Deep Learning in High Energy Physics

Amir Farbin

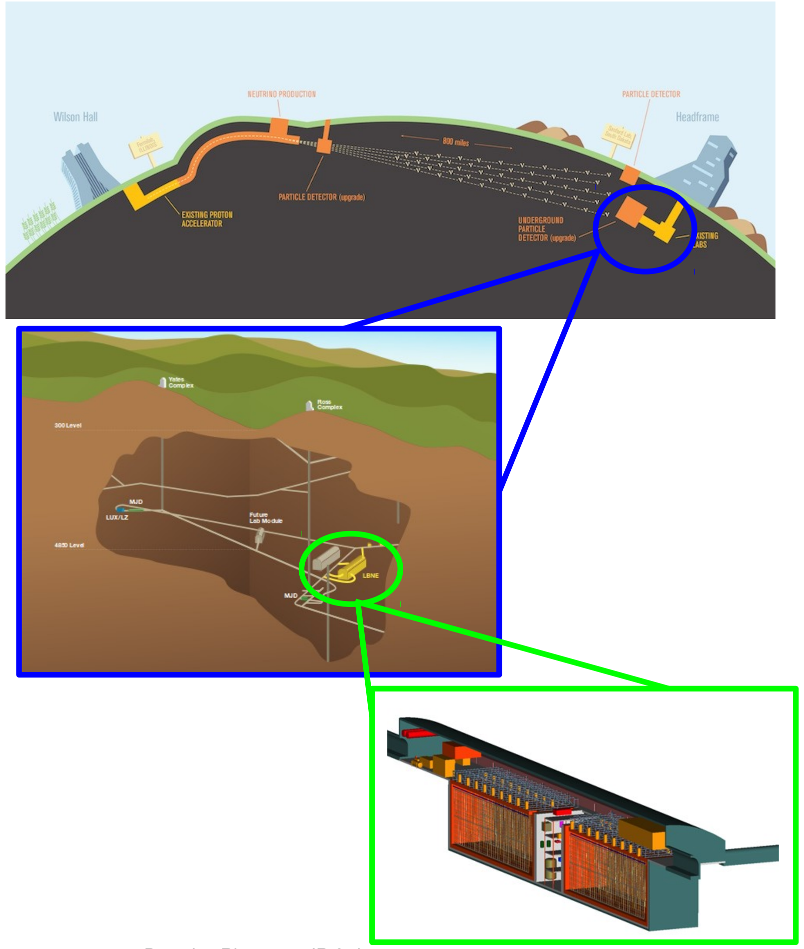
UNIVERSITY OF TEXAS ARLINGTON

ATLAS EXPERIMENT

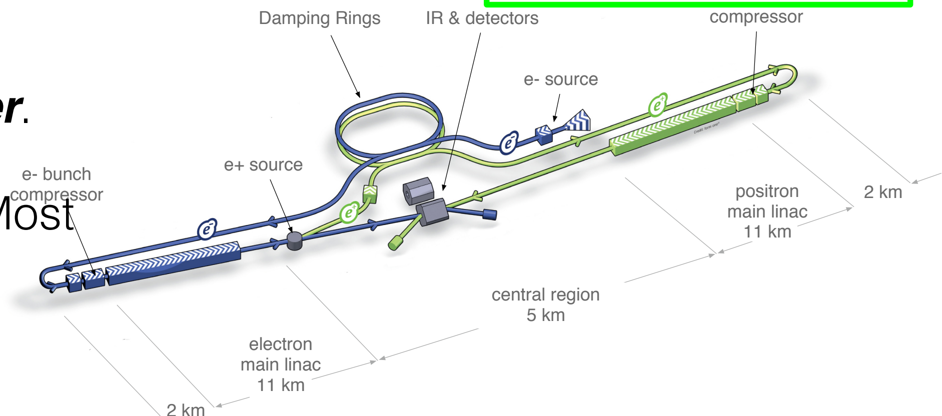
~~DUNE DEEP UNDERGROUND NEUTRINO EXPERIMENT~~

Frontiers

- ***Energy Frontier***: ***Large Hadron Collider (LHC)*** at 13 TeV now, ***High Luminosity (HL)-LHC*** by 2025, perhaps 33 TeV LHC or 100 TeV Chinese machine in a couple of decades.
    - Having found Higgs, moving to studying the SM ***Higgs*** find new Higgses
    - Test ***naturalness*** (Was the Universe and accident?) by searching for New Physics like Supersymmetry that keeps Higgs light without 1 part in 10^{18} fine-tuning of parameters.
    - Find ***Dark Matter*** (reasons to think related to naturalness)



- ***Intensity Frontier***:
    - ***B Factories***: upcoming SuperKEKB/SuperBelle
    - ***Neutrino Beam Experiments***:
        - Series of current and upcoming experiments: Nova, MicroBooNE, SBND, ICURUS
        - ***US's flagship experiment*** in next decade: ***Long Baseline Neutrino Facility (LBNF)/Deep Underground Neutrino Experiment (DUNE) at Intensity Frontier***
    - Measure properties of ***b-quarks*** and ***neutrinos*** (newly discovered mass)… search for ***matter/anti-matter asymmetry***.
    - Auxiliary Physics: Study ***Supernova***. Search for ***Proton Decay*** and ***Dark Matter***.



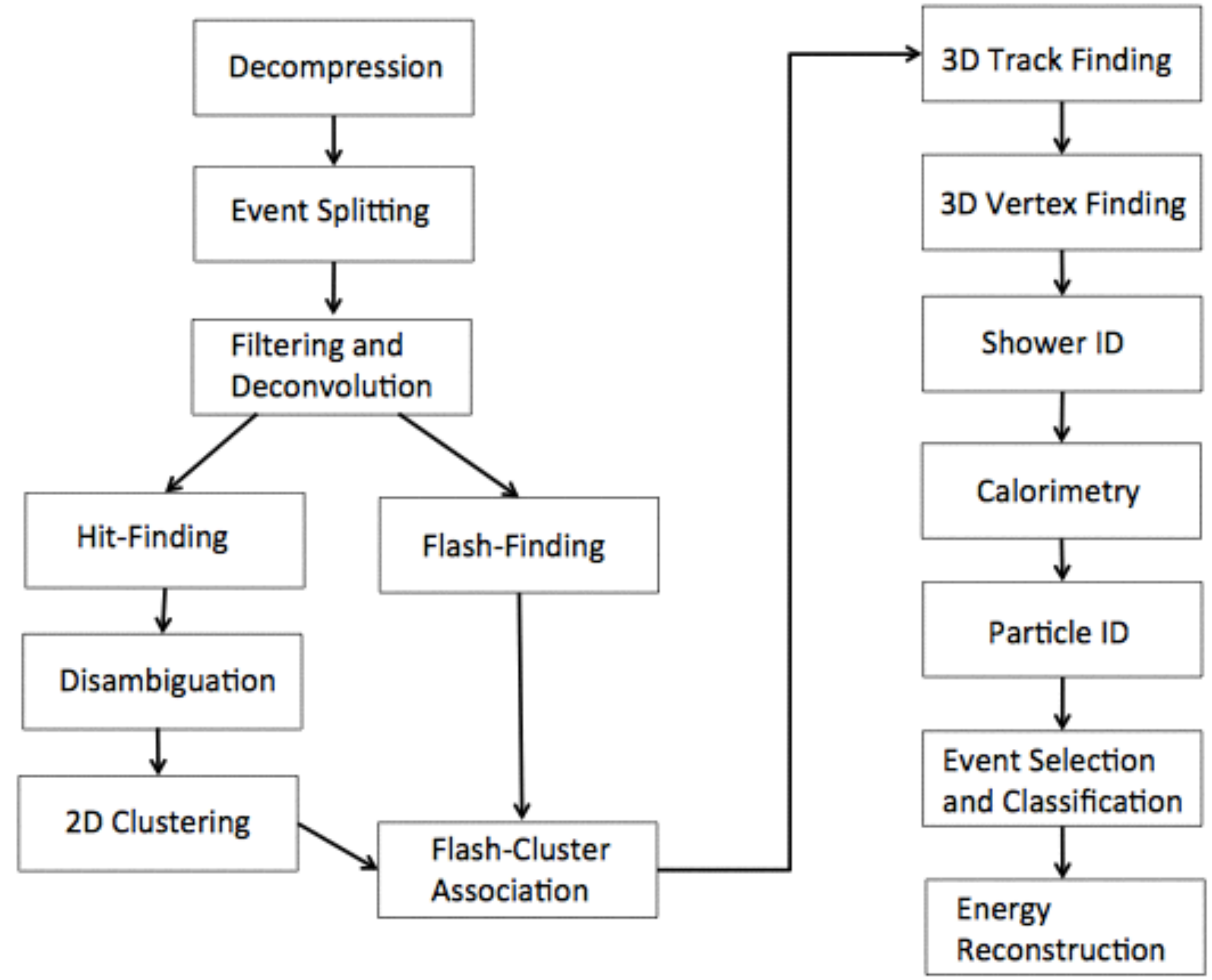
- ***Precision Frontier***: ***International Linear Collider (ILC)***, hopefully in next decade. Most energetic $e^+ e^-$ machine.
    - ***Precision studies*** of ***Higgs*** and hopefully ***new particles*** found at LHC.

