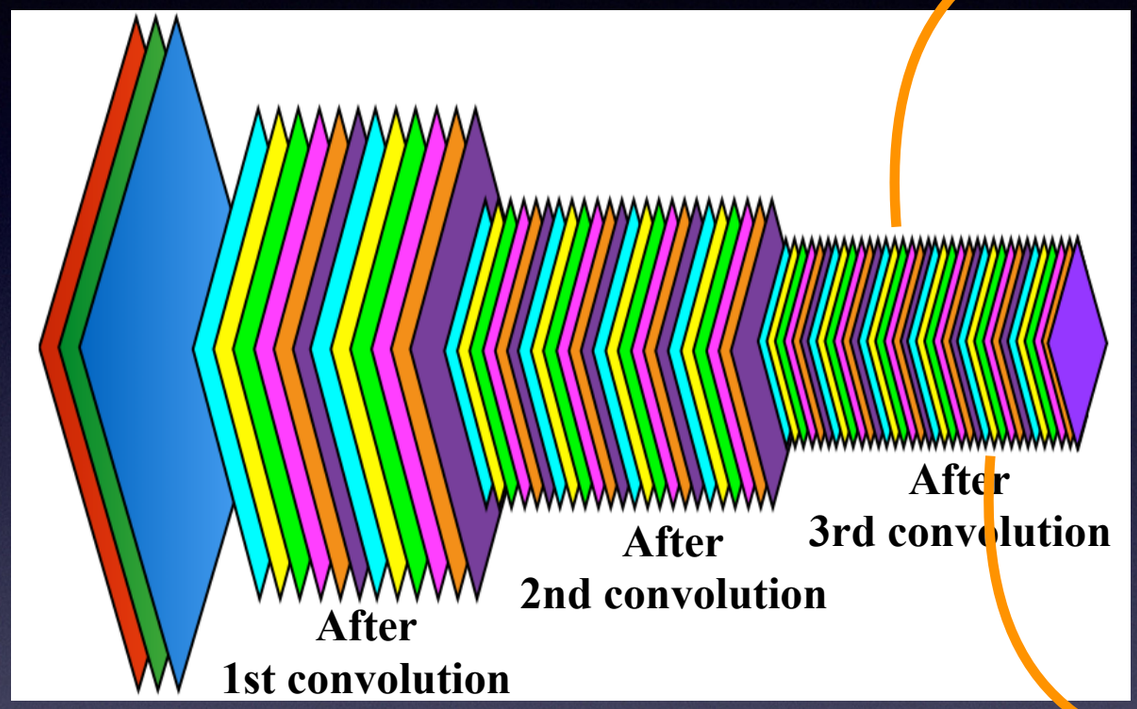
FCN-8: ***Fully-Convolutional-Network*** (FCN)

VISUALIZING FEATURES

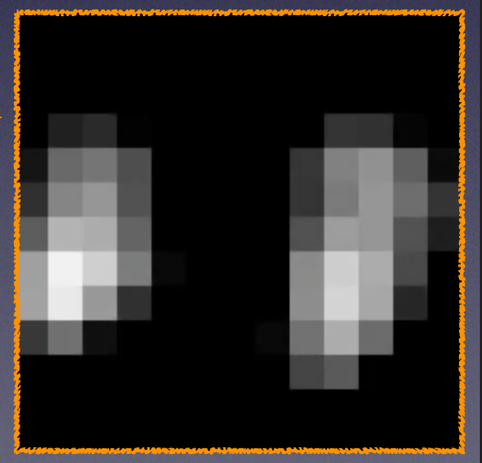
Feature extraction by CNN



“Written Texts” feature map



Feature extraction by CNN

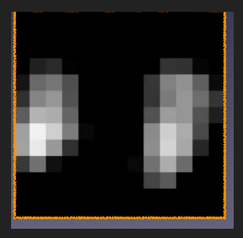


“Human Face” feature map

After steps of down-sampling, **“feature map” still preserves a rough object location** information

Pictures from DeepVis toolkit from http://yosinski.com/deepvis

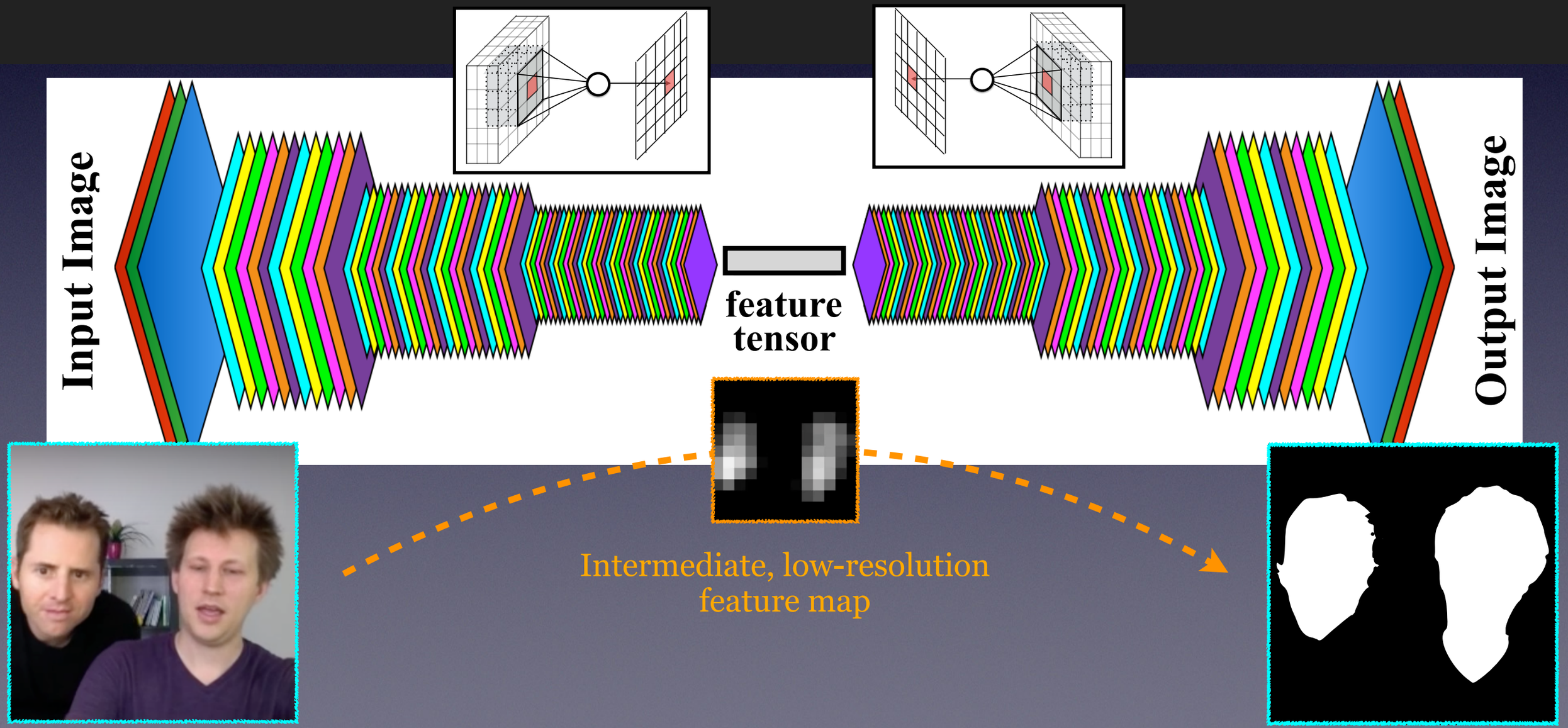
Recall *Image Classification*



- Individual feature maps (produced by a neuron in a layer) contain spatial information
- However, down-sampled
- For semantic segmentation, we want to use this information
- Need to:
  - up-scale feature map
  - smooth to match contours of object in original image

SEMANTIC SEGMENTATION

- Solution: have a network understand how to upscale the information for you
- Now, have feature information for every pixel – can use it to make a pixel-level decision as to what object the pixel belongs to

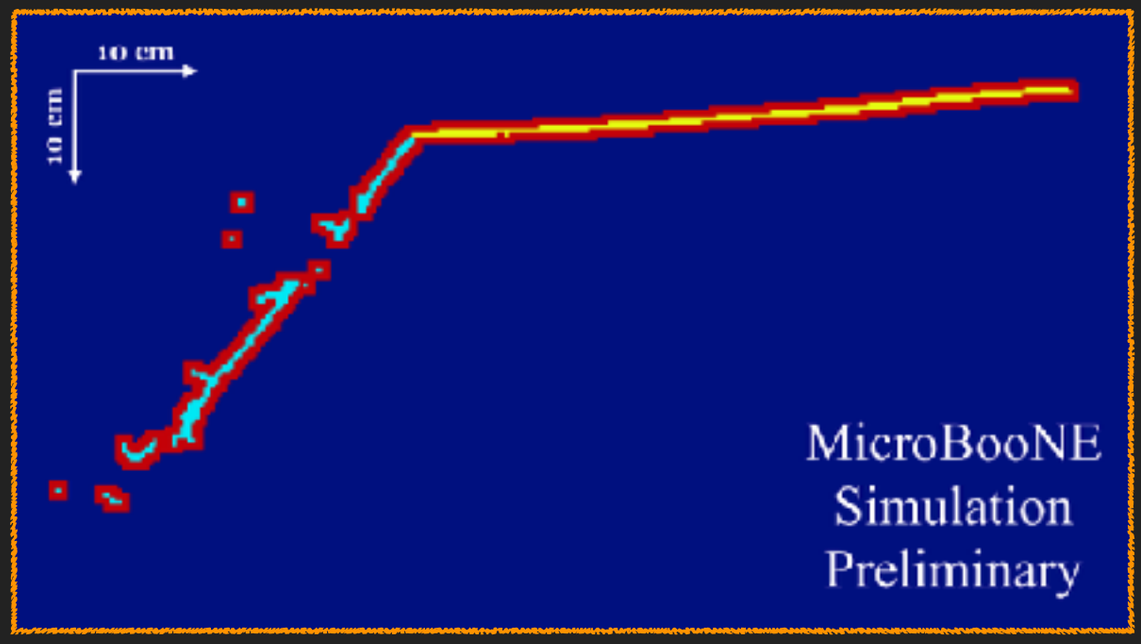


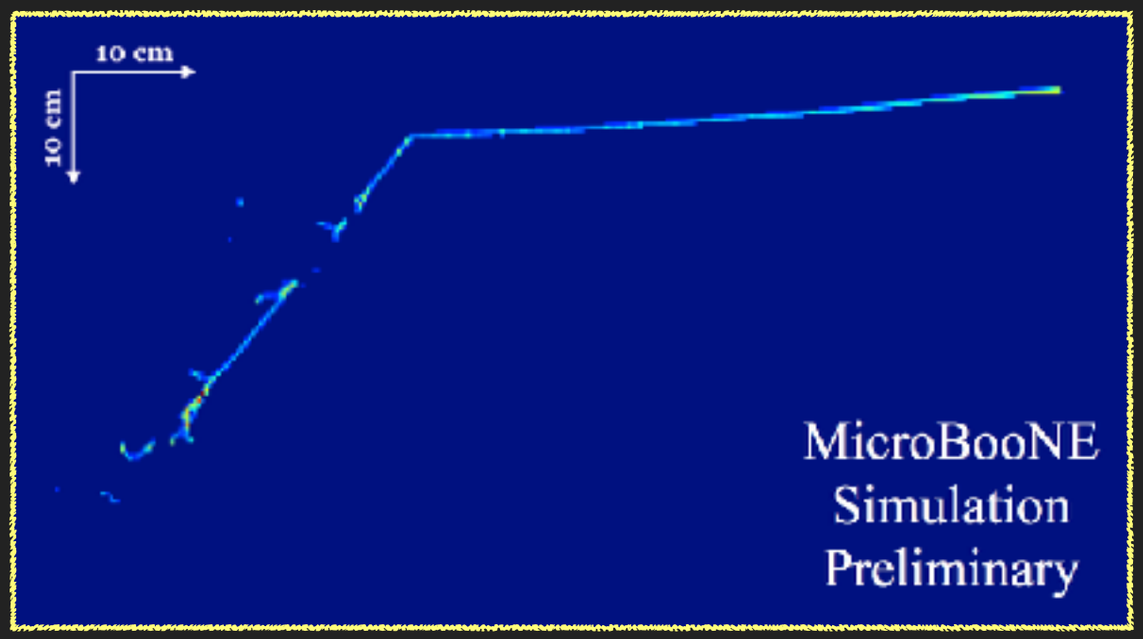
SEMANTIC SEGMENTATION IN LARTPC

Supervised Training (UB)

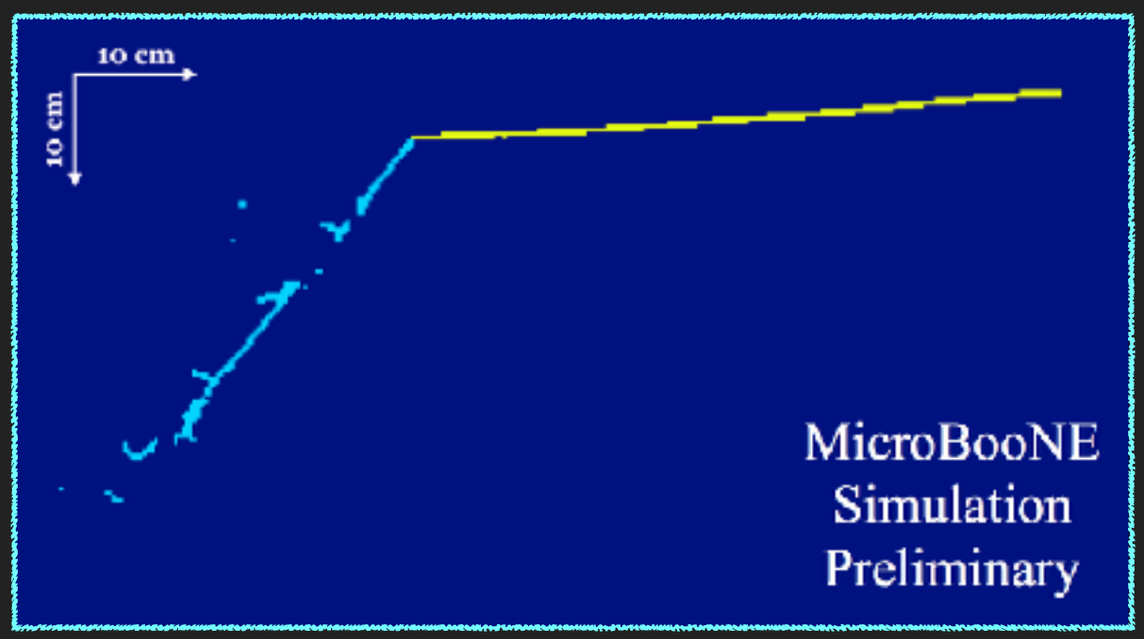
- Assign pixel-wise "weight" to penalize mistakes
- Weights inversely proportional to each "category" of pixel count
- Useful for LArTPC images (low information density)
- U-Net (arXiv:1505.04597)

"Weight" Image
(for training)





Input Image



"Label" Image
(for training)