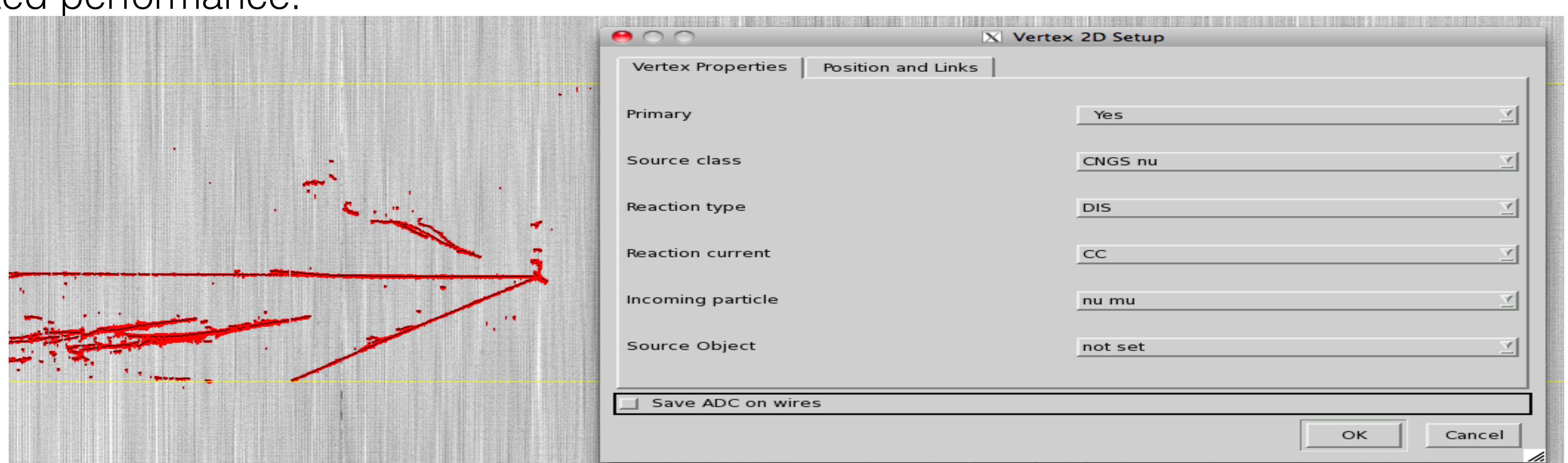
Where is ML needed?

- Traditionally ML Techniques in HEP
    - Applied to Particle/Object Identification
    - Signal/Background separation
    - Here, ML maximizes reach of existing data/detector… equivalent to additional integral luminosity.
    - There is lots of interesting work here… but I'll skip today.
        - Some slides in backup…
- Now we hope ML can help address looming computing problems
    - Reconstruction
        - LArTPC- Algorithmic Approach very difficult
        - HL-LHC Tracking- Pattern Recognition blows up due to combinatorics
    - Simulation
        - LHC Calorimetry- Large Fraction of ATLAS CPU goes into shower simulation.

LArTPC Reco Challenge

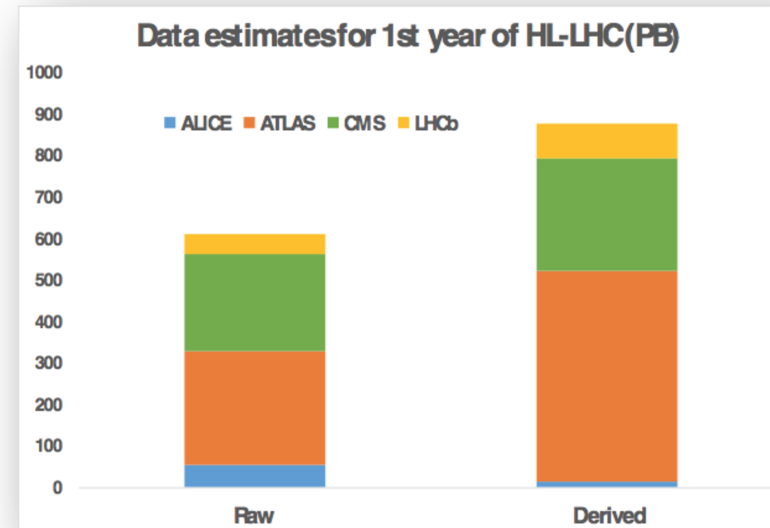
- Neutrino Physics has a long history of *hand scans*.
  - QScan: ICARUS user assisted reconstruction.
- Full automatic reconstruction has yet to be demonstrated.
  - LArSoft project:
    - art framework + LArTPC reconstruction algorithm
    - started in ArgoNeuT and contributed to/used by many experiments.
  - Full neutrino reconstruction is still far from expected performance.



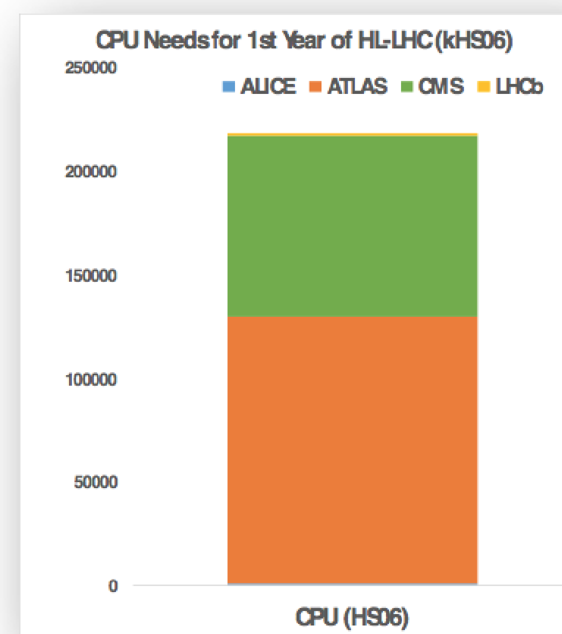


Computing Challenge

- ***Computing*** is perhaps the biggest challenge for the HL-LHC
  - ***Higher Granularity*** = larger events.
  - ***O(200) proton collision / crossing: tracking pattern recognition combinatorics becomes untenable***.
  - O(100) times data = multi ***exabyte datasets***.
  - ***Moore's law has stalled***: Cost of adding more transistors/silicon area no longer decreasing…. for processors. Many-core co-processors still ok.
    - Naively we need 60x more CPU, with 20%/year Moore's law giving only 6-10x in 10-11 years.
  - Preliminary estimates of ***HL-LHC computing budget many times larger than LHC***.
- ***Solutions***:
  - ***Leverage opportunistic resources and HPC*** (most computation power in highly parallel processors).
  - ***Highly parallel processors*** (e.g. GPUs) are already > 10x CPUs for certain computations.
    - Trend is away from x86 towards ***specialized hardware*** (e.g. GPUs, Mics, FPGAs, Custom DL Chips)
    - Unfortunately parallelization (i.e. Multi-core/GPU) has been extremely difficult for HEP.



Data:
- Raw 2016: 50 PB → 2027: 600 PB
- Derived (1 copy): 2016: 80 PB → 2027: 900 PB



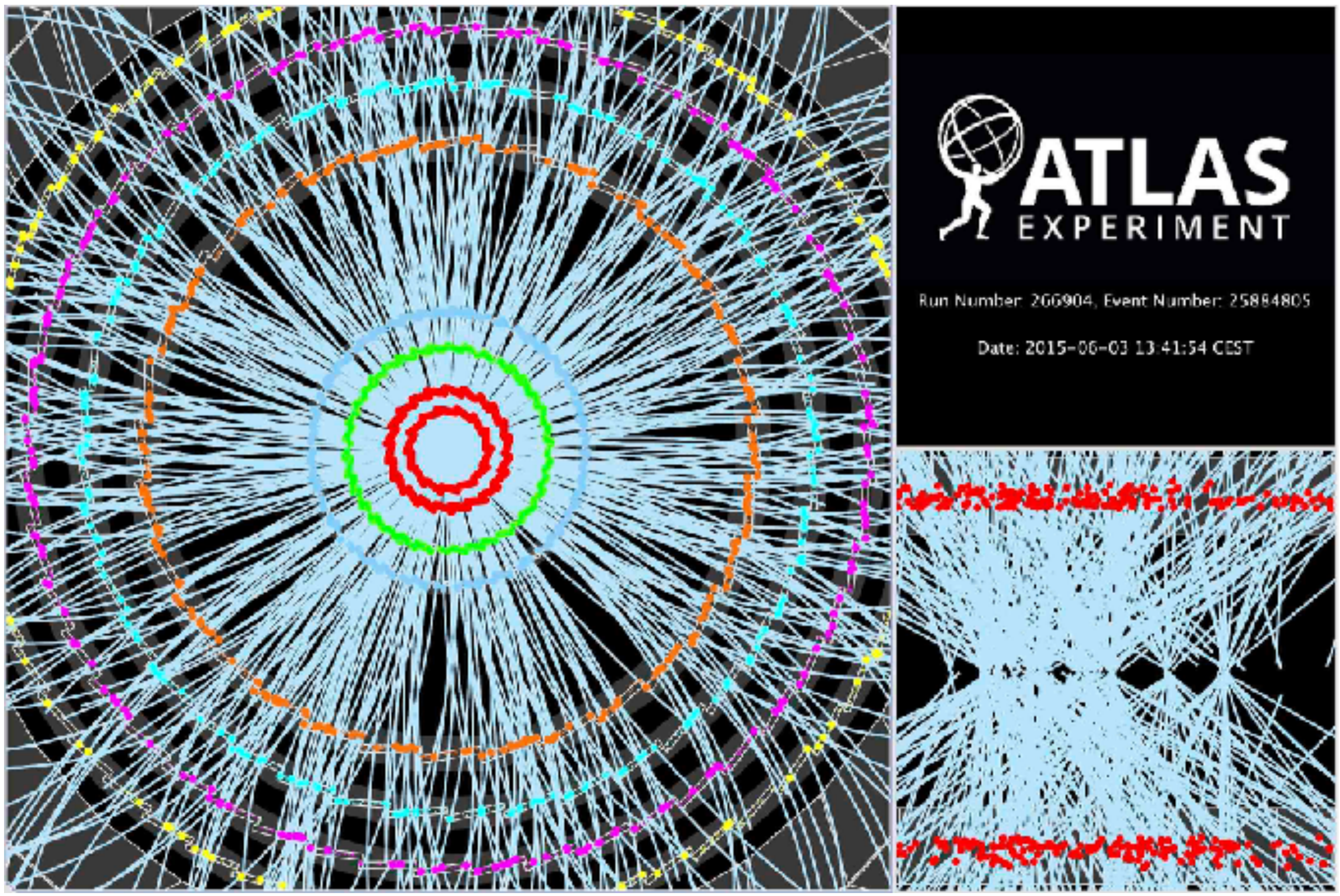
CPU:
- x60 from 2016

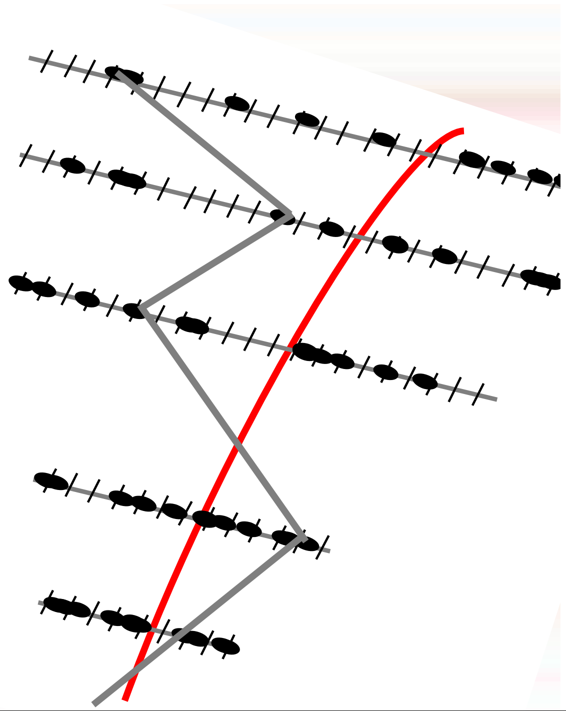
From WLCG Workshop Intro, Ian Bird, 8 Oct, 2016

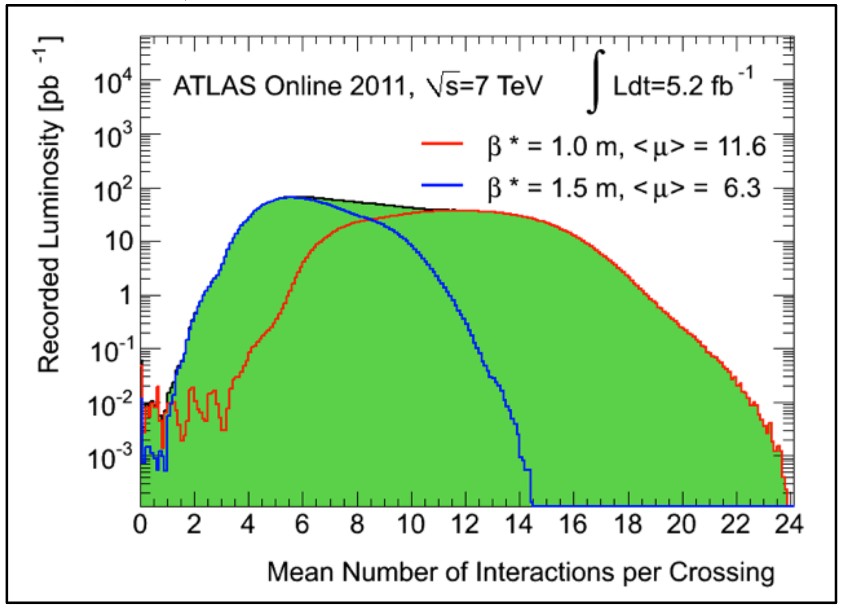
Reconstruction

Tracking

- Measure Charged particle trajectories. If B-field, then measure momentum.