DATA/MC DISAGREEMENT

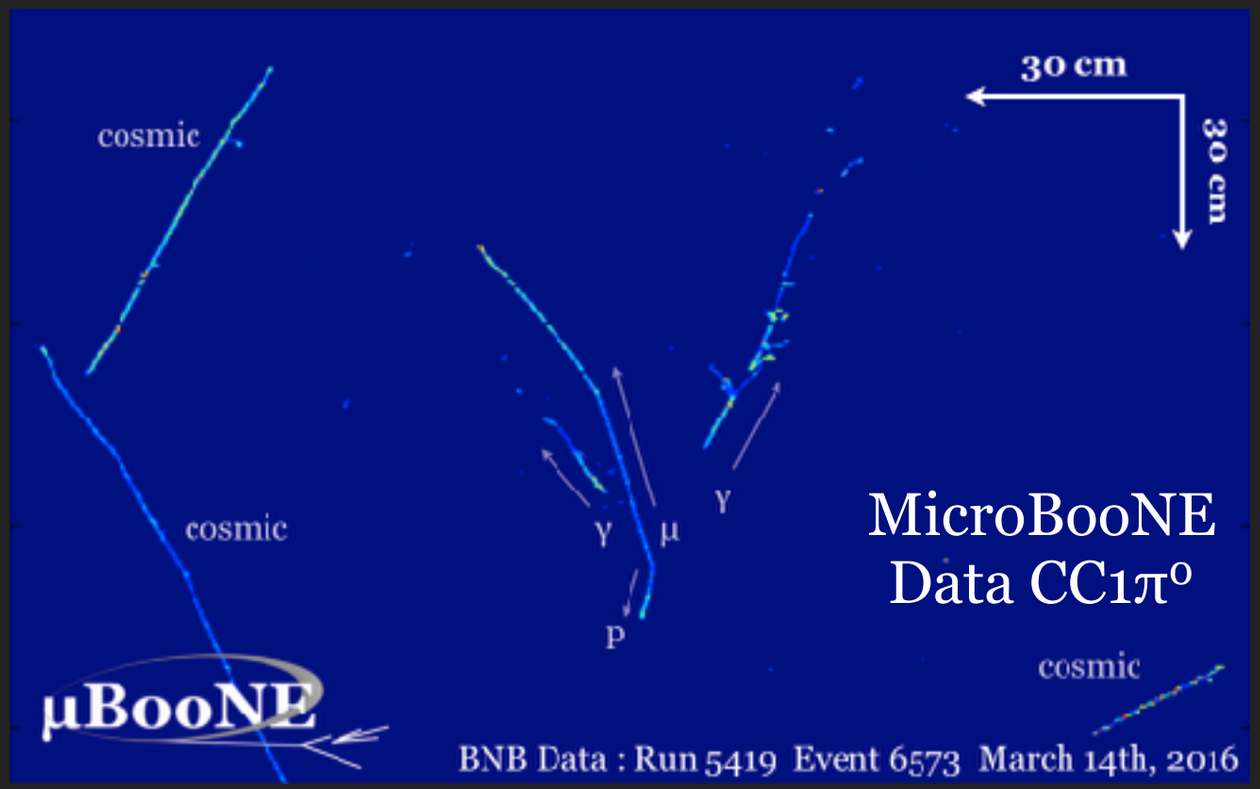
- In physics, we have the luxury of generating large training data sets using simulation
  - In other contexts, training sets are costly and hard to come by
- But downside is that training set will never be a perfect representation of the data – simulation will always have some level of mis-modeling, even if small
- Important to have validation data set

VALIDATION SETS

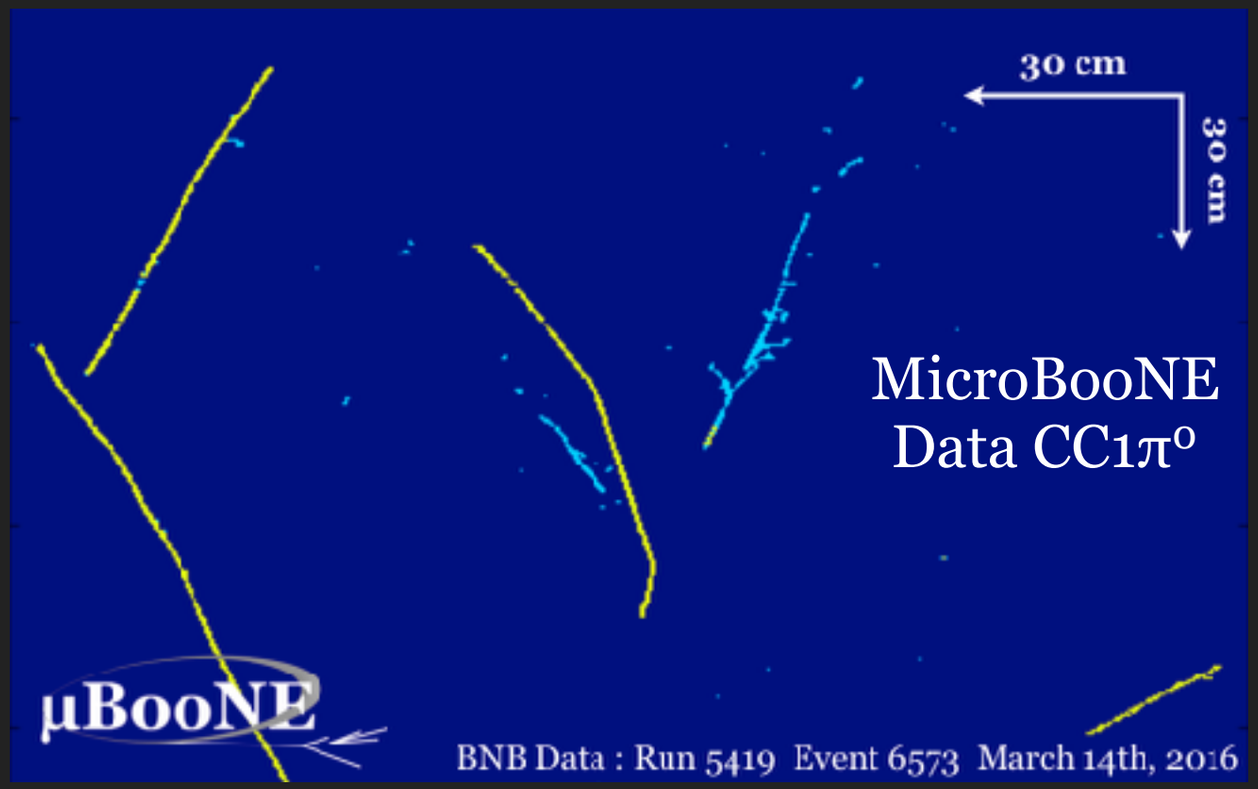
- Sources of validation sets
  - Another similar detector that produces a sample of neutrino interactions
  - Side-bands
  - Cosmic ray particles
  - Some hybrid data/MC event

SEMANTIC SEGMENTATION

- Example check of Semantic Segmentation on side-band neutrino interaction



ADC Image



Network Output

NEXT STEPS

- On-going effort to understand how to quantify and mitigate systematics from training on MC events
  - Testing on cosmic ray samples
  - Semantic aware-training
  - Feature-constrained training (to avoid leaning MC-specific features)

SUMMARY

- CNNs provide a way to tackle many of the problems faced in event reconstruction in LArTPC data – but potentially for all kinds of particle detector data
  - Classification, object detection, semantic segmentation
  - Details in paper: JINST 12 (02) P02017
  - Other neutrino detectors (Nova): arXiv:1604.01444
- But many more applications to explore
- Working to understand how to bridge the MC-data divide
- Physic Experiment-Friendly (i.e. ROOT) interfaces to Caffe and Tensorflow
  - LArCV: https://github.com/LArbys/LArCV
  - Caffe 1-fork: https://github.com/LArbys/caffe

THANK YOU

- Thanks for your attention
- And thank you to the funding agencies for making this work possible

BACK-UPS

LARTPCS

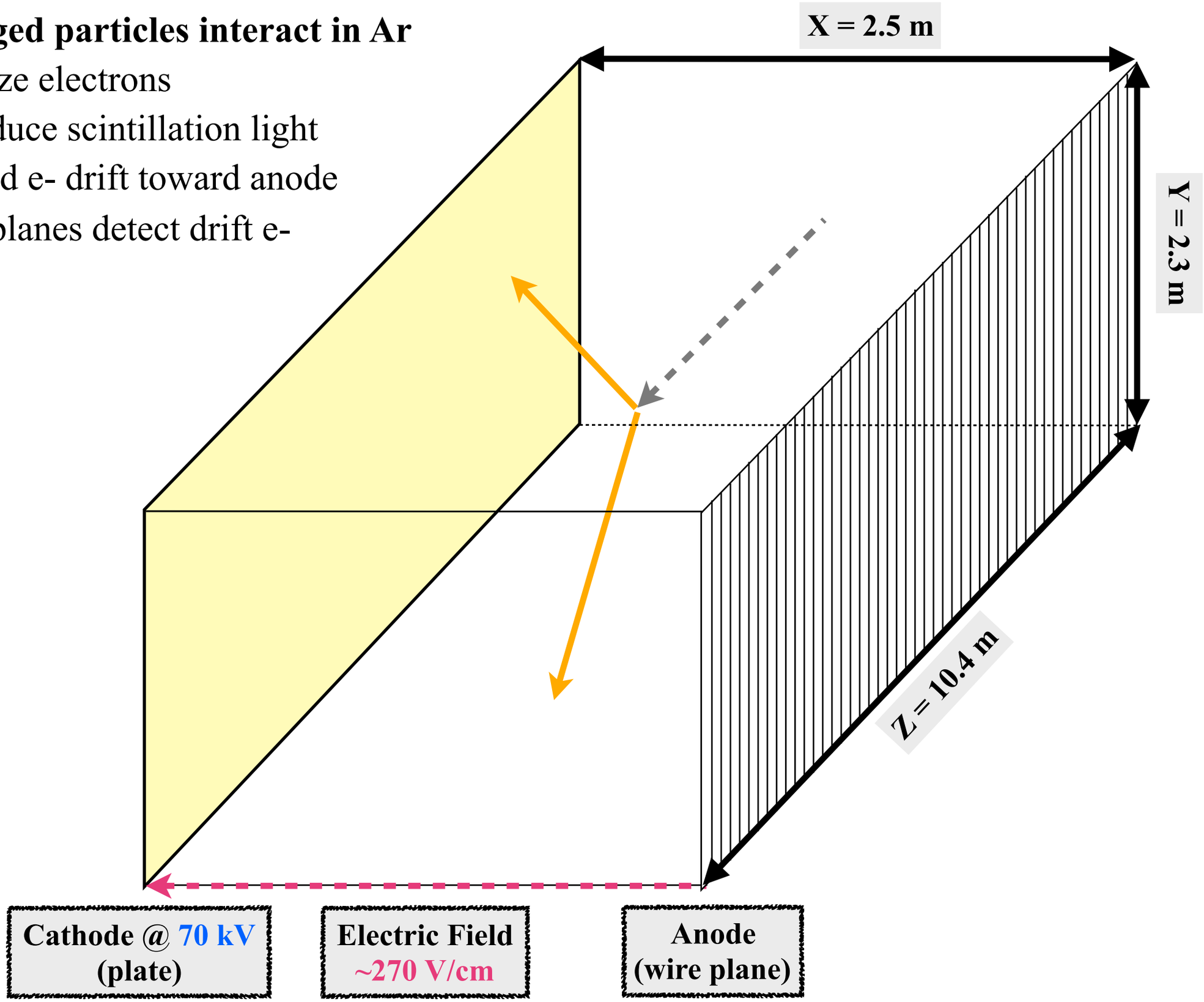
LARTPCS

- Picture of MicroBooNE during construction
- LArTPCs consist of vessel for liquid argon
- And a TPC (the rectangular structure)

TPC WORKING PRINCIPLE

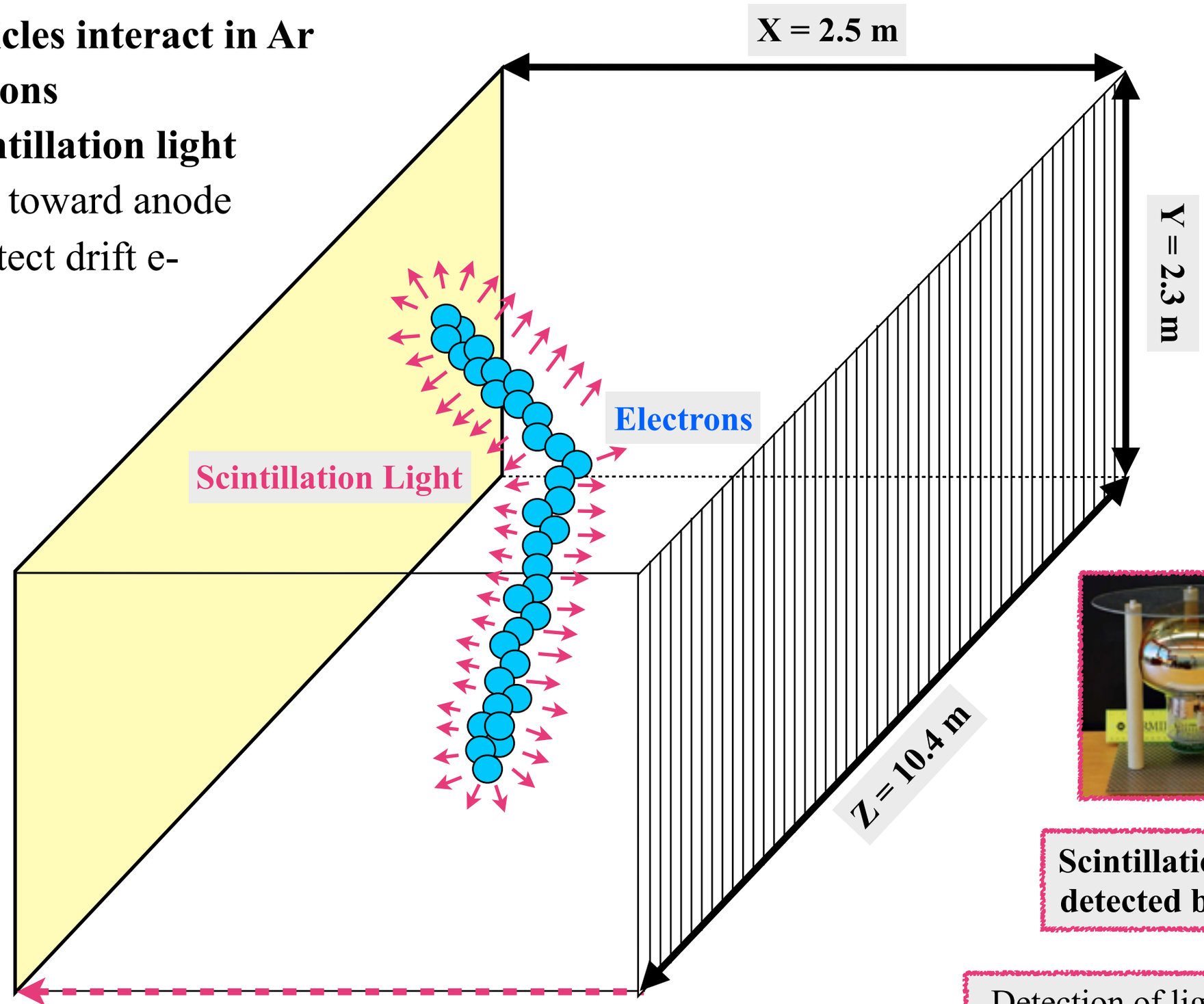
1. Charged particles interact in Ar
- Ionize electrons
- Produce scintillation light