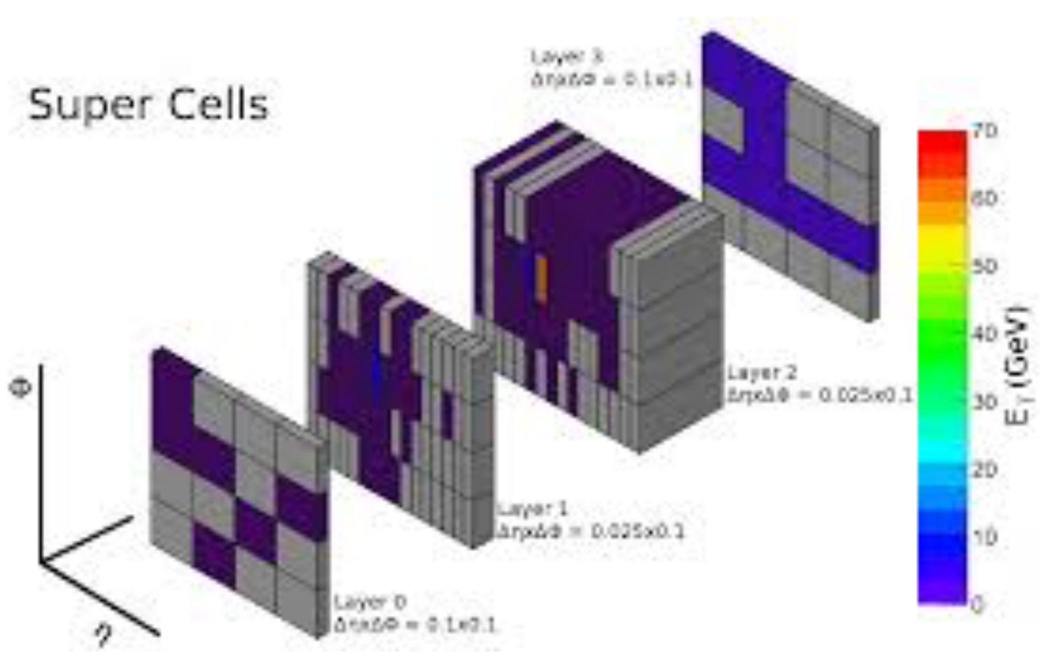


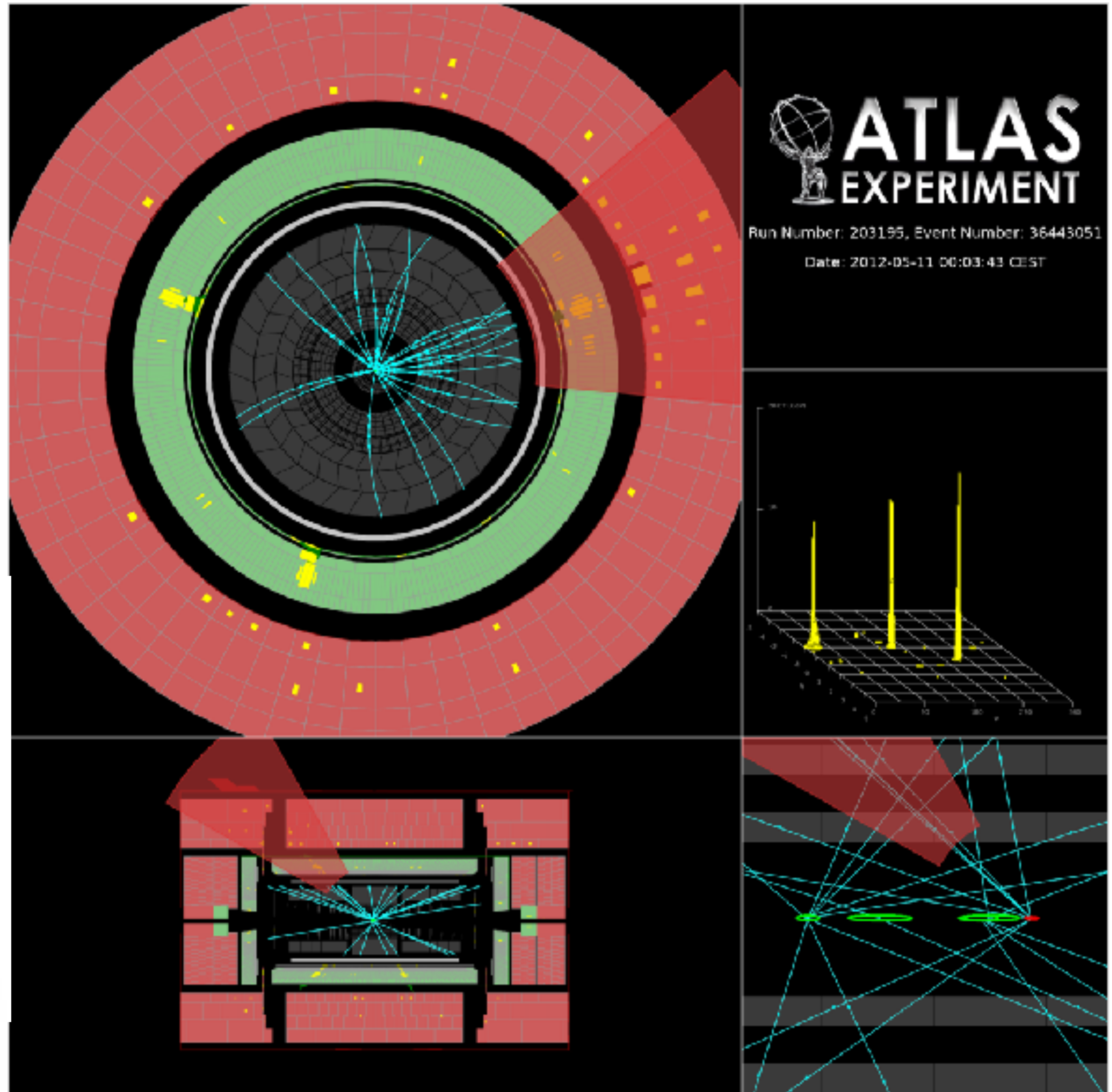




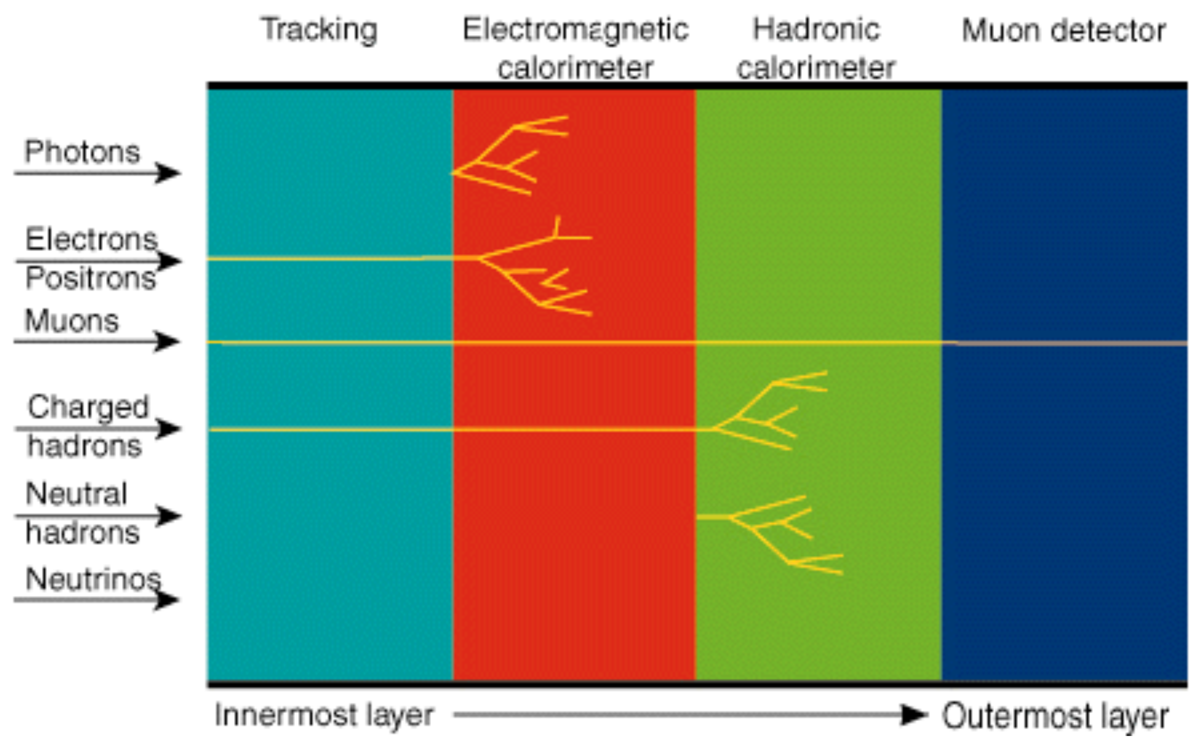
Calorimetry

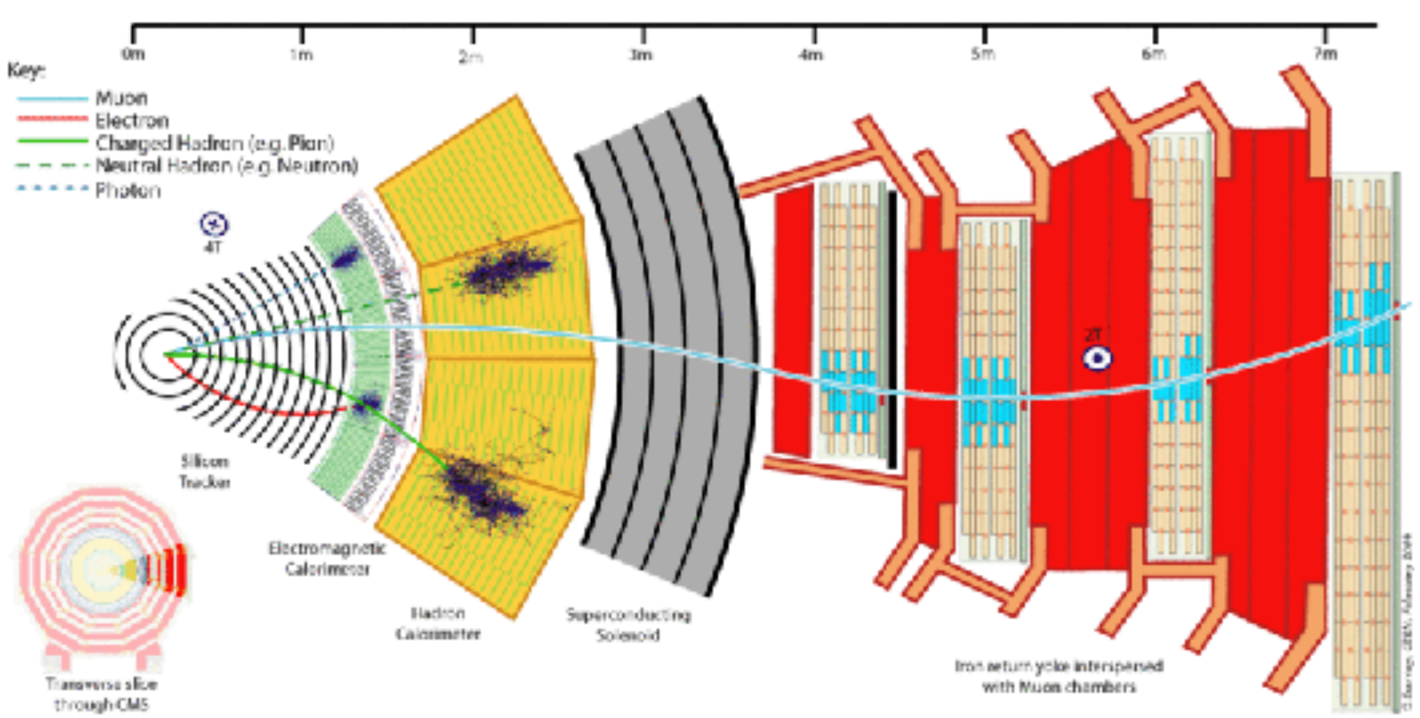
- Make particle interact and loose all energy, which we measure. 2 types:
    - Electromagnetic: e.g. crystals in CMS, Liquid Argon in ATLAS.
    - Hadronic: e.g. steel + scintillators
    - e.g ATLAS:
        - 200K Calorimeter cells measure energy deposits.
        - 64 x 36 x 7 3D Image

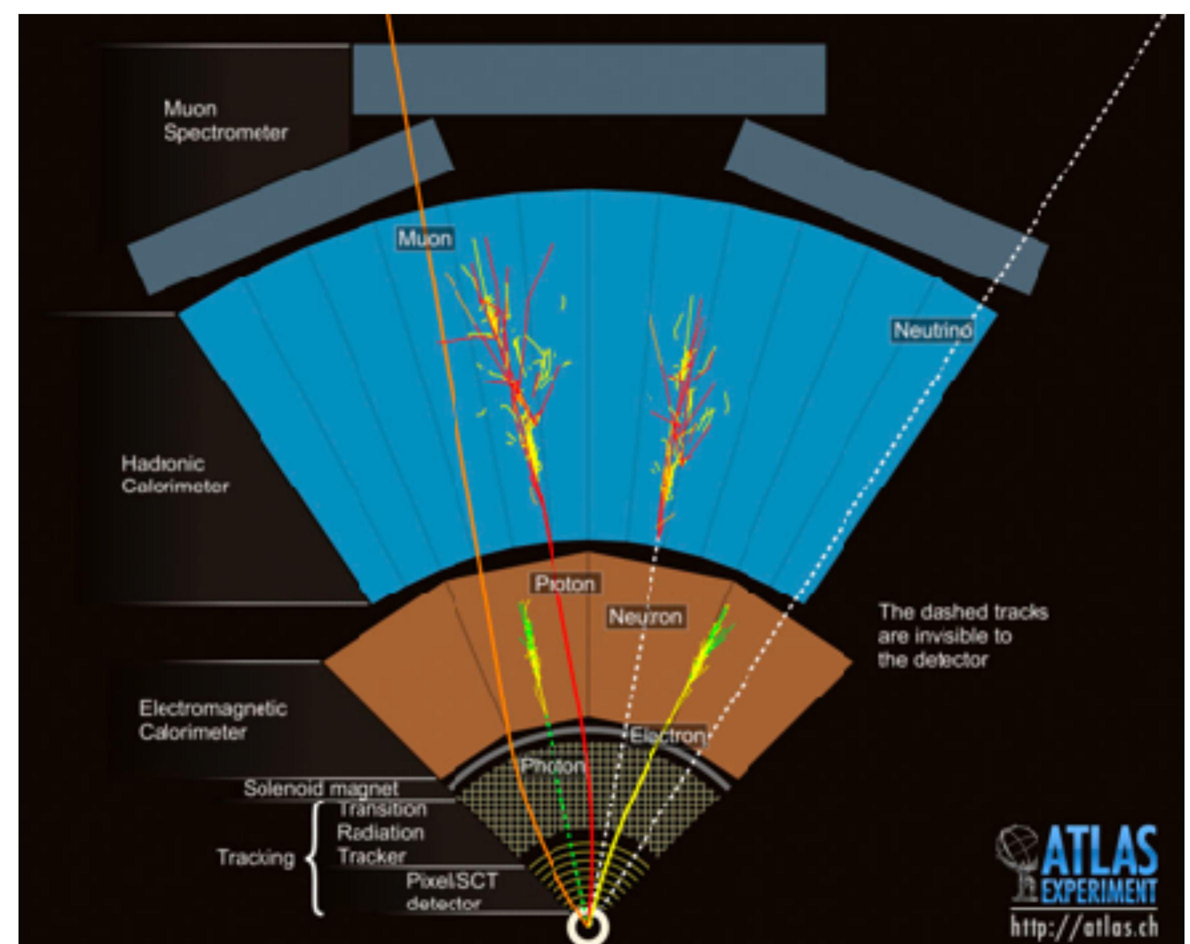




LHC/ILC detectors

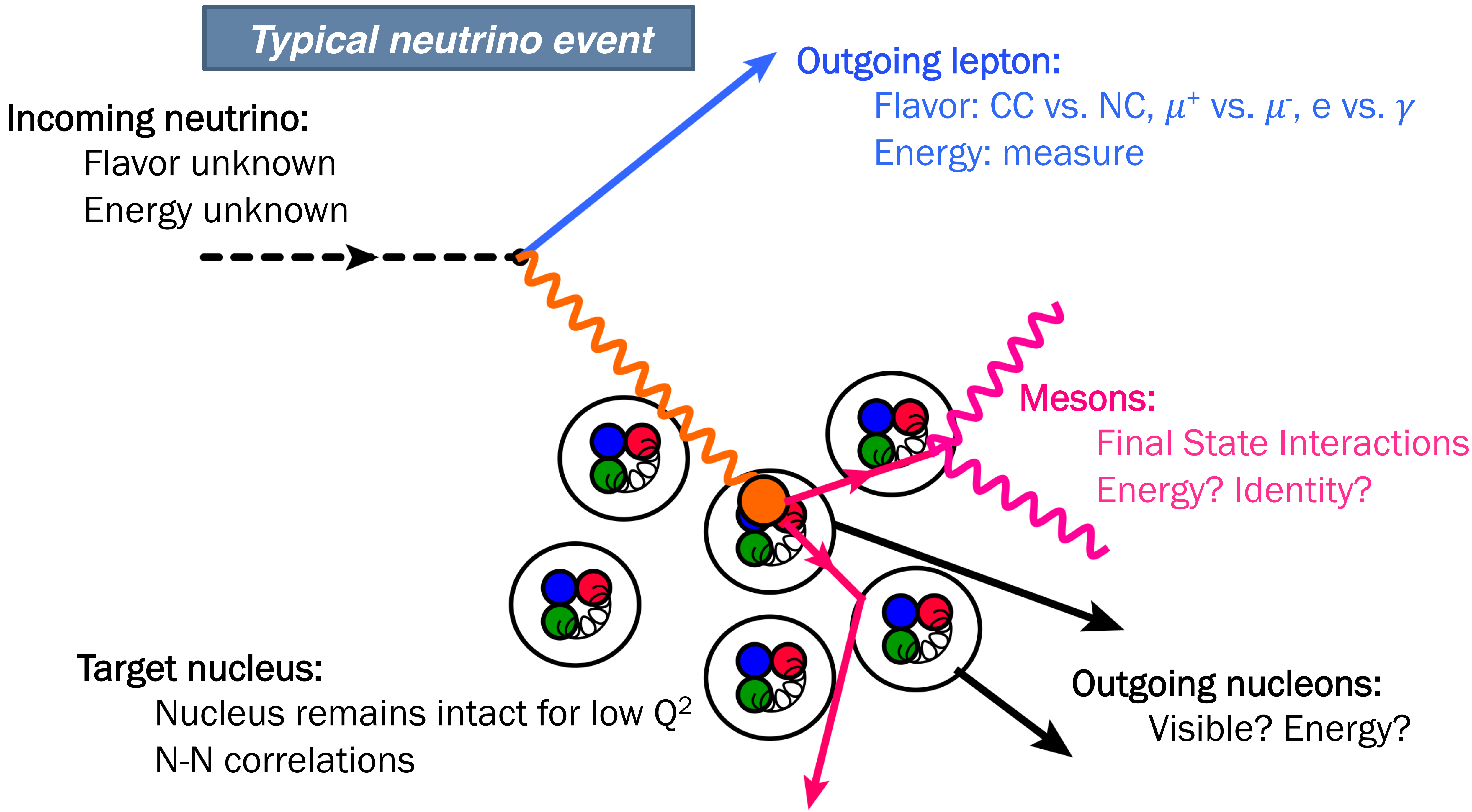






Neutrino Detection

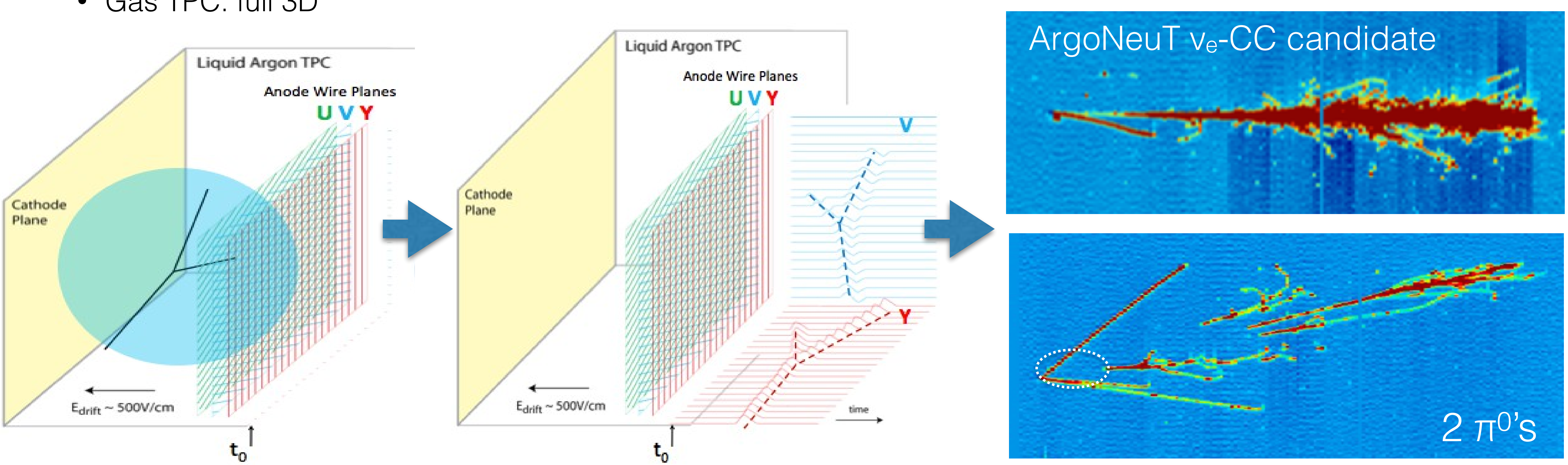
In neutrino experiments, try to determine flavor and estimate energy of incoming neutrino by looking at outgoing products of the interaction.