2. Ionized e- drift toward anode
3. Wire planes detect drift e-



TPC WORKING PRINCIPLE

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

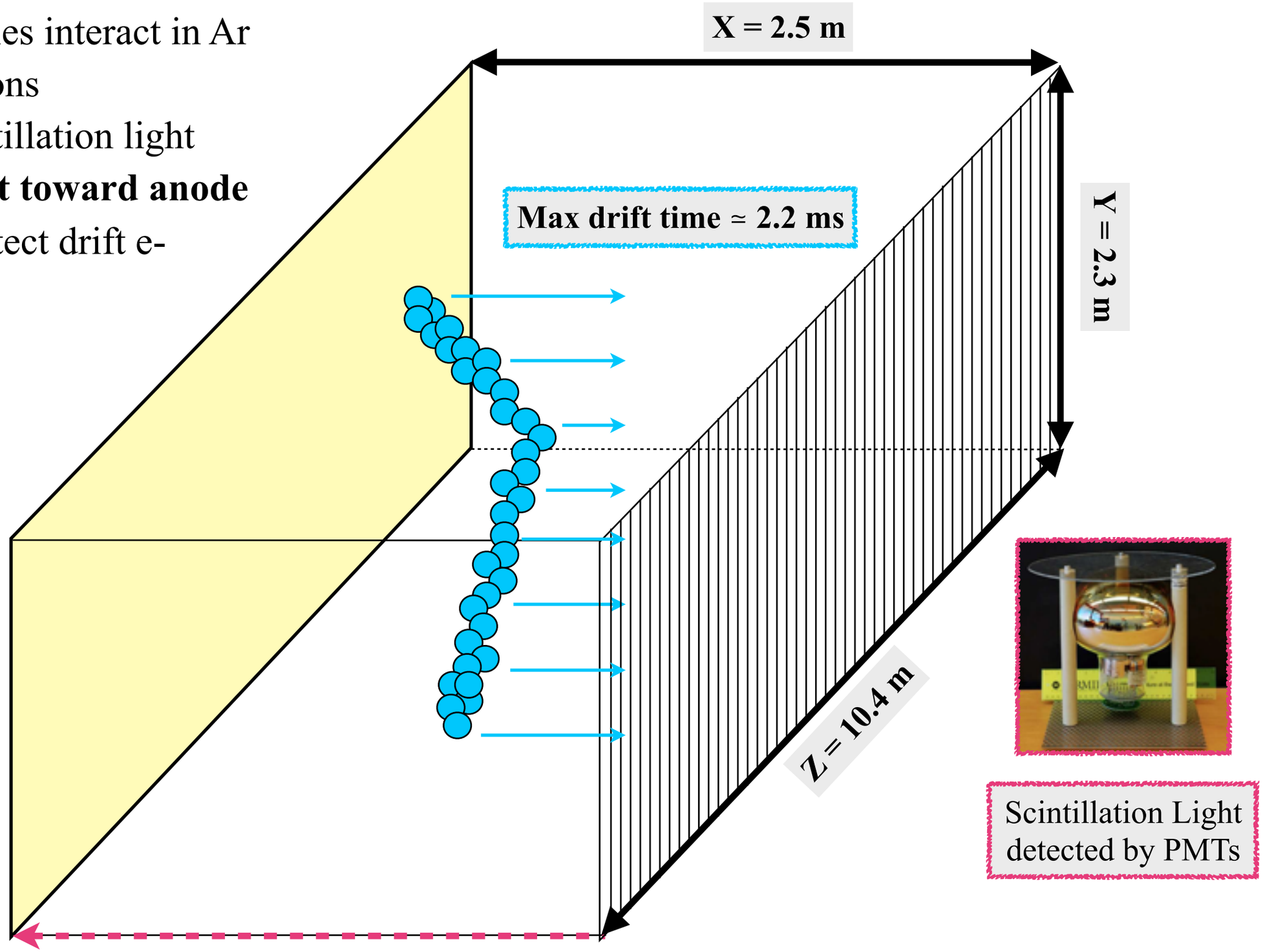


Scintillation Light detected by PMTs

Detection of light defines **TIME** of event

TPC WORKING PRINCIPLE

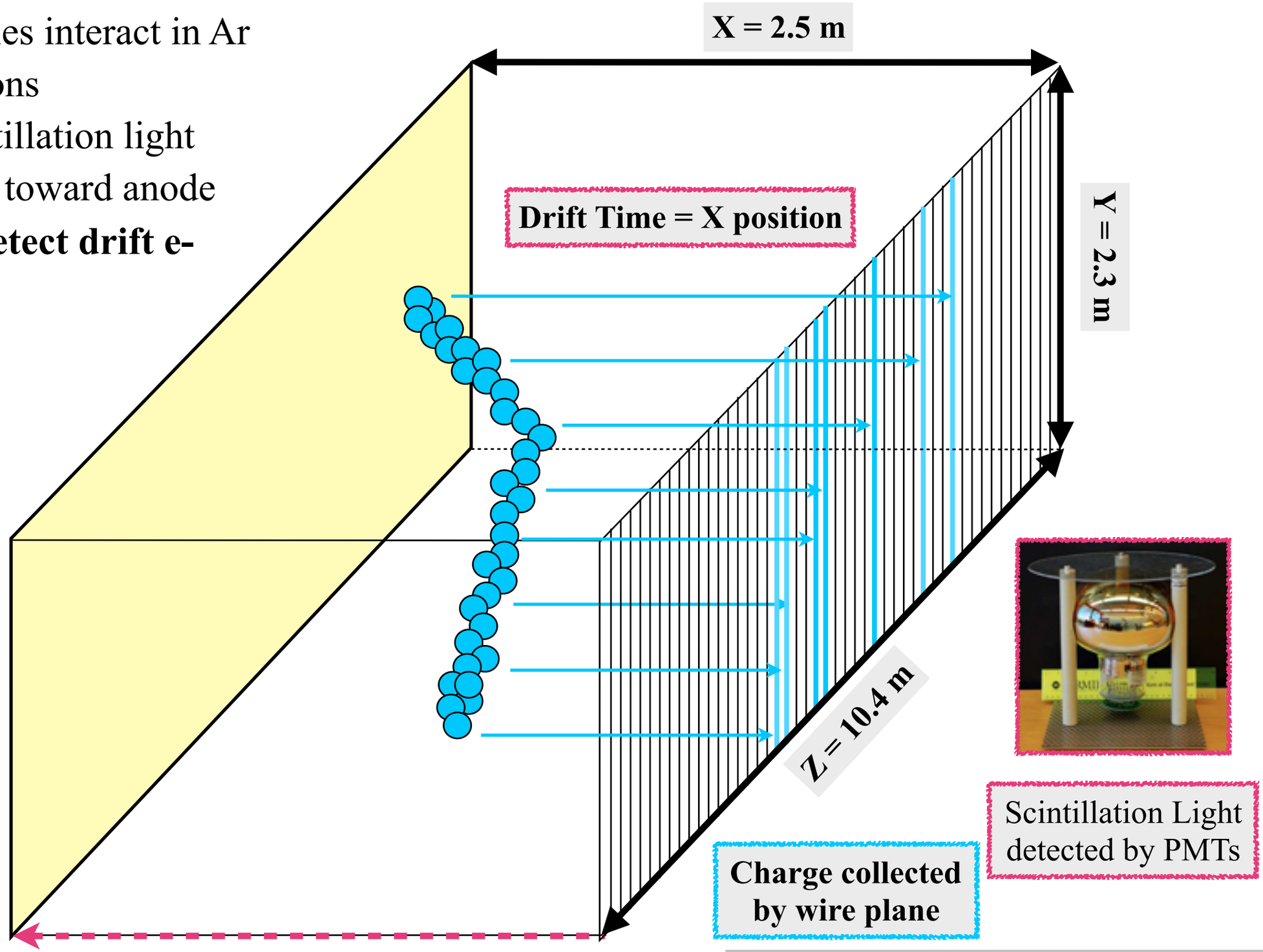
1. Charged particles interact in Ar
   - Ionize electrons
   - Produce scintillation light
2. **Ionized e- drift toward anode**
3. Wire planes detect drift e-



Scintillation Light detected by PMTs

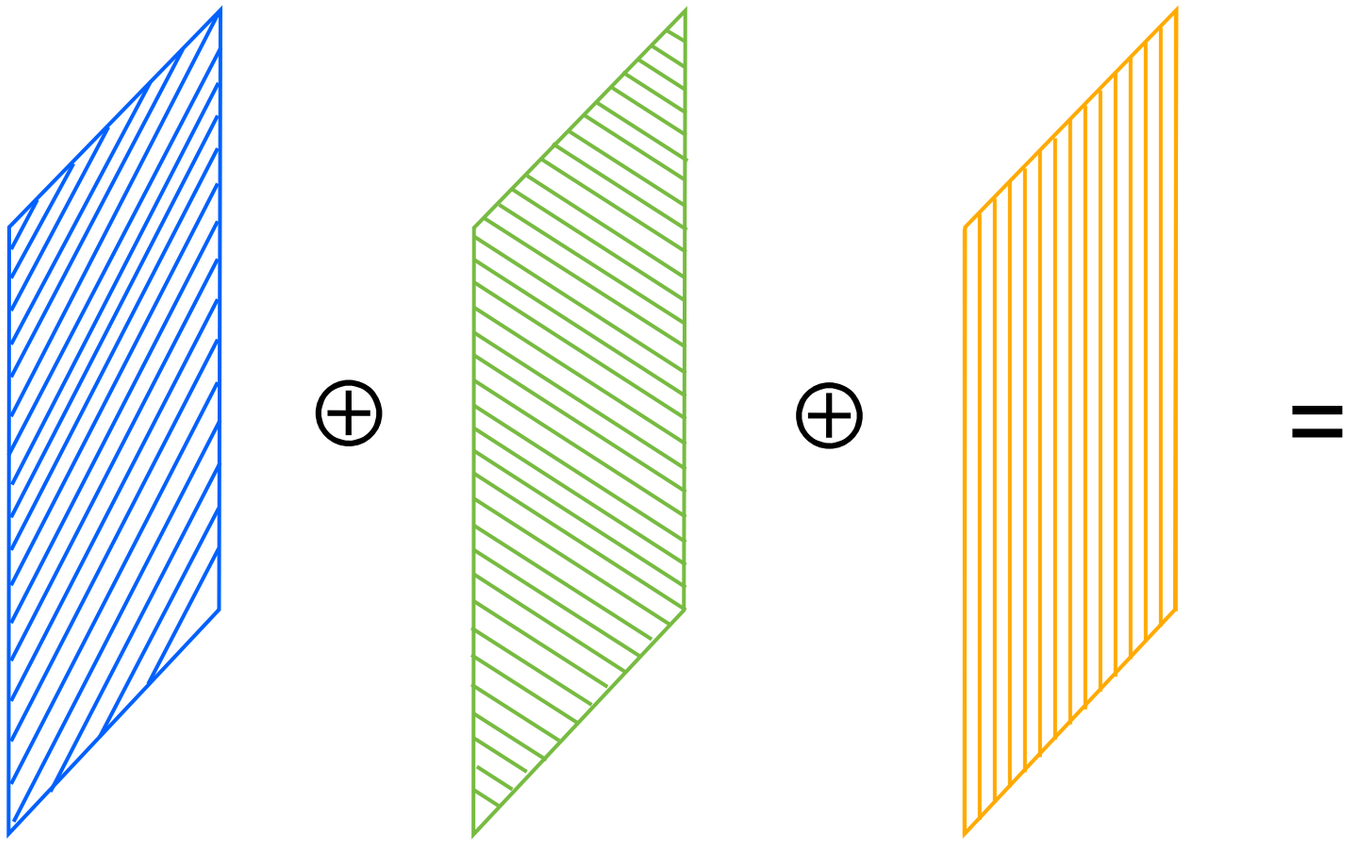
TPC WORKING PRINCIPLE

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



"Cold" (in LAr) readout electronics greatly reduces the noise level

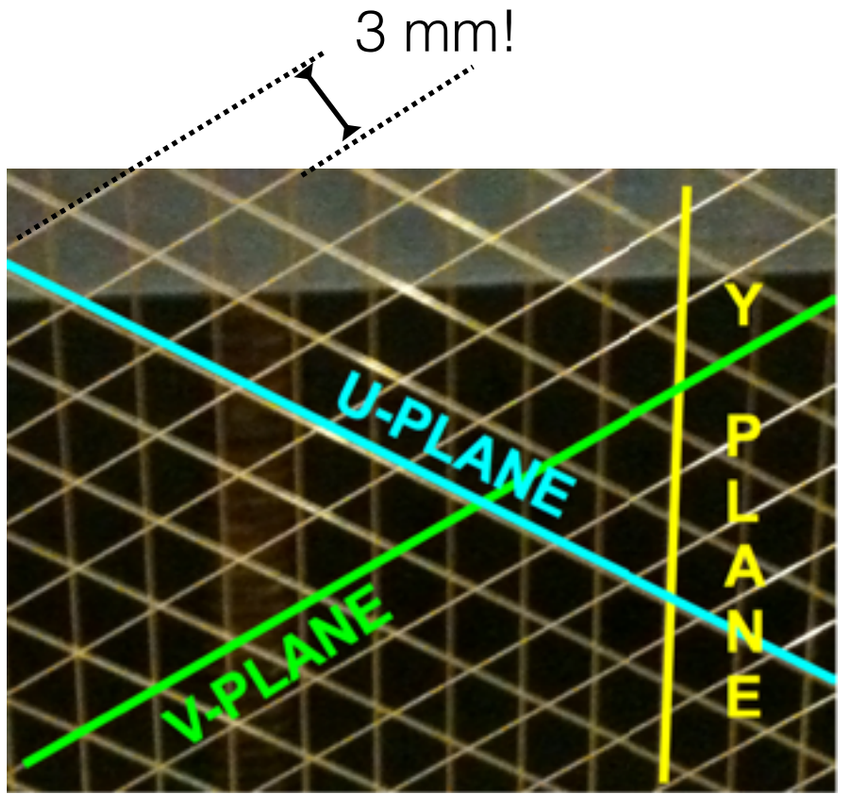
Three Wire Planes



U plane (induction)

V plane (induction)

Y plane (collection)



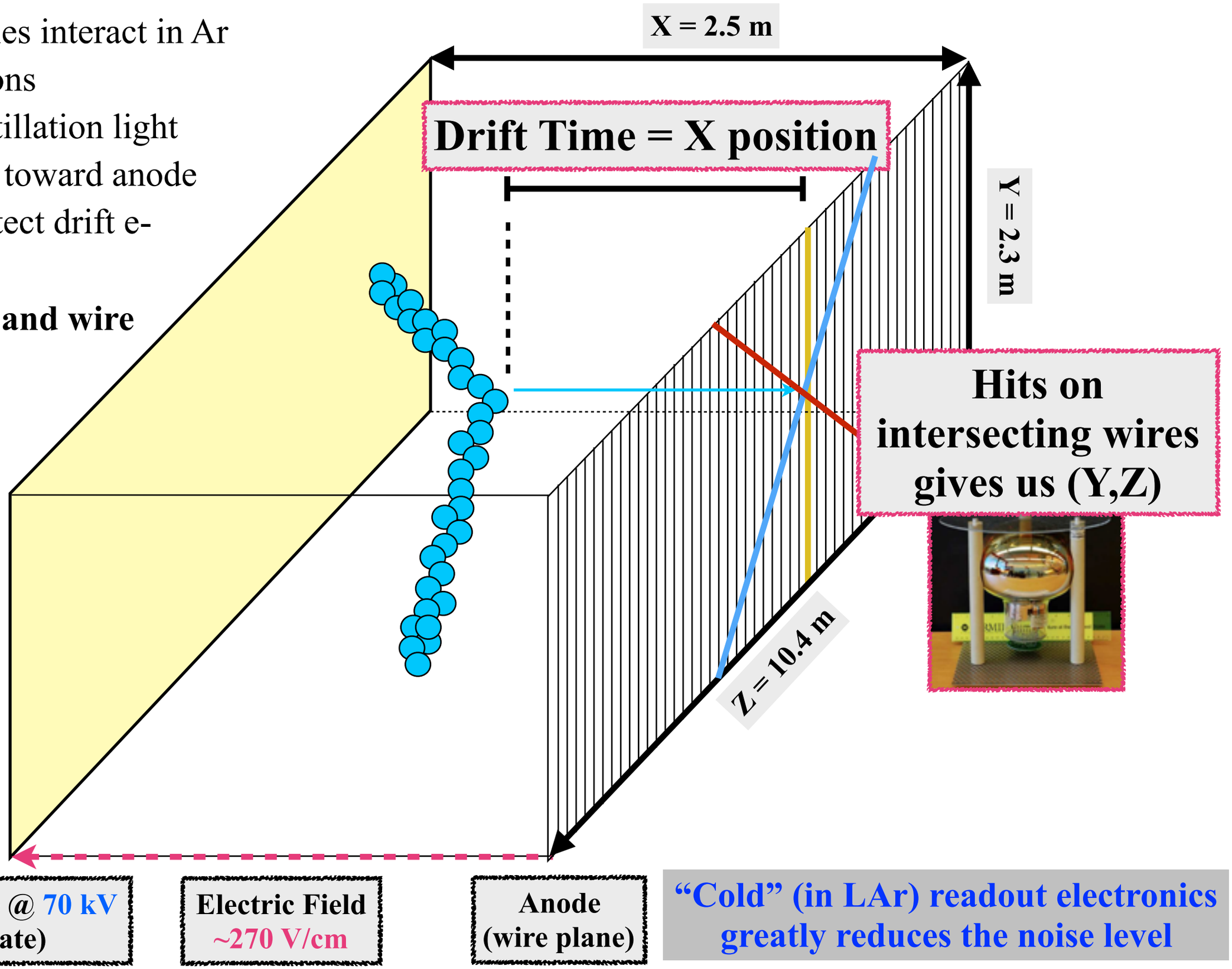
8256 wires w/ pitch = 3mm

(Y, Z) = coincidence on wire

TPC WORKING PRINCIPLE

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

Combine timing and wire info. to get 3D reconstruction.