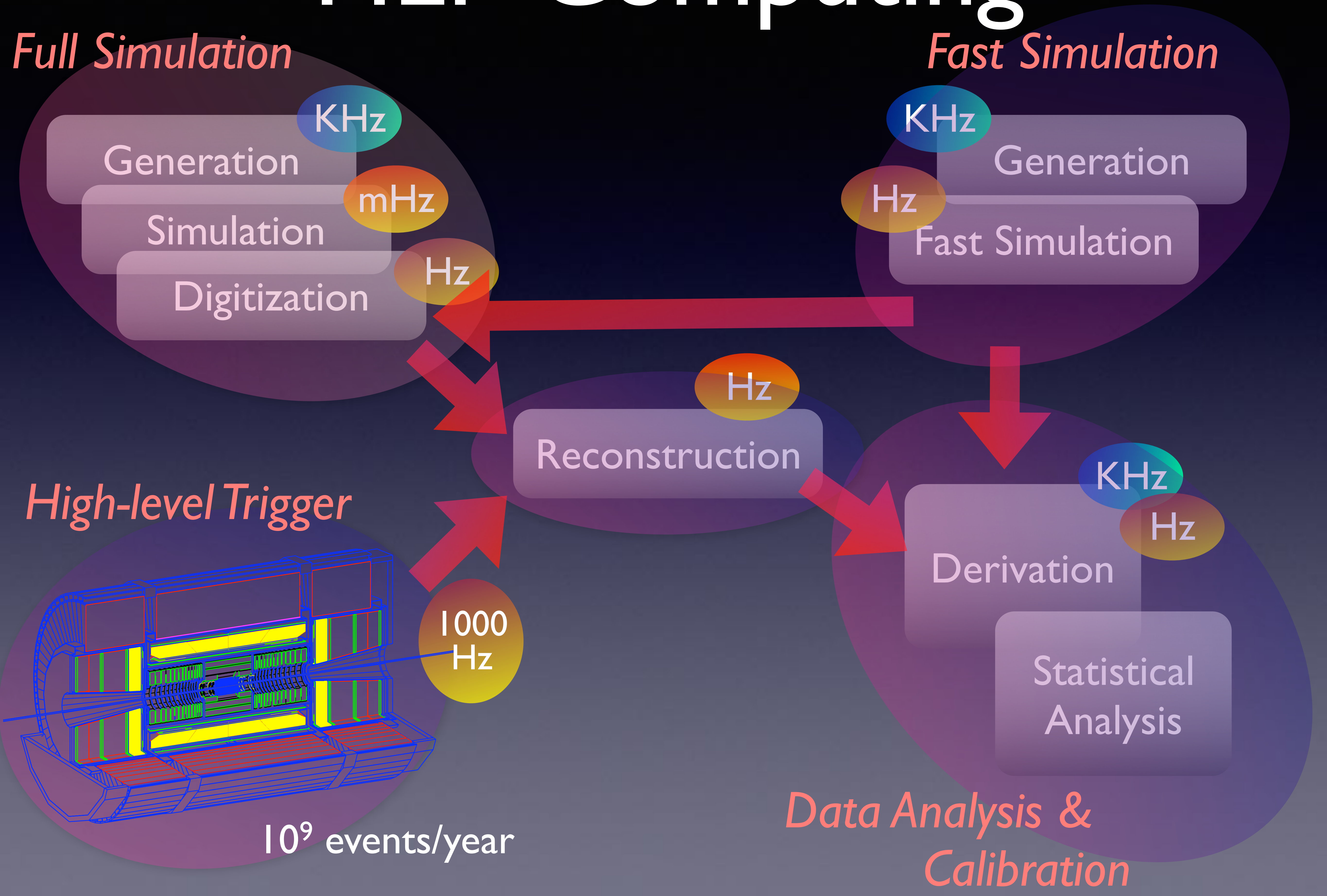
Jen Raaf

Neutrino Detectors

- ***Need large mass/volume*** to maximize chance of neutrino interaction.
- Technologies:
  - Water/Oil Cherenkov
  - Segmented Scintillators
  - ***Liquid Argon Time Projection Chamber: promises ~ 2x detection efficiency.***
    - ***Provides tracking, calorimetry, and ID all in same detector***.
    - Chosen technology for US’s flagship LBNF/DUNE program.
    - Usually 2D read-out… 3D inferred.
  - Gas TPC: full 3D

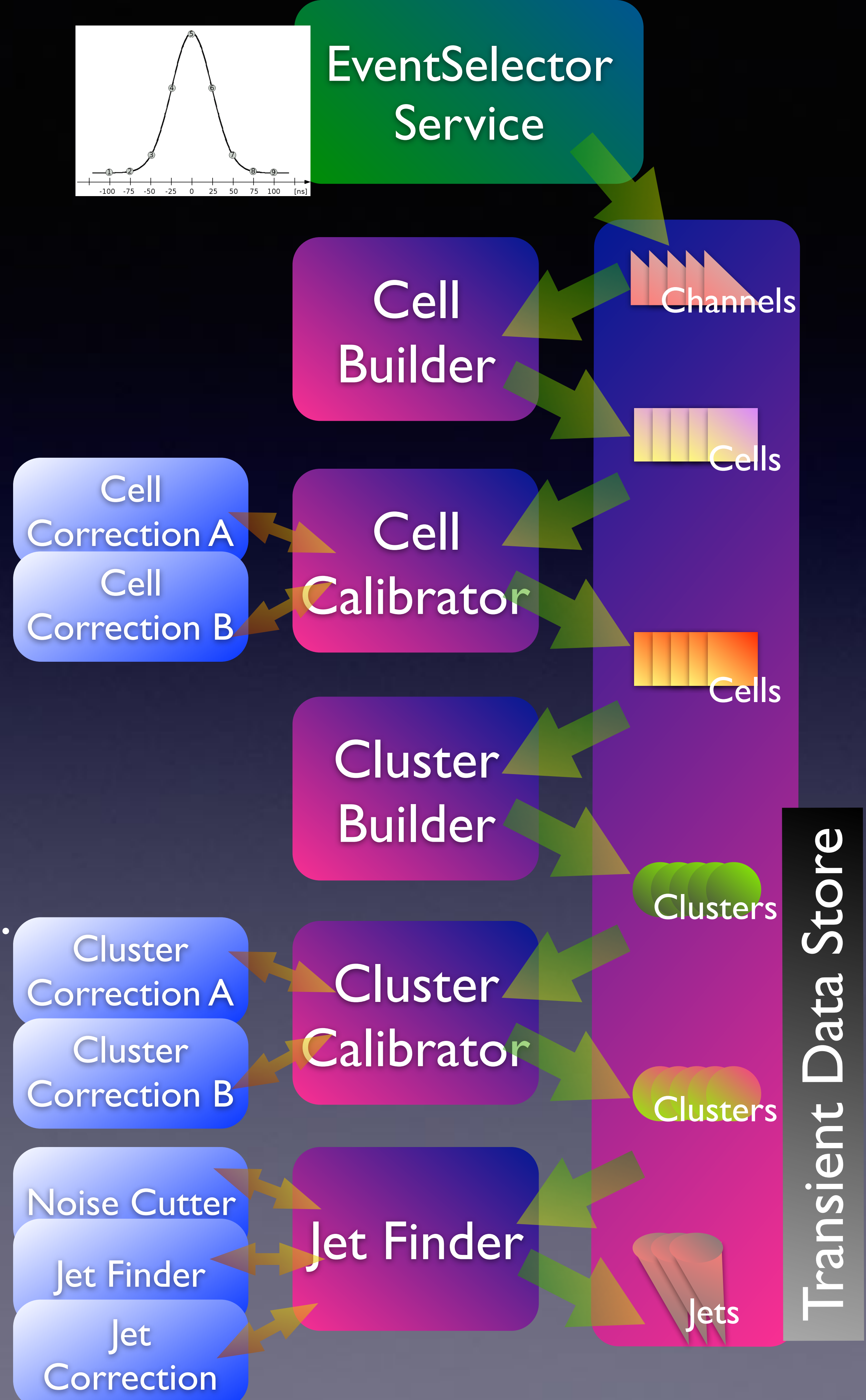


HEP Computing



Reconstruction

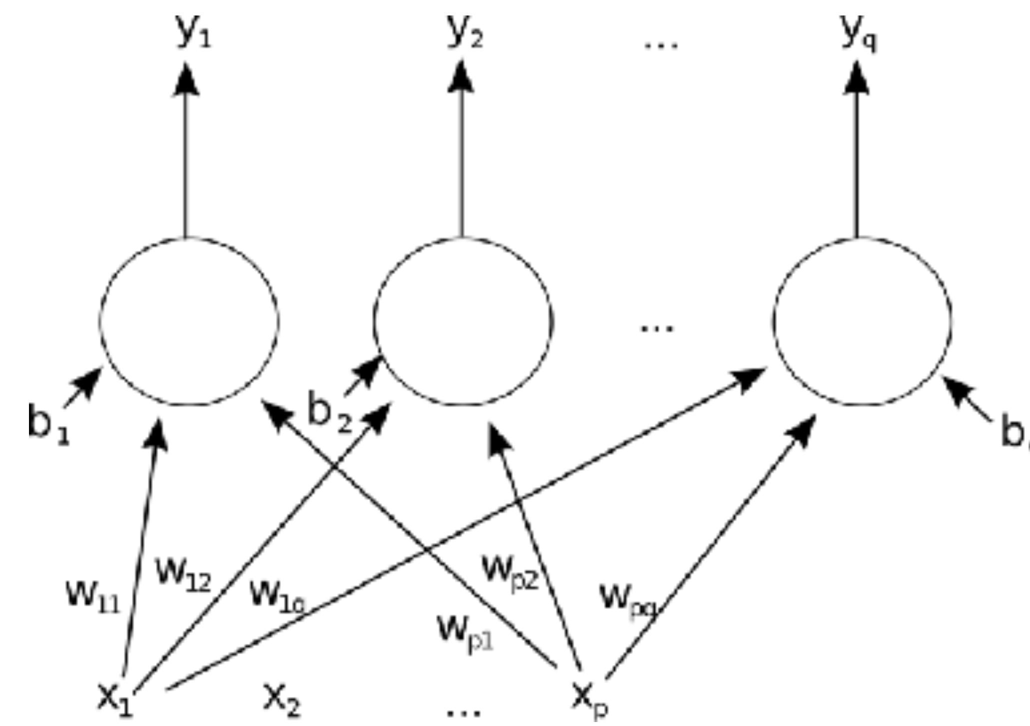
- Starts with ***raw inputs*** (e.g. Voltages)
- Low level ***Feature Extraction***: e,g, Energy/Time in each Calo Cell
- ***Pattern Recognition***: Cluster adjacent cells. Find hit pattern.
- ***Fitting***: Fit tracks to hits.
- ***Combined reco***: e.g.:
  - Matching Track+EM Cluster = Electron.
  - Matching Track in inter detector + muon system = Muon
- ***Output particle candidates*** and measurements of their properties (e.g. energy)

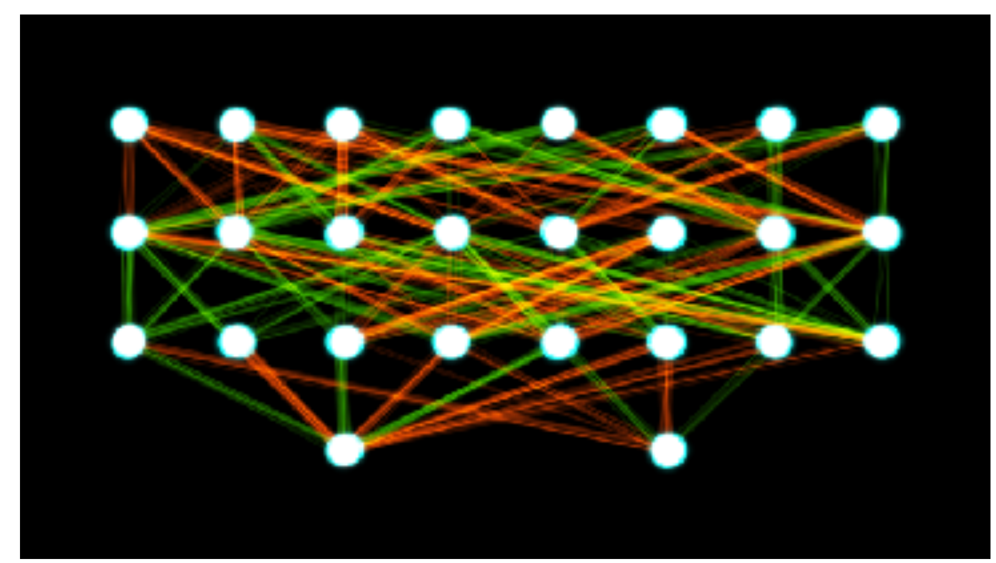


Deep Learning

Artificial Neural Networks

- ***Biologically inspired computation***, (first attempts in 1943)
  - ***Probabilistic Inference***: e.g. signal vs background
  - ***Universal Computation Theorem*** (1989)
- Common use in HEP, ***signal/background classification*** or particle ID with ***high-level features derived from raw data*** as input.
- Multi-layer (***Deep***) Neutral Networks:
  - Not a new idea (1965), just impractical to train. ***Vanishing Gradient problem*** (1991)
  - Solutions:
    - New techniques: e.g. better activation or layer-wise training
    - ***More training***: big training datasets and lots of computation … ***big data and GPUs***
  - ***Deep Learning Renaissance***. First DNN in HEP (2014).
  - ***Amazing Feats***: Audio/Image/Video recognition, captioning, and generation. Text (sentiment) analysis. Language Translation. Video game playing agents.
  - ***Rich field***: Variety of architectures, techniques, and applications.





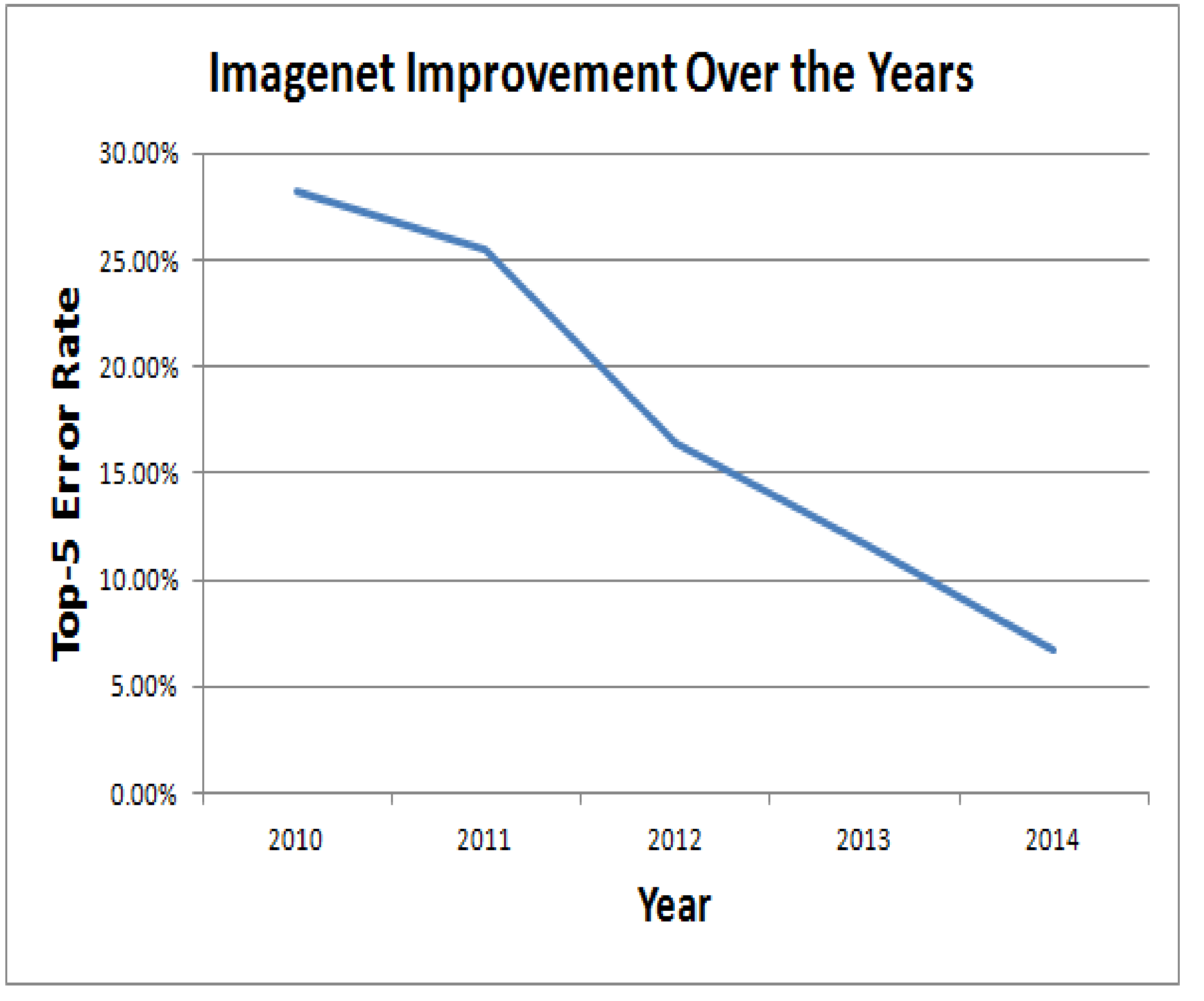
Images from Wikipedia

Recent History

- Deep Learning feats that sparked broad interest:
  - 2012, Google 1B DNN learns to identify cats (and 20000 other types of objects) (Wired Article, paper)
    - *Raw input*: trained with 200x200 pixel images from YouTube
    - *Unsupervised*: the pictures were unlabeled.
    - Google cluster 16000 cores ~ $1M. Redone with $20k system with GPUs.
  - 2013: Deep Mind builds AI that plays ATARI (Blogpost, Nature,YouTube,YouTube)

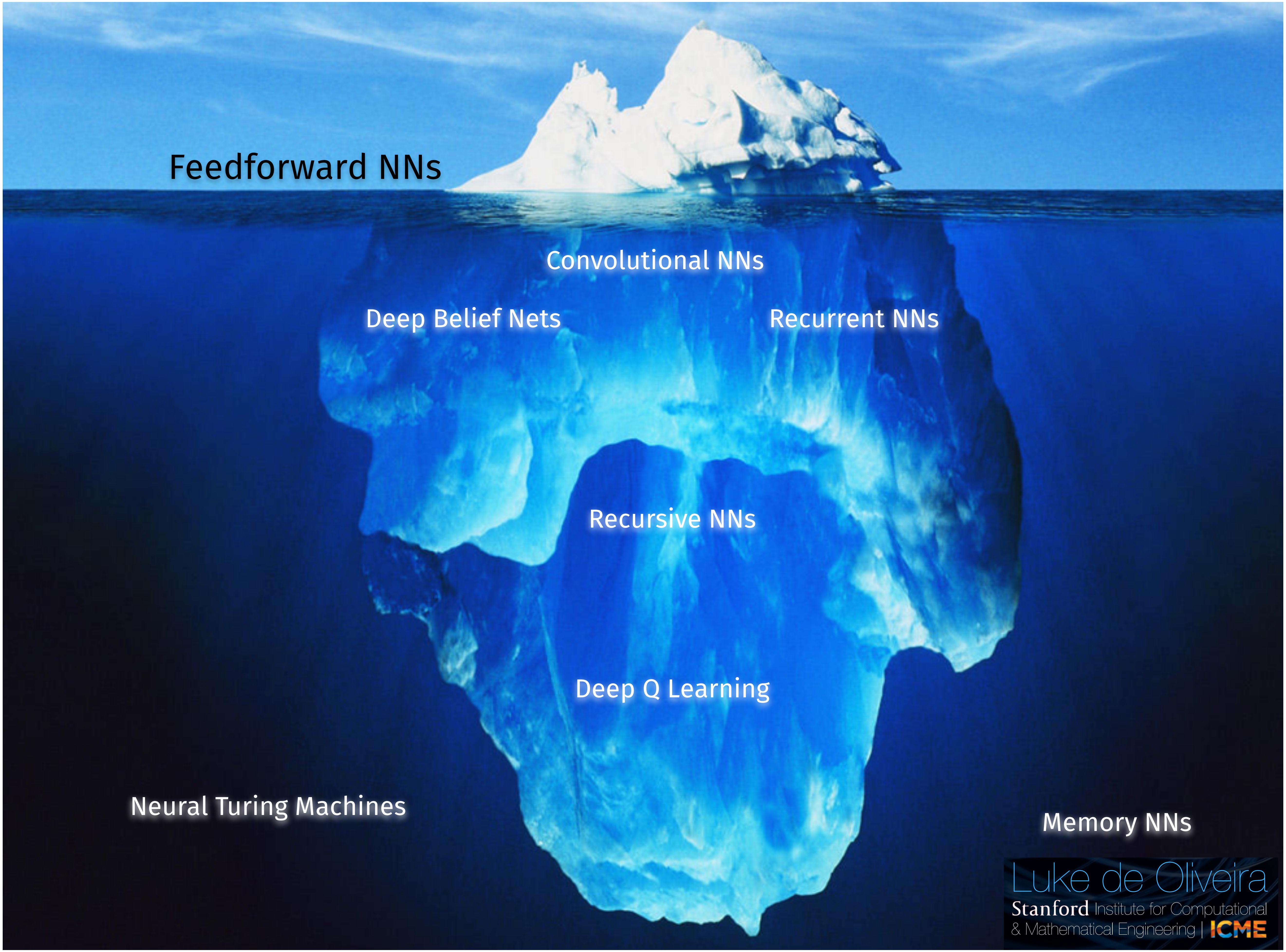
Computer Vision - Image Classification

- **Imagenet**
- Over 1 million images, 1000 classes, different sizes, avg 482x415, color
- 16.42% Deep CNN dropout in 2012
- 6.66% 22 layer CNN (GoogLeNet) in 2014
- 4.9% (Google, Microsoft) super-human performance in 2015