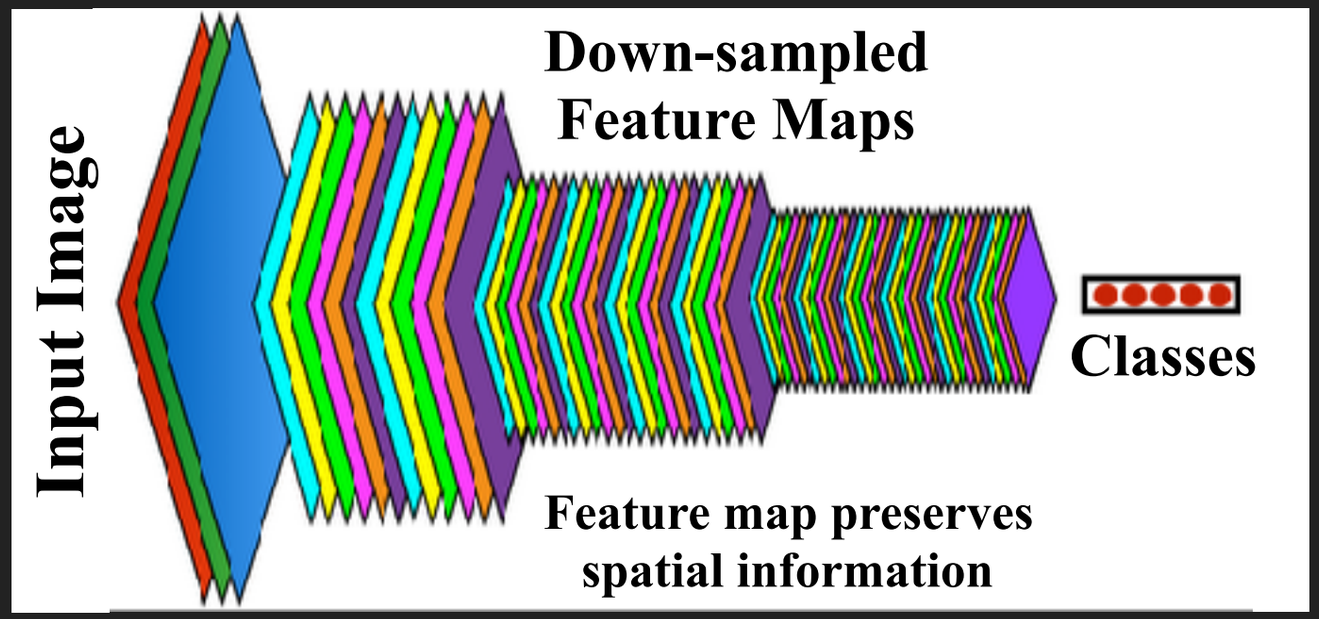
"Cold" (in LAr) readout electronics greatly reduces the noise level

CNN DETAILS

	Classified Particle Type				
Image, Network	e^- [%]	γ [%]	μ^- [%]	π^- [%]	proton [%]
HiRes, AlexNet	73.6 ± 0.7	81.3 ± 0.6	84.8 ± 0.6	73.1 ± 0.7	87.2 ± 0.5
LoRes, AlexNet	64.1 ± 0.8	77.3 ± 0.7	75.2 ± 0.7	74.2 ± 0.7	85.8 ± 0.6
HiRes, GoogLeNet	77.8 ± 0.7	83.4 ± 0.6	89.7 ± 0.5	71.0 ± 0.7	91.2 ± 0.5
LoRes, GoogLeNet	74.0 ± 0.7	74.0 ± 0.7	84.1 ± 0.6	75.2 ± 0.7	84.6 ± 0.6

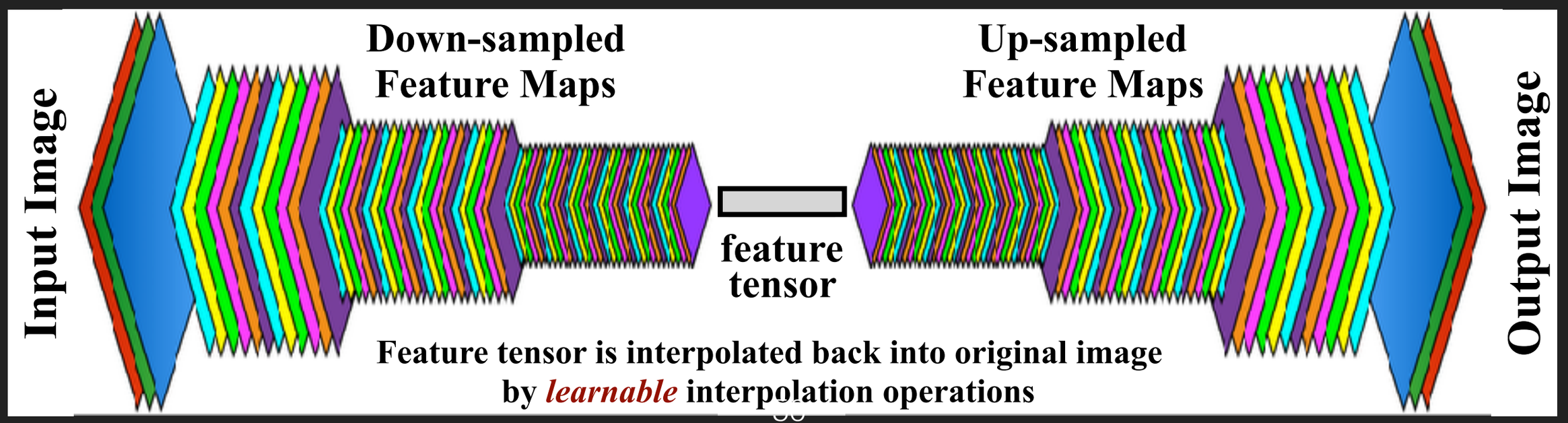
How is it different from *Image Classification*?

Example CNN for Image Classification



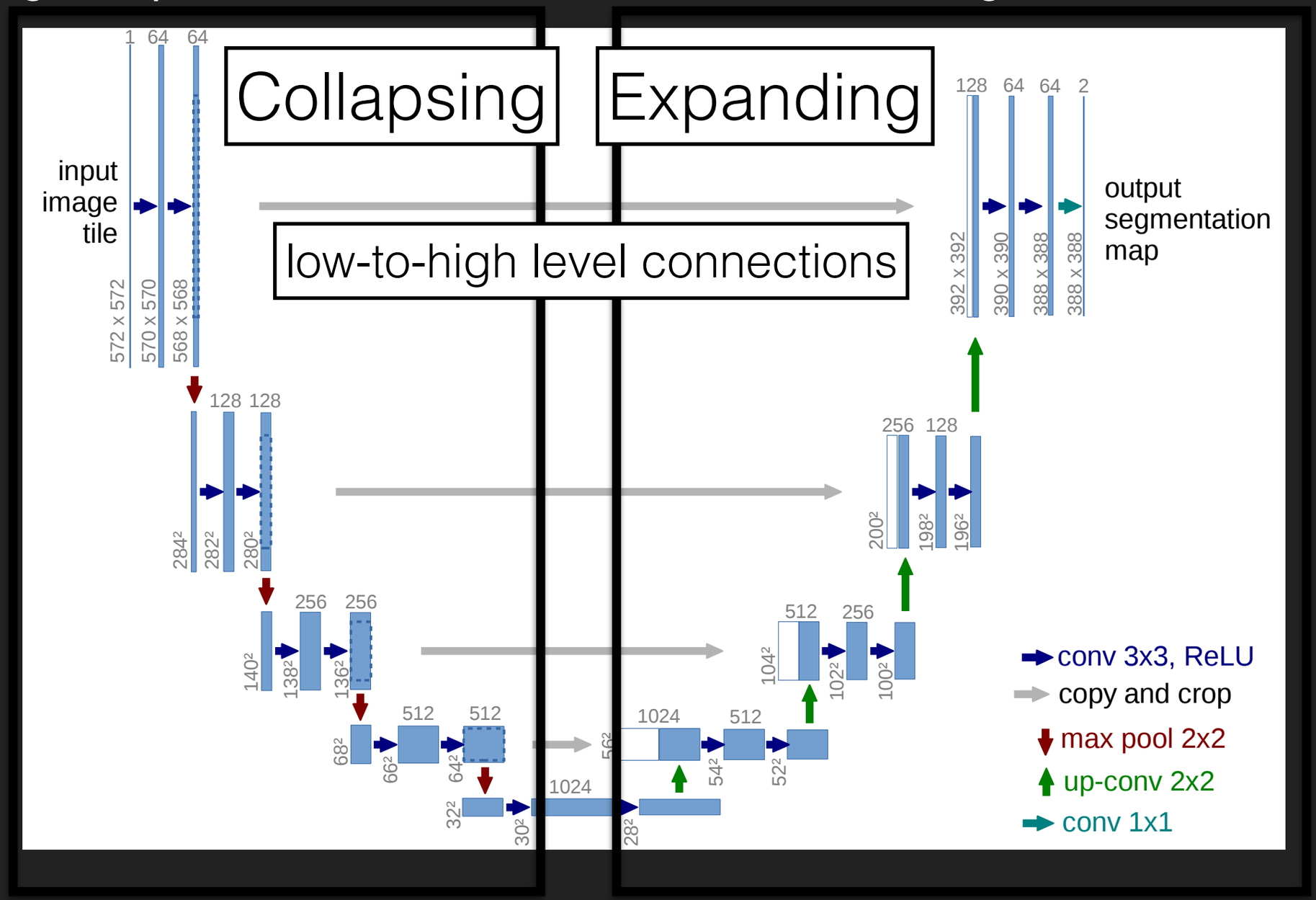
- Classification network reduces the whole image into final "class" 1D aray
- SSNet, after extracting class feature tensor, interpolates back into original image size

Example CNN for Semantic Segmentation



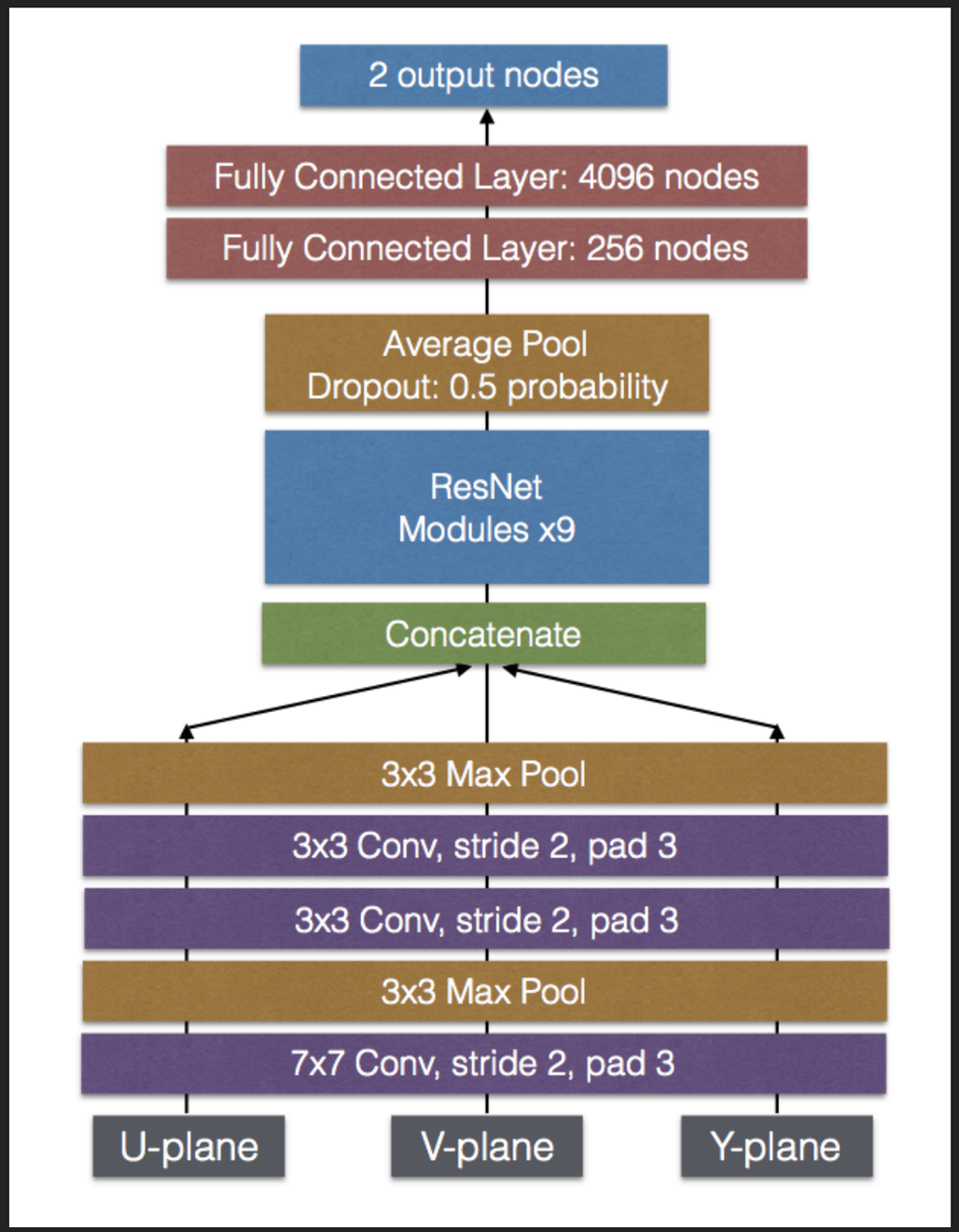
uBoone U-ResNet (or UBURN) Architecture

- U-Net gets it name from its graph diagram: network composed of a collapsing and expanding half, plus connections between low level and high-level feature maps



CLASSIFICATION

- Network used in paper
- Uses ResNet modules
- BatchNorm
- DropOut
- Convolution "stem" (purple and gold) where weights shared across application of 3 views



SYSTEMATICS AWARE TRAINING

Generative Adversarial Networks (GANs)

A GAN is a CNN that takes in a random vector and transforms it into an image. The image produced is then fed through a classifier CNN, which classifies the image as either real or fake.

The goal of a GAN is to produce images that the classifier thinks are real.

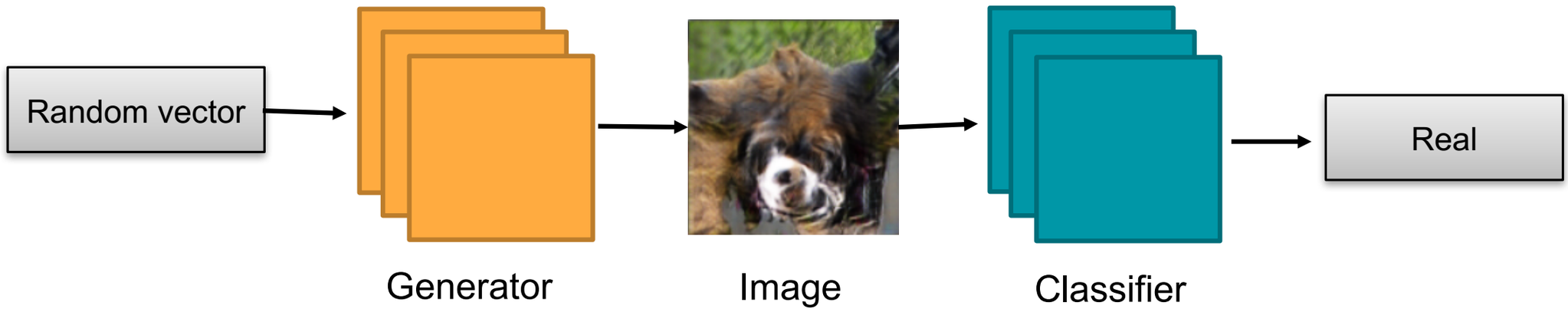
A GAN that uses feature mapping has a modified goal: to produce images that, when fed through the classifier, cause the neurons in the classifier network to activate in the same way as they would when viewing real images.

http://arxiv.org/abs/1511.06434

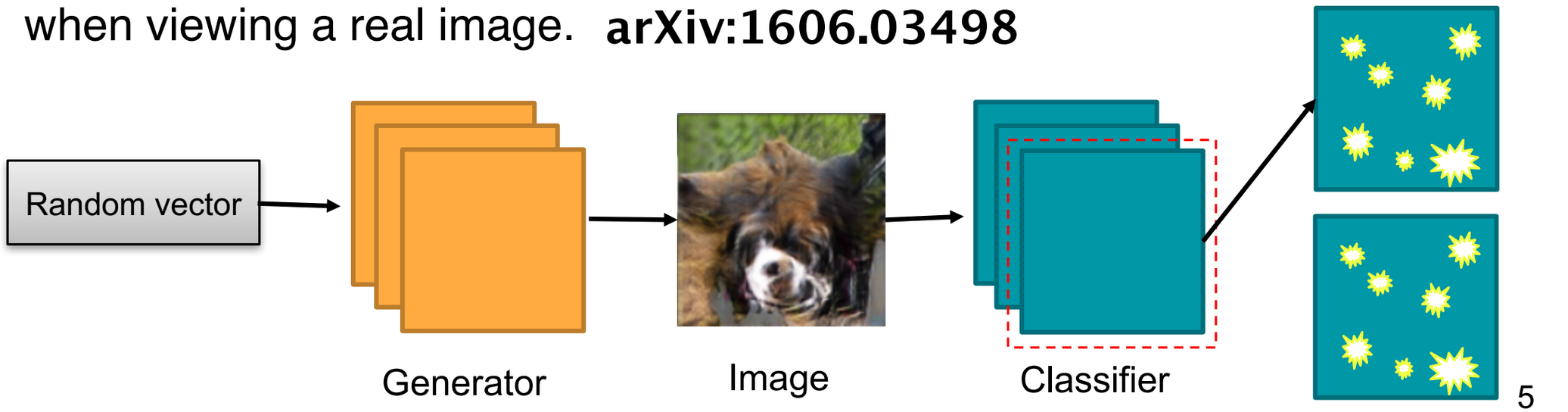
arXiv:1606.03498

Feature Matching in GANs

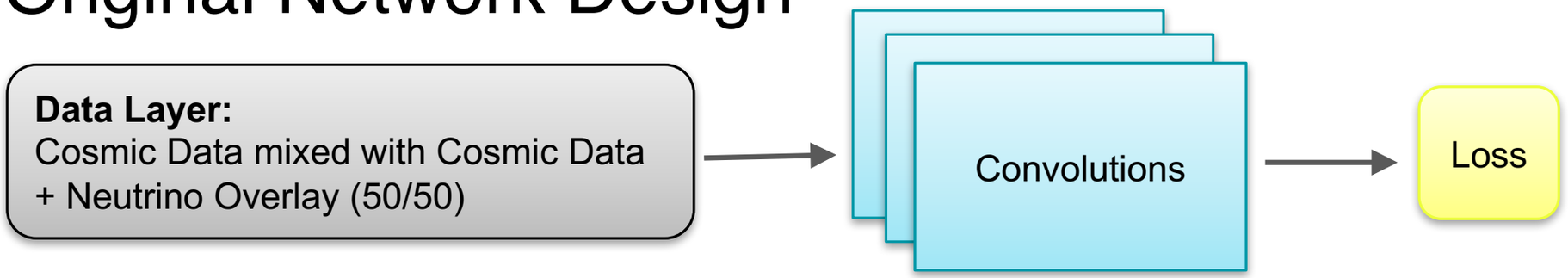
Standard GAN: GAN is rewarded when classifier network classifies the image as real. **arXiv:1511.06434**