Sources: Krizhevsky et al ImageNet Classification with Deep Convolutional Neural Networks, Lee et al Deeply supervised nets 2014, Szegedy et al, Going Deeper with convolutions, ILSVRC2014, Sanchez & Perronnin CVPR 2011, http://www.clarifai.com/ Benenson, http://rodrigob.github.io/are_we_there_yet/build/classification_datasets_results.html

P.Baldi


Feedforward NNs
Convolutional NNs
Deep Belief Nets
Recurrent NNs
Recursive NNs
Deep Q Learning
Neural Turing Machines
Memory NNs
Luke de Oliveira
Stanford Institute for Computational
& Mathematical Engineering | ICME

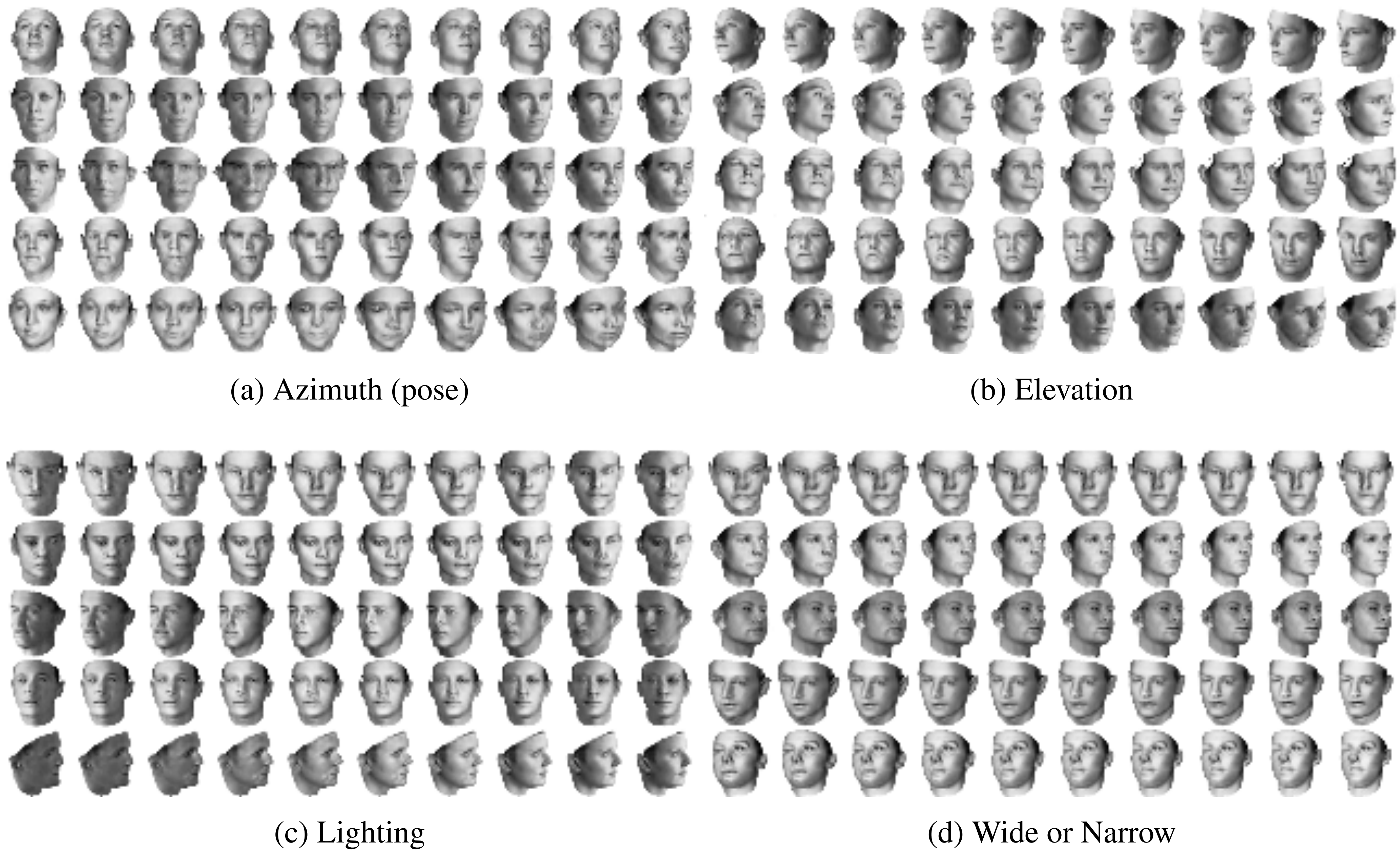
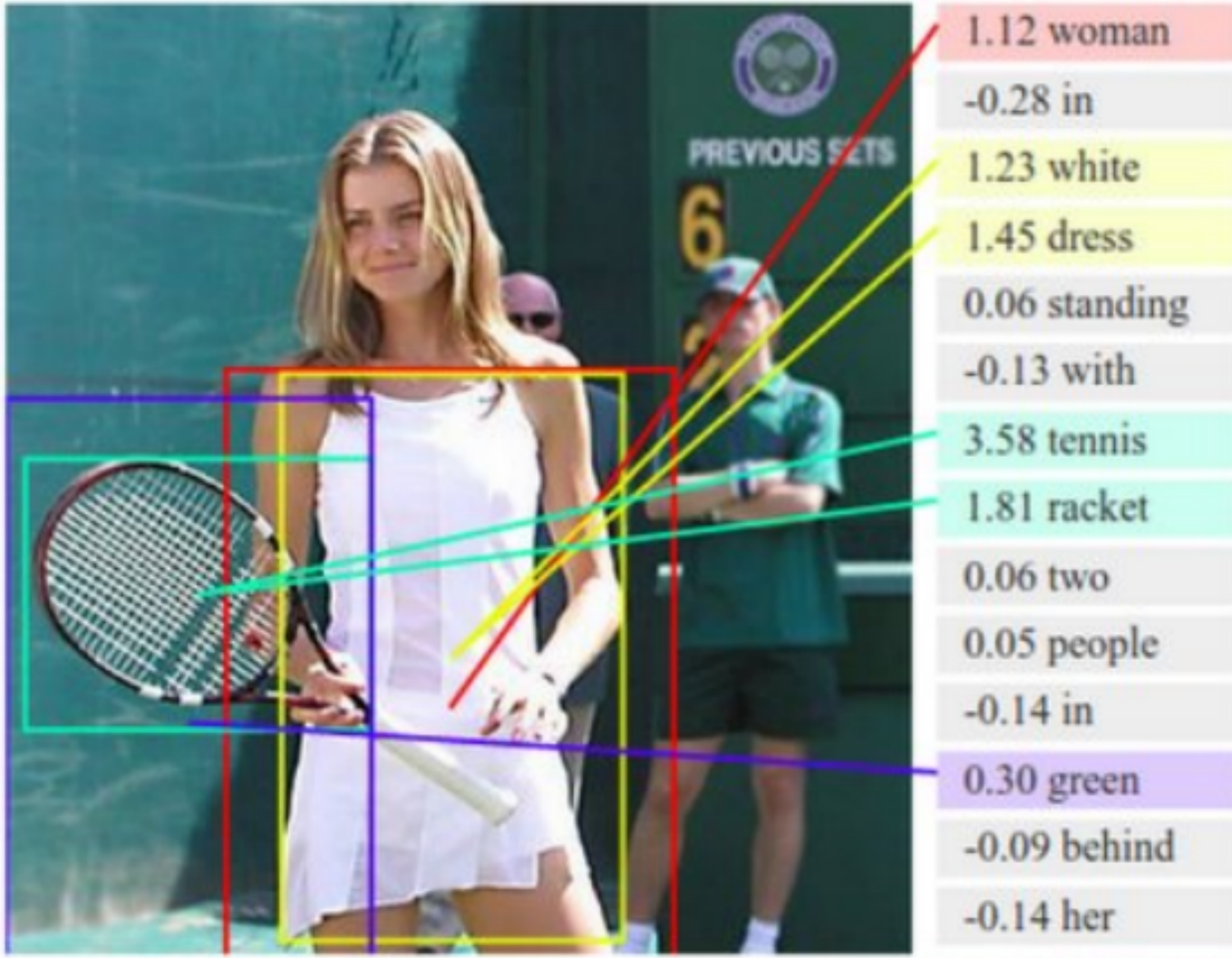