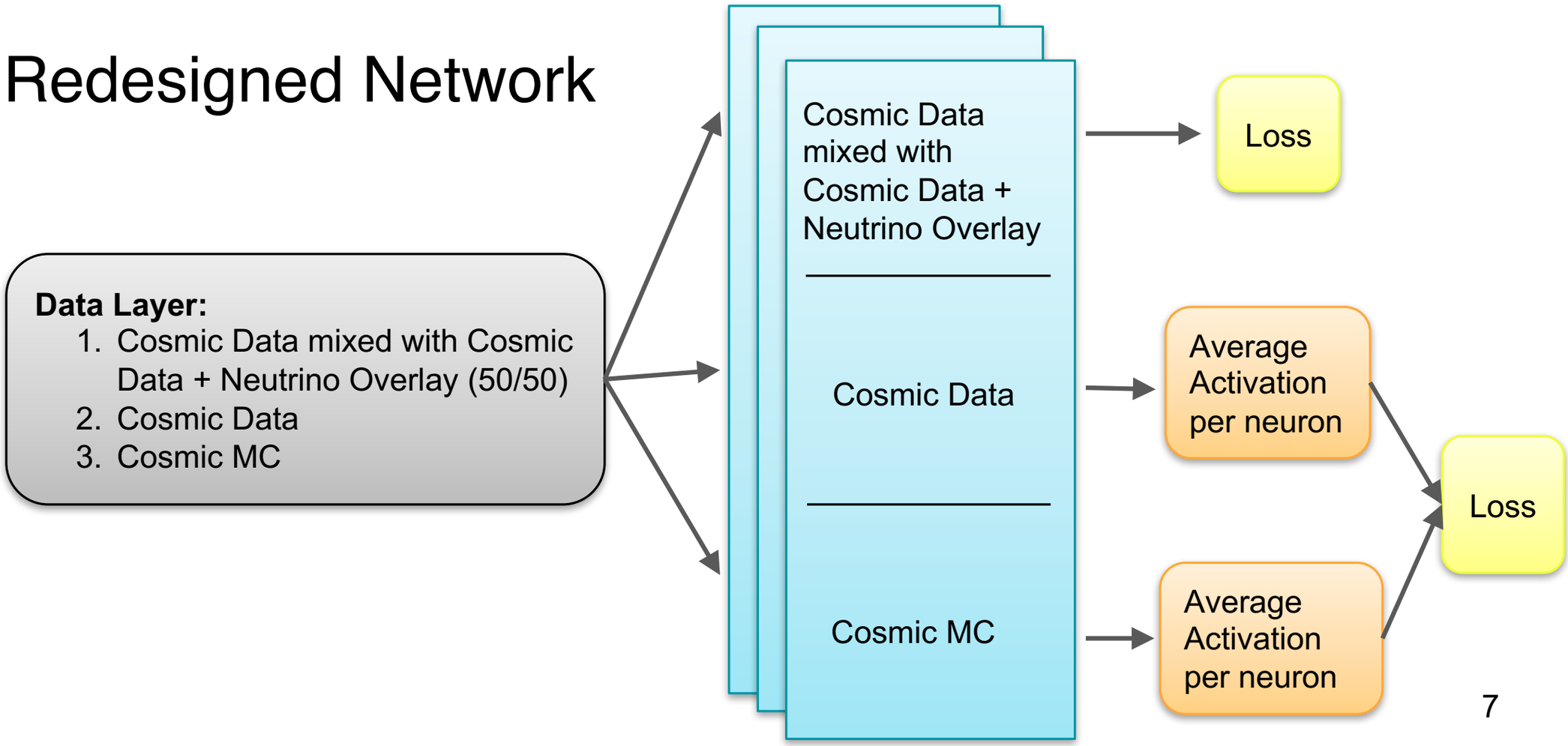
Feature-matching GAN: GAN is rewarded when neurons in an intermediate layer of the classifier network activate in the same way as when viewing a real image. **arXiv:1606.03498**



Original Network Design



Redesigned Network

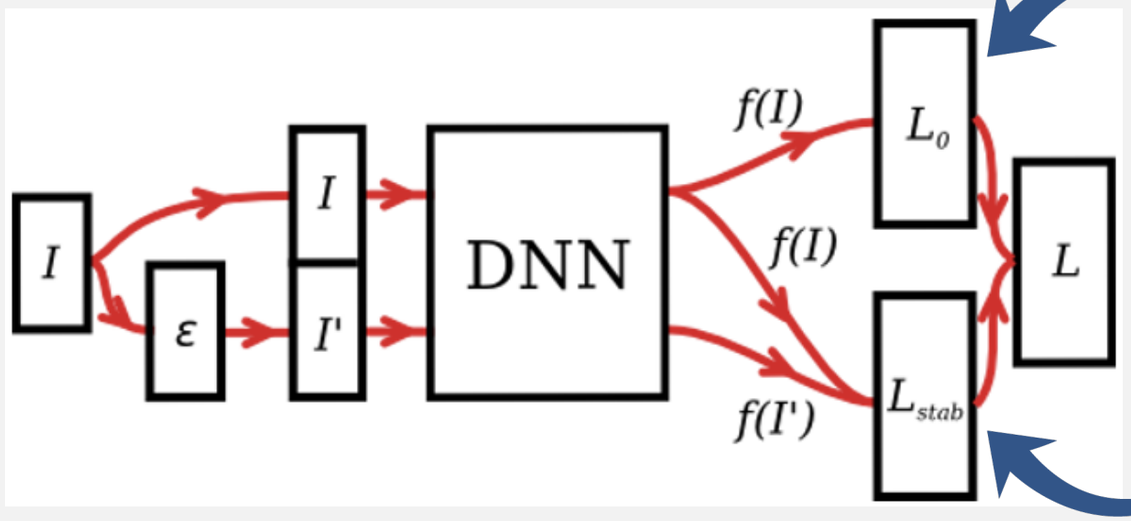


WHAT IS STABILITY TRAINING?

https://arxiv.org/pdf/1604.04326.pdf

- Small perturbations in images can cause large shifts in classification scores
- We modify our loss function with a "Stability Term"
- Run "original image" and "original image plus gaussian noise" and minimize difference in score

$$L_{total} = L_0 + L_{stability}$$



OTHER/FUTURE DIRECTIONS

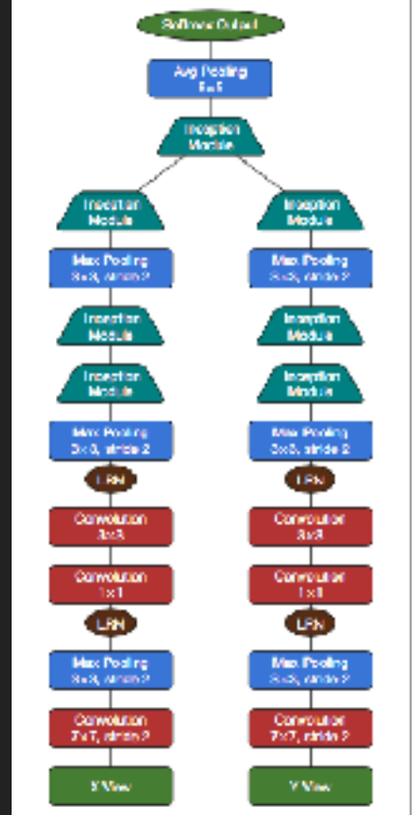
OTHER DIRECTIONS

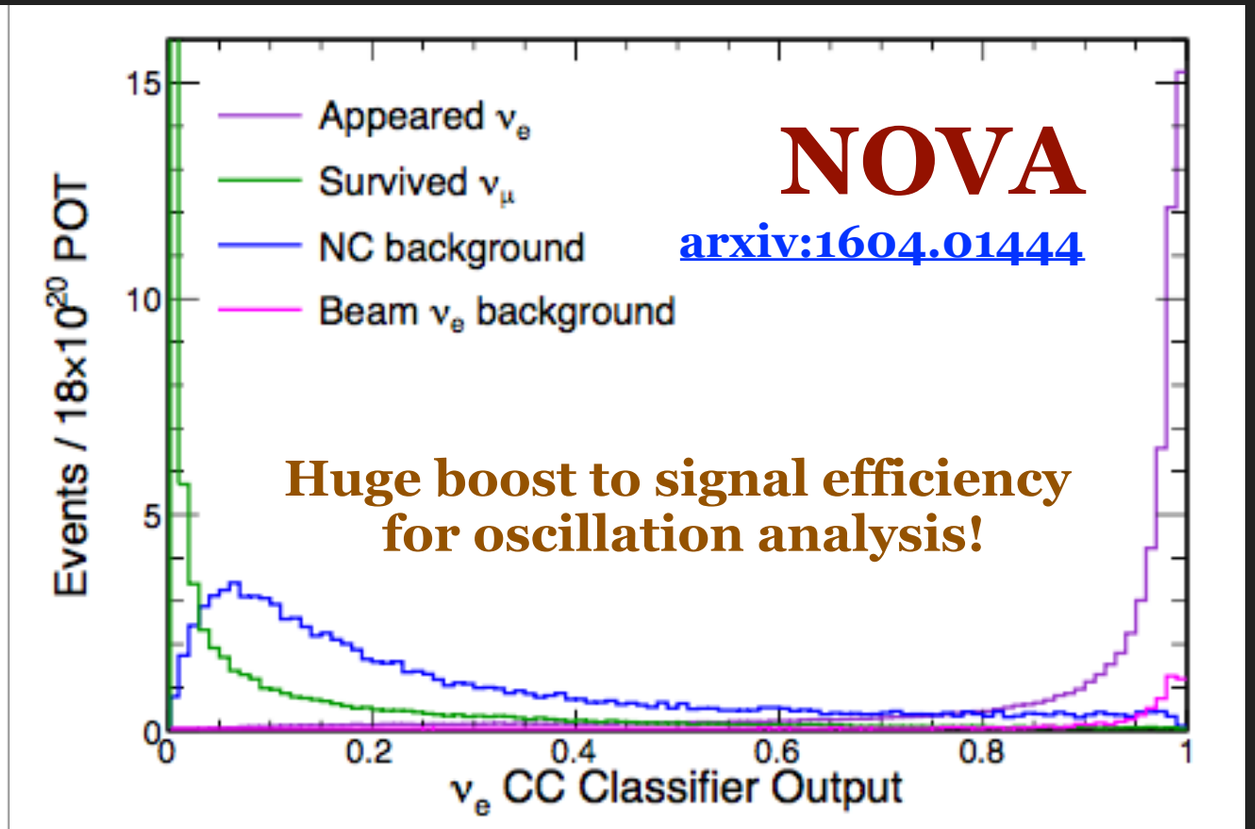
- Other directions beyond augmenting/complementing traditional reconstruction algorithms
  - Not use simulated training data to classify (unsupervised techniques)
    - Anomoly detection
  - Fast, approximate simulation
  - Using recurrent neural networks to analyze time sequence of images

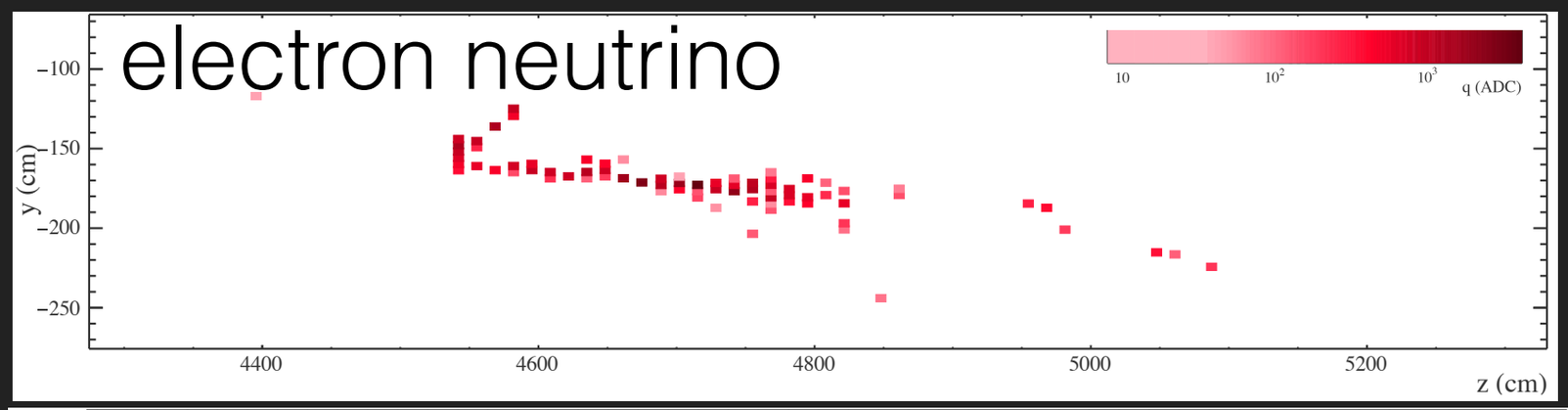
INTERACTION CLASSIFICATION BY NOVA

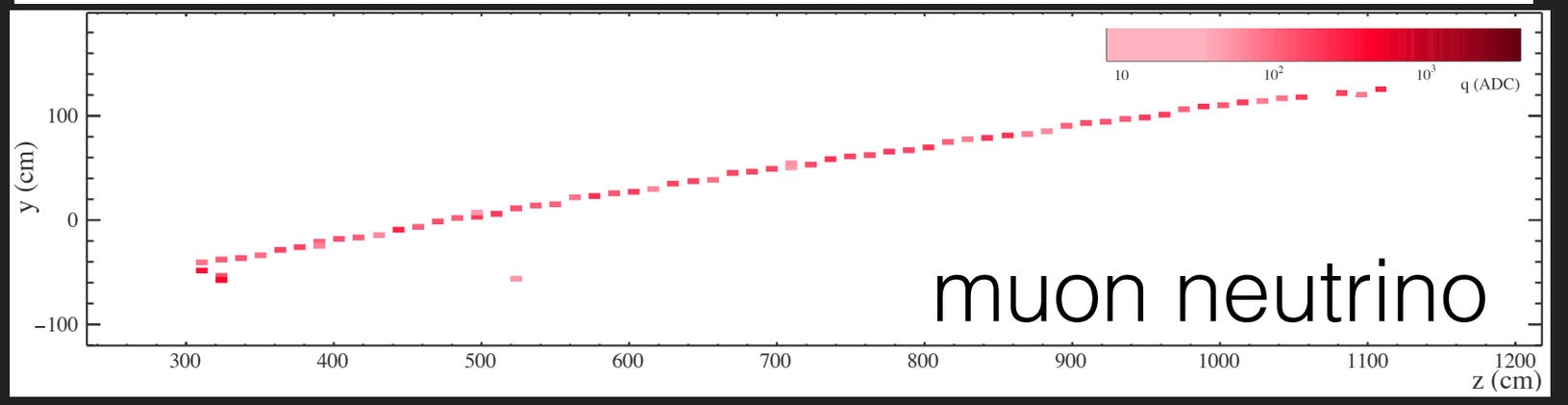
Neutrino Event Classifier

Instead of targeting single particles, classified the interaction as coming from electron or muon neutrino



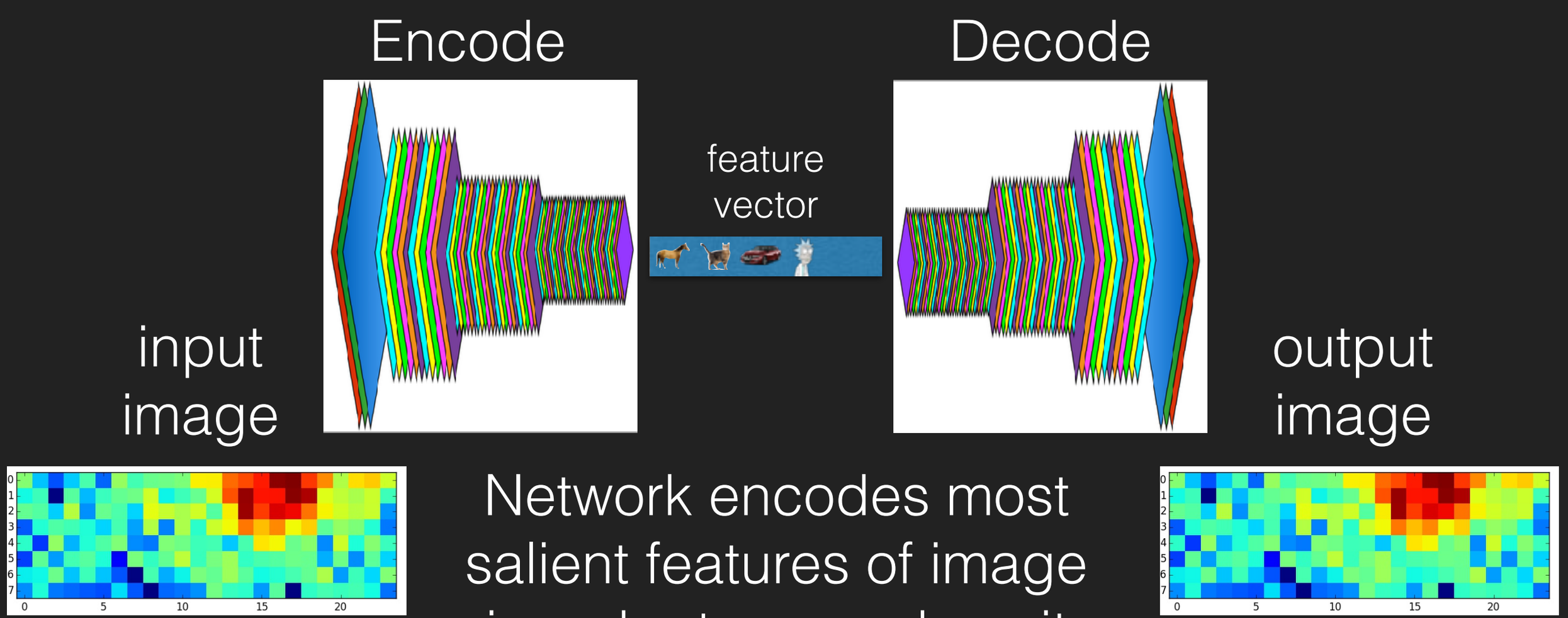






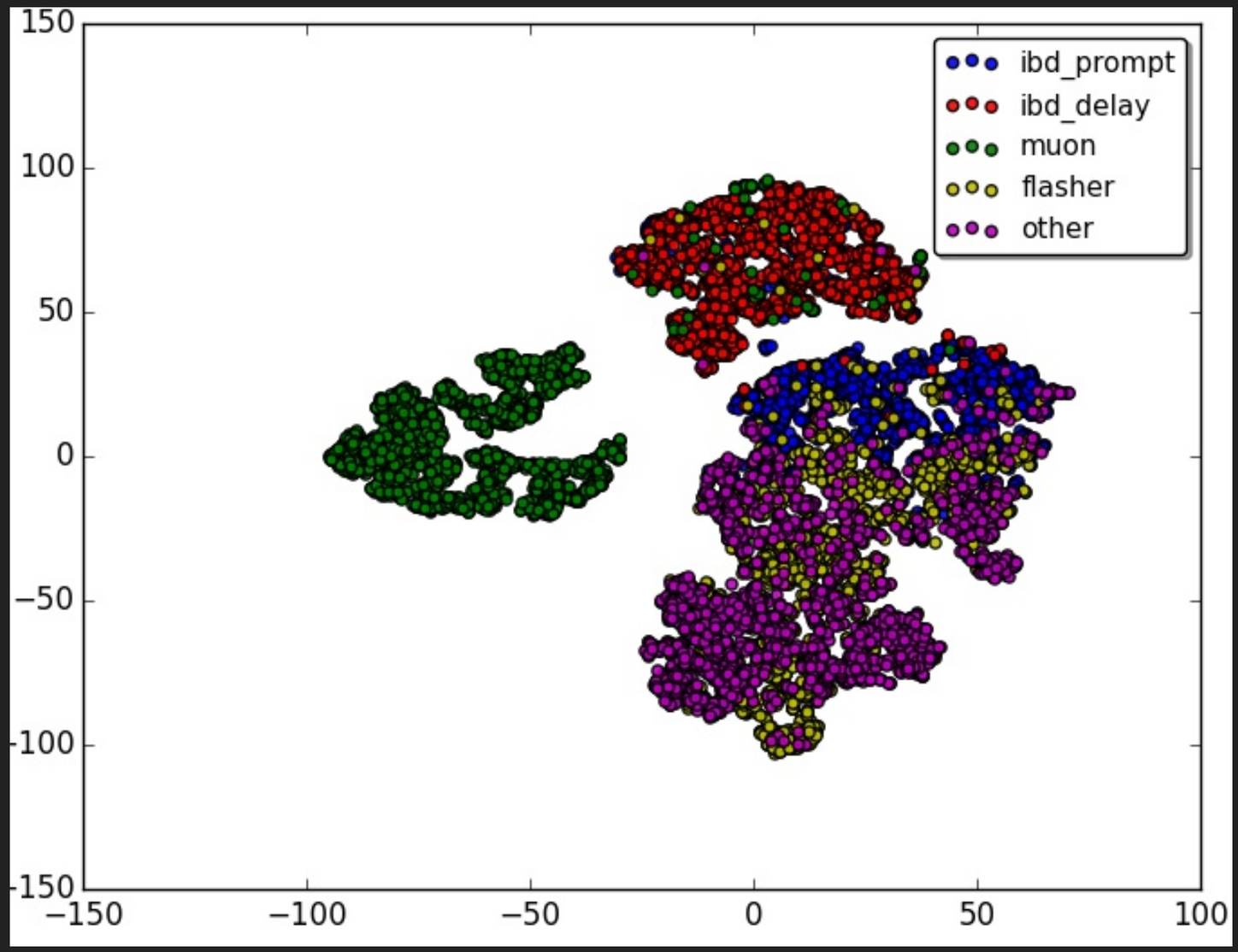
UNSUPERVISED LEARNING ON DAYA BAY

- Unsupervised learning: networks can learn to separate data into distinct groups without labels
- Use an auto-encoder network (similar to SSNet)



UNSUPERVISED LEARNING ON DAYA BAY

- t-SNE plot of feature vector – a way to visualize the higher-dimensional space in 2D, emphasizing clusters of events
- Network, without labels, clusters distinct event types



GENERATIVE NETWORKS

- One can also train networks to generate data given some parameter vector
- One algorithm is Generative Adversarial Networks: two networks play a game
- Generator network makes images; a discriminator network decides if its real or fake
- Networks play this game until generator can produce images that fool the discriminator

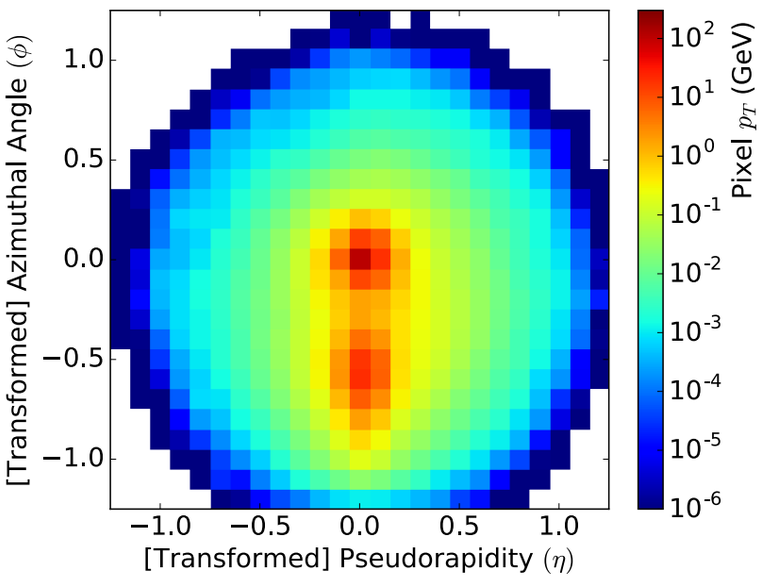
http://arxiv.org/abs/1511.06434

arXiv:1606.03498

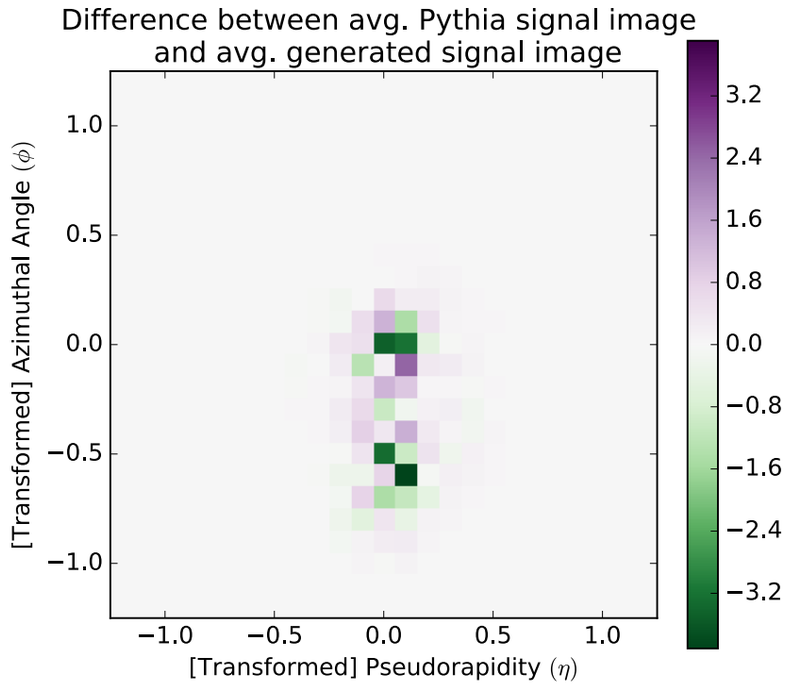
GENERATIVE NETWORKS

- LHC physicists exploring the generation of jet images
- 10 times faster to make GAN image with CPU
- 200 times faster on a GPU
- Aim is to provide a fast-simulation tool

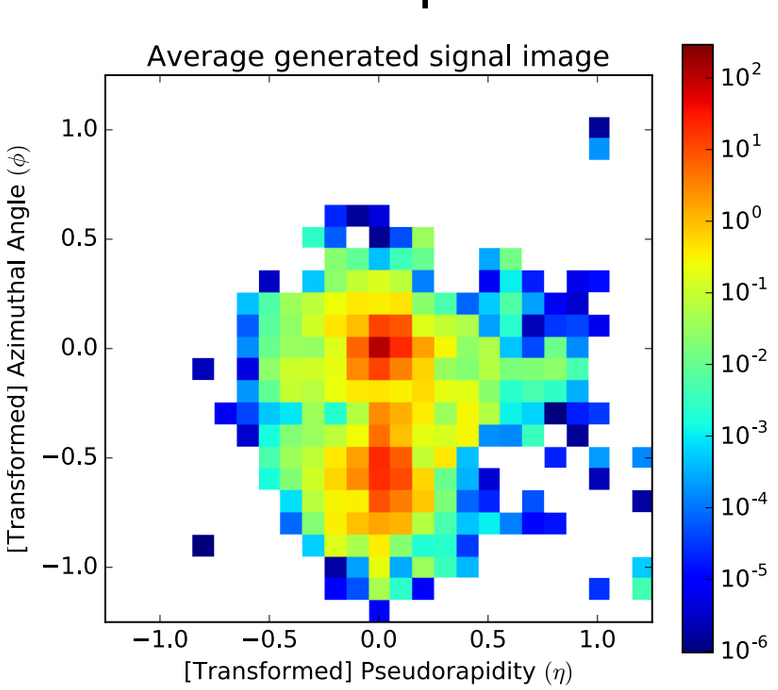
Average full simulation



difference

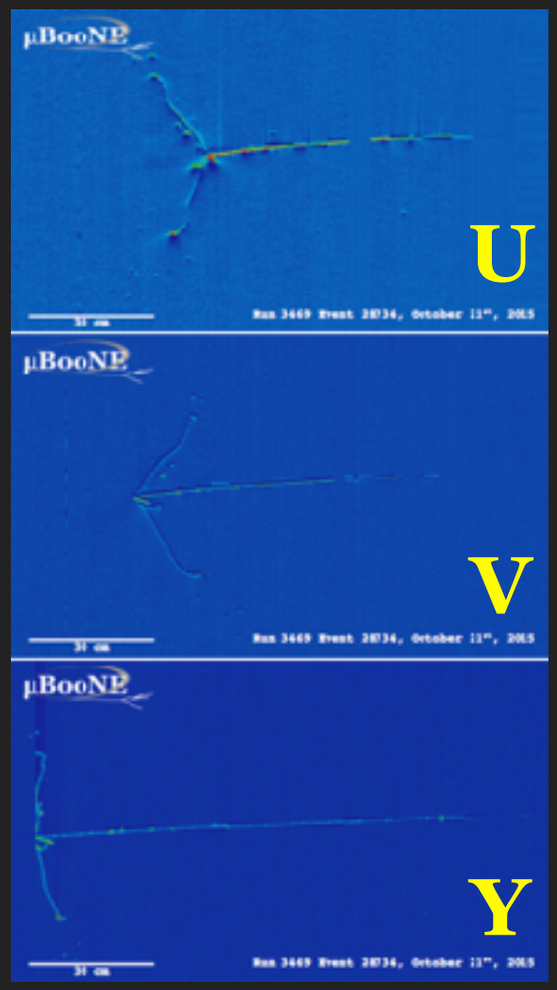


Ave. GAN-produced



https://arxiv.org/pdf/1701.05927.pdf

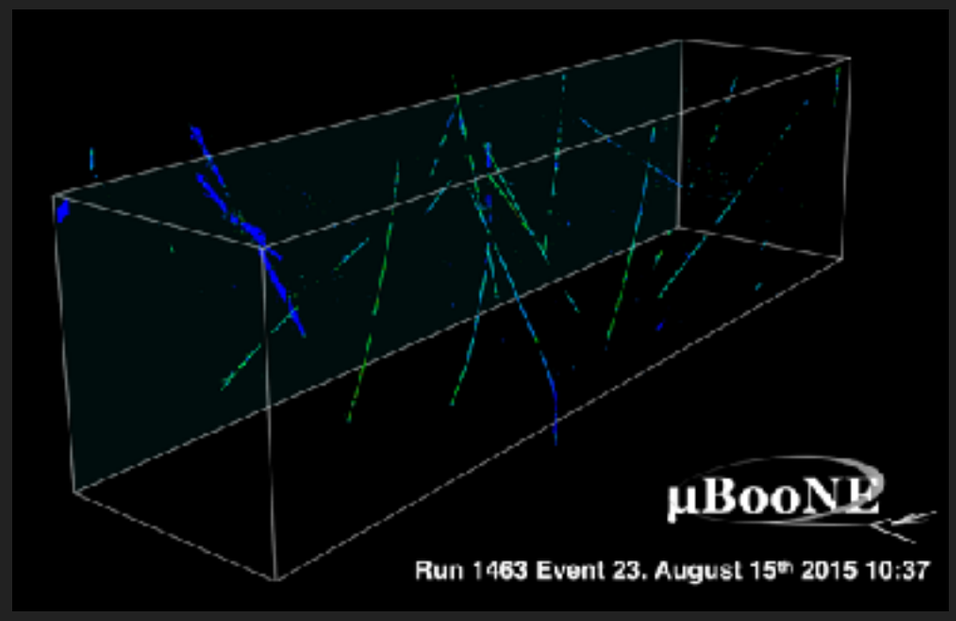
2D TO 3D POSITION RECONSTRUCTION



2D Views

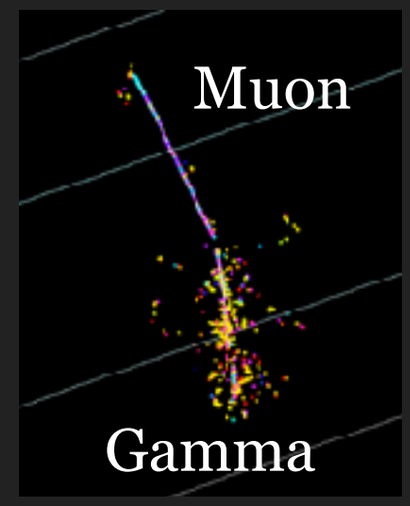
Algorithm based

Network



Reconstructed 3D View

Network



3D Clustering + PID

- Predict 3D point with charge from 2D plane views
- 3D feature recognition (3D point clustering + PID)

OTHER CNN APPLICATIONS

- A wealth of other techniques to try
  - Instance-aware segmentation: cluster individual particles
  - Key-point detection: find vertex, interactions of secondary particles, location of particle decay
  - Structured prediction: for a given type of interaction, find a set of key points and their physically-consistent arrangement

GOING BEYOND WHAT TRADITIONAL RECO. CAN DO

- Such CNN applications are interesting and useful
- But can new DL-techniques allow us to analyze data that has been traditional difficult: high-particle multiplicity events

more particles

more complex interactions

A selection of exclusive final states

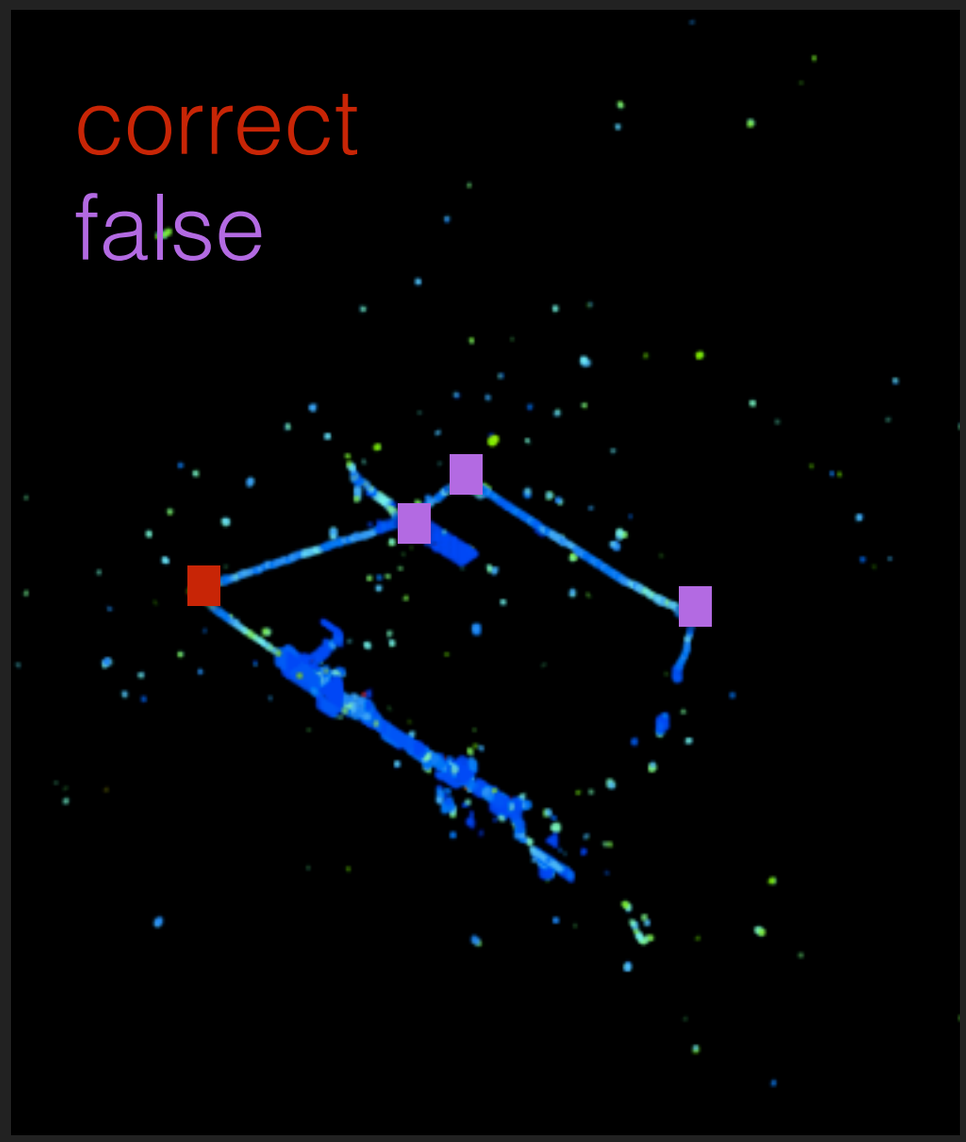
Interaction type	Final state particles				
CCQE	μ	μ+p	μ+2p	μ+3p	μ+4p
NEvents	21,039	14,065	803	176	37
Correct [%]	90.4	84.7	75.3	65.3	62.2
CCRES	μ	μ+p	μ+2p	μ+3p	μ+4p
NEvents	12,401	3,310	2,635	1,112	466
Correct [%]	92.0	78.0	73.0	62.1	51.5
CCRES	$\mu+\pi^+$	$\mu+\pi^++p$	$\mu+\pi^++2p$	$\mu+\pi^++3p$	$\mu+\pi^++4p$
NEvents	8,581	8,416	981	132	45
Correct [%]	75.3	69.3	56.8	40.9	31.1
CCRES	$\mu+\pi^0$	$\mu+\pi^0+p$	$\mu+\pi^0+2p$	$\mu+\pi^0+3p$	$\mu+\pi^0+4p$
NEvents	1,645	3,512	612	91	24
Correct [%]	38.8	35.2	28.8	22.0	16.7
CCDIS	$\mu+\pi^+$	$\mu+\pi^++p$	$\mu+\pi^++2p$	$\mu+\pi^++3p$	$\mu+\pi^++4p$
NEvents	13,397	4,500	968	447	209
Correct [%]	60.7	54.0	50.7	36.9	25.8
CCDIS	$\mu+\pi^0$	$\mu+\pi^0+p$	$\mu+\pi^0+2p$	$\mu+\pi^0+3p$	$\mu+\pi^0+4p$
NEvents	1,872	5,322	959	370	195
Correct [%]	23.6	25.2	23.5	17.0	12.3

← Correct fraction

← nEvents, for context

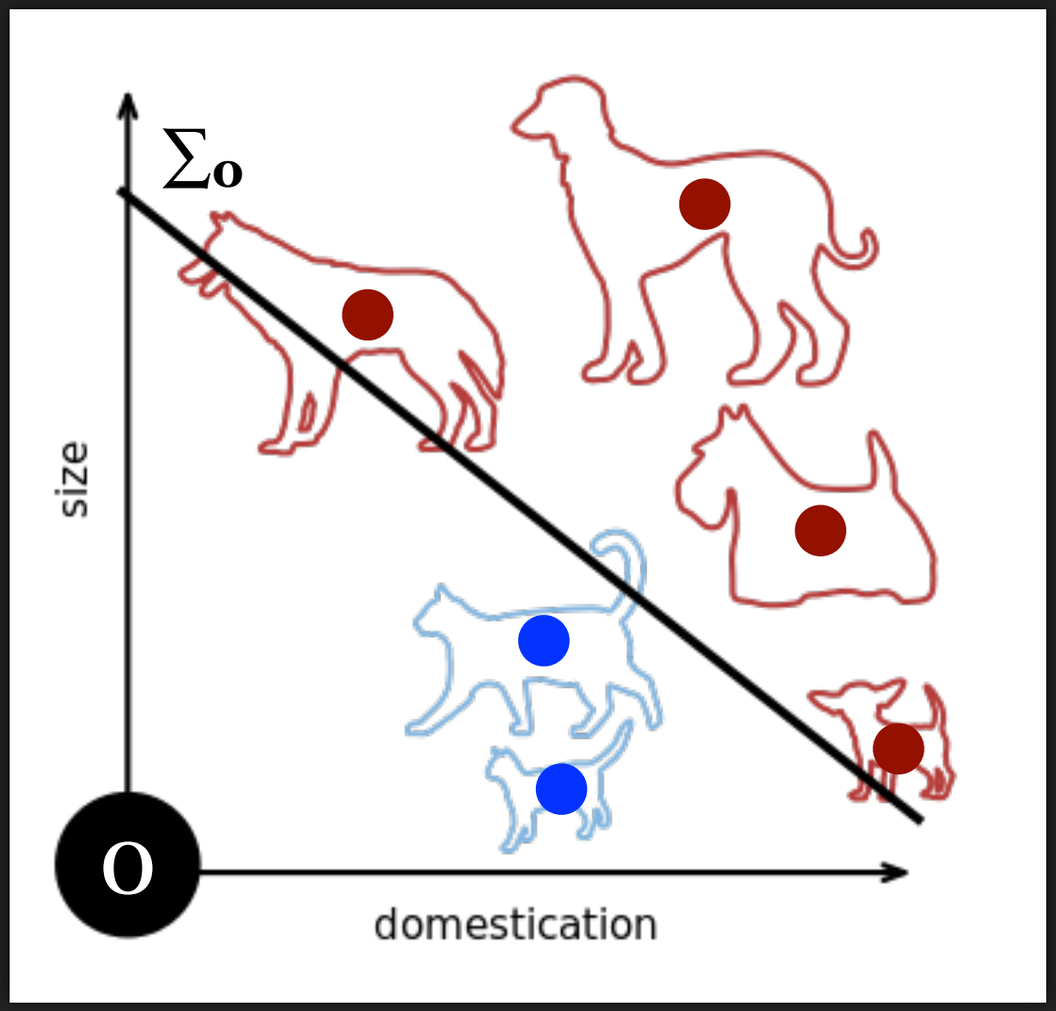
GOING BEYOND WHAT TRADITIONAL RECO CAN DO

- Illustration of potential difficulty
- Key starting point for algorithms is the “vertex”: the point where neutrino interaction occurs
- Build out from there
- One strategy: find location of where most lines intersect
  - But leads us astray here
- More complex interactions require more cases to handle, more candidate solutions to explore – large “state-space” to traverse



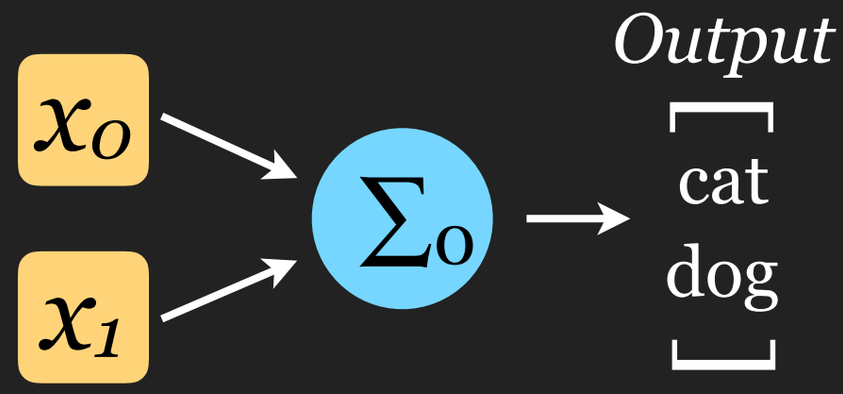
NEURAL NET BACKGROUND

Imagine using two features to separate cats and dogs



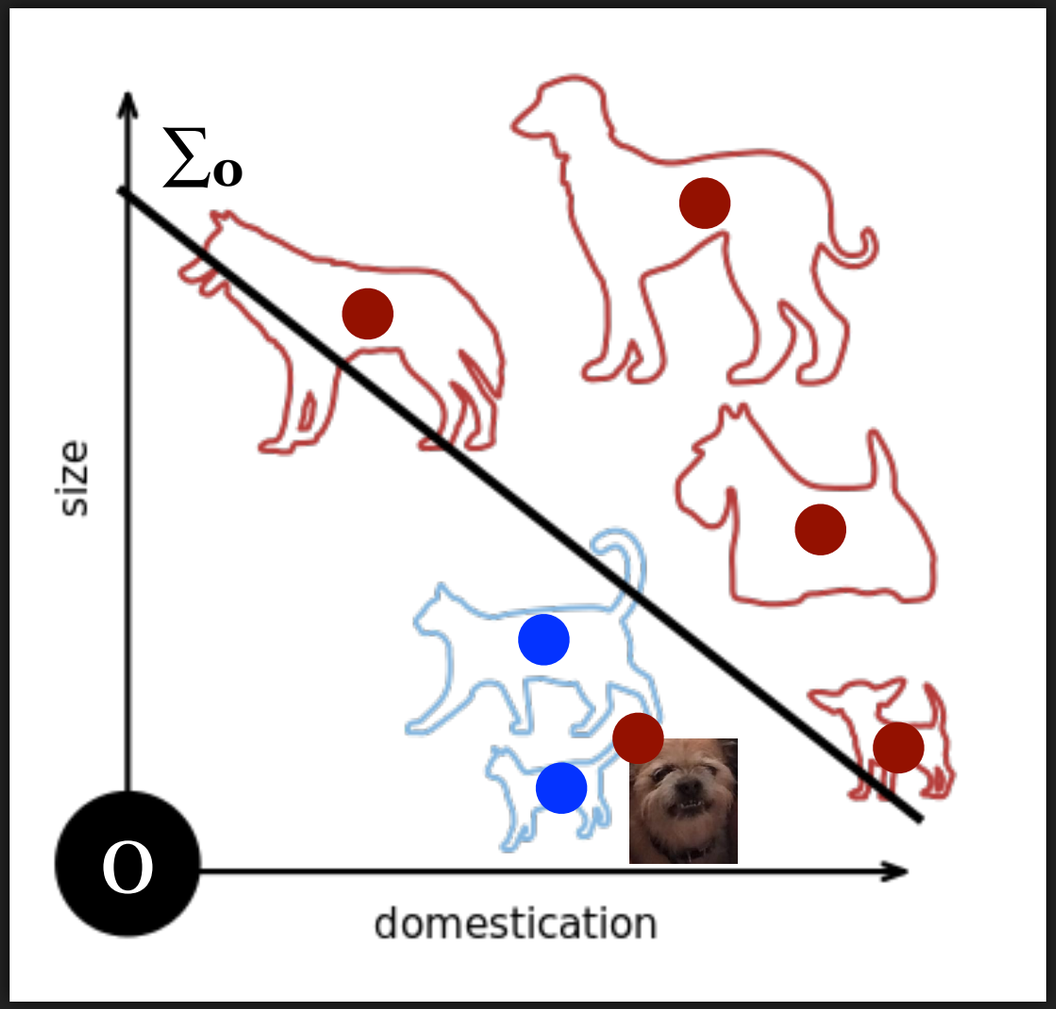
from [wikipedia]

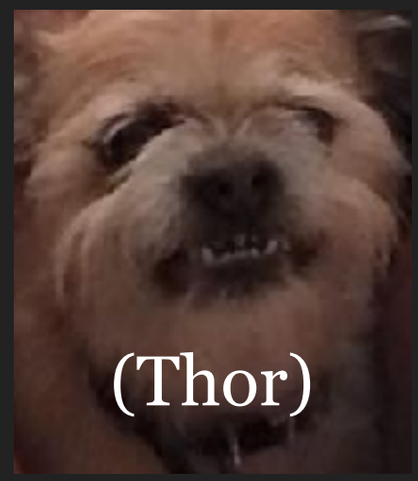
$$\sigma(\vec{x}) = \begin{cases} \vec{w}_i \cdot \vec{x} + b_i & \vec{w}_i \cdot \vec{x} + b_i \geq 0 \\ 0 & \vec{w}_i \cdot \vec{x} + b_i < 0. \end{cases}$$



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

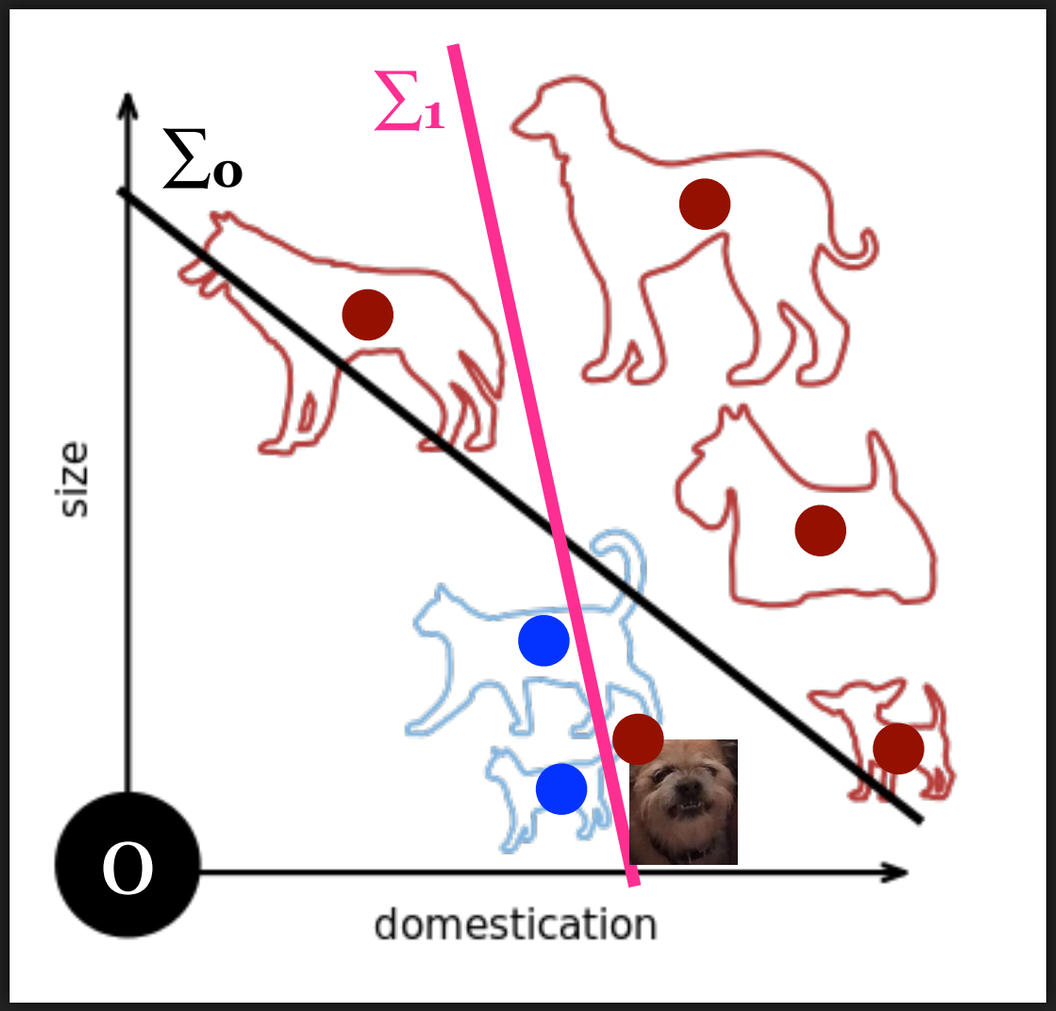
Maybe we need to do better: assume new data point
(My sister's dog — small but not as well behaved)



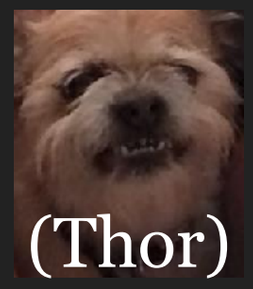


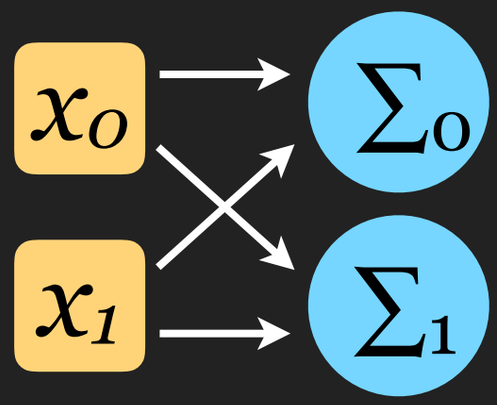
from wikipedia

Maybe we need to do better: assume new data point
(My sister's dog — small but not as well behaved)



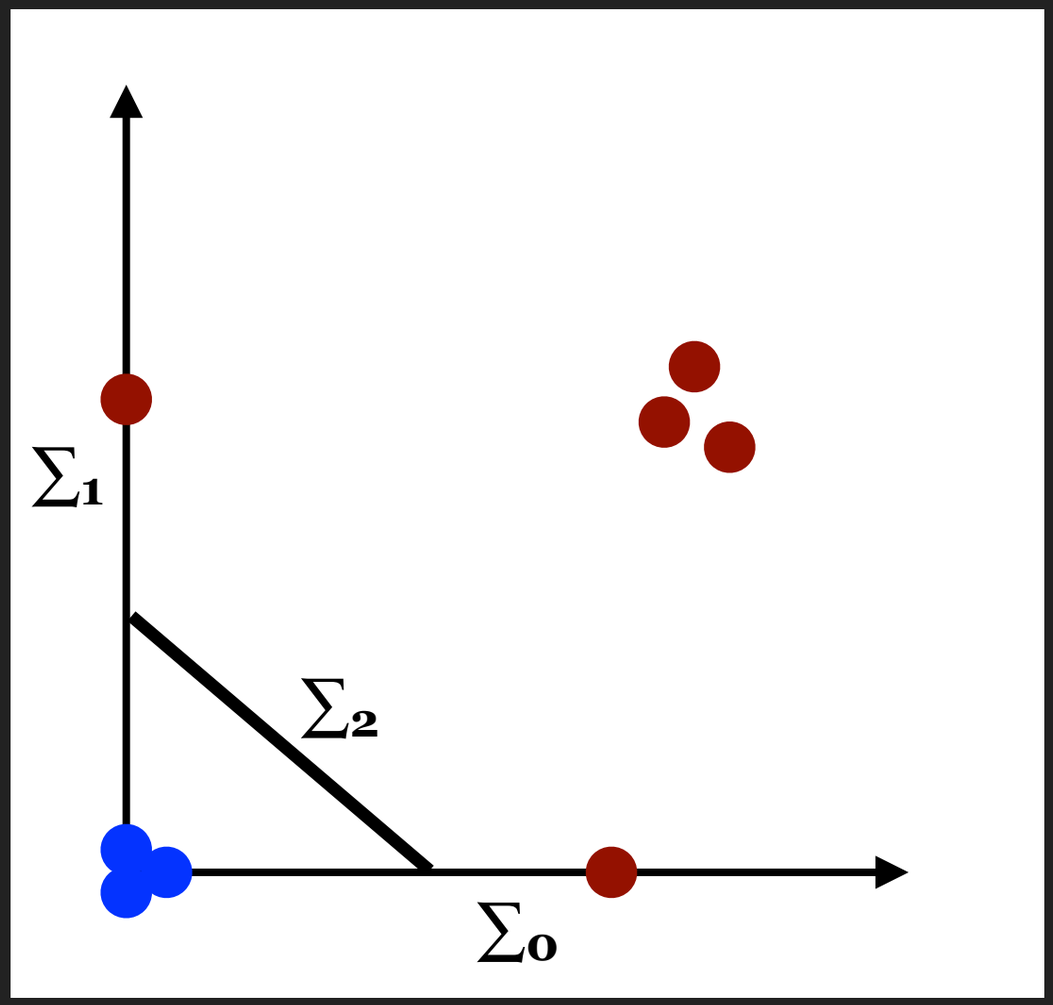
from wikipedia

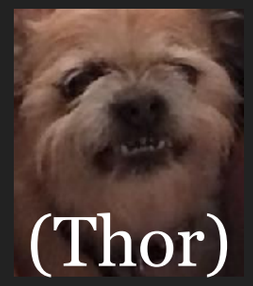

(Thor)

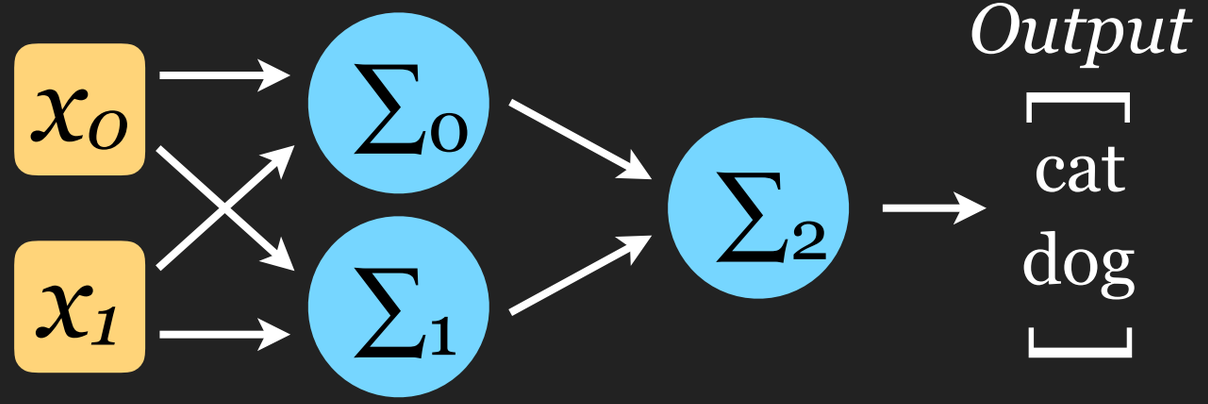


We can add another perceptron
to increase "features"

Maybe we need to do better: assume new data point
(My sister's dog — small but not as well behaved)



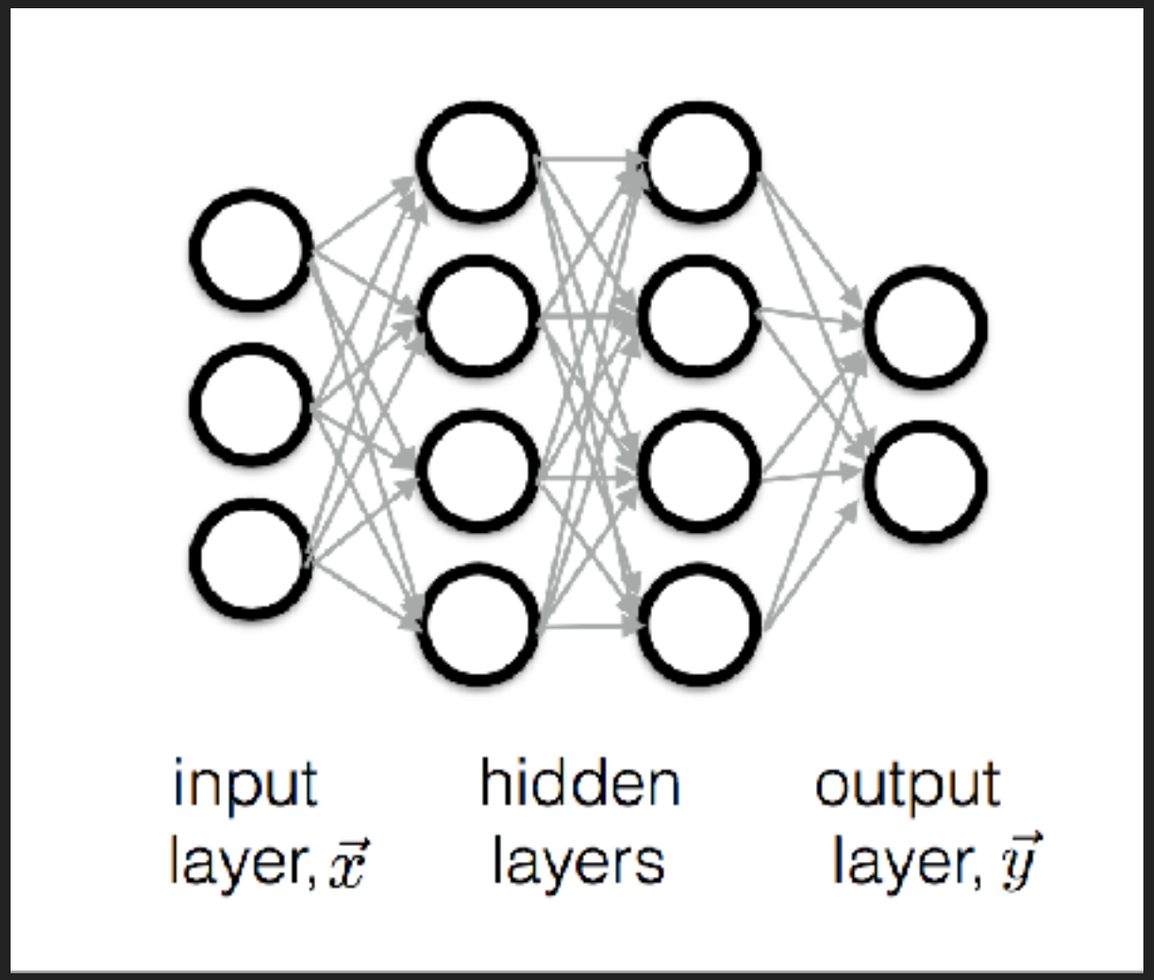

(Thor)



Another layer can classify based on preceding feature layer output

"Traditional neural net" in HEP

Fully-Connected Multi-Layer Perceptrons

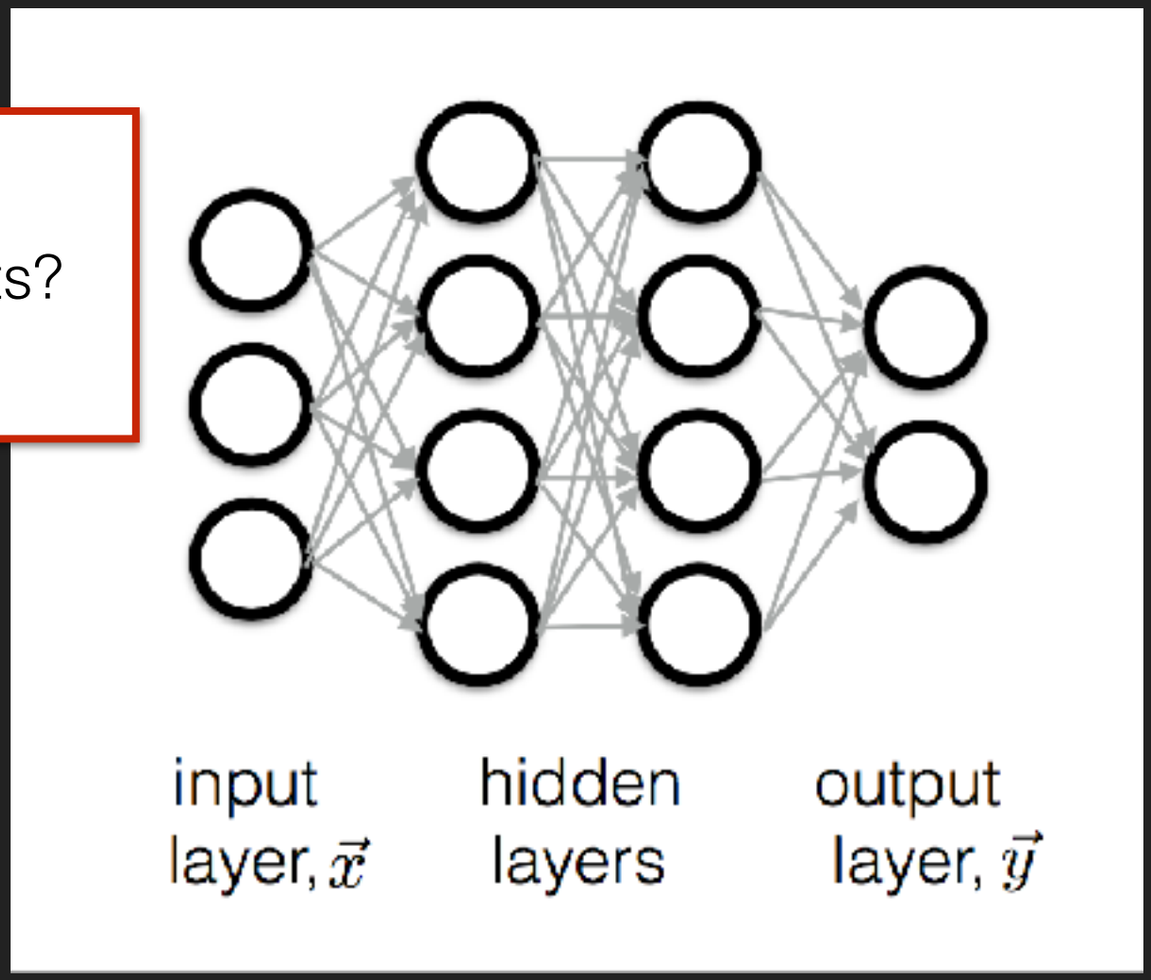


A traditional neural network consists of a stack of layers of such neurons where each neuron is ***fully connected*** to other neurons of the neighbor layers

"Traditional neural net" in HEP

Fully-Connected Multi-Layer Perceptrons

What are the inputs?



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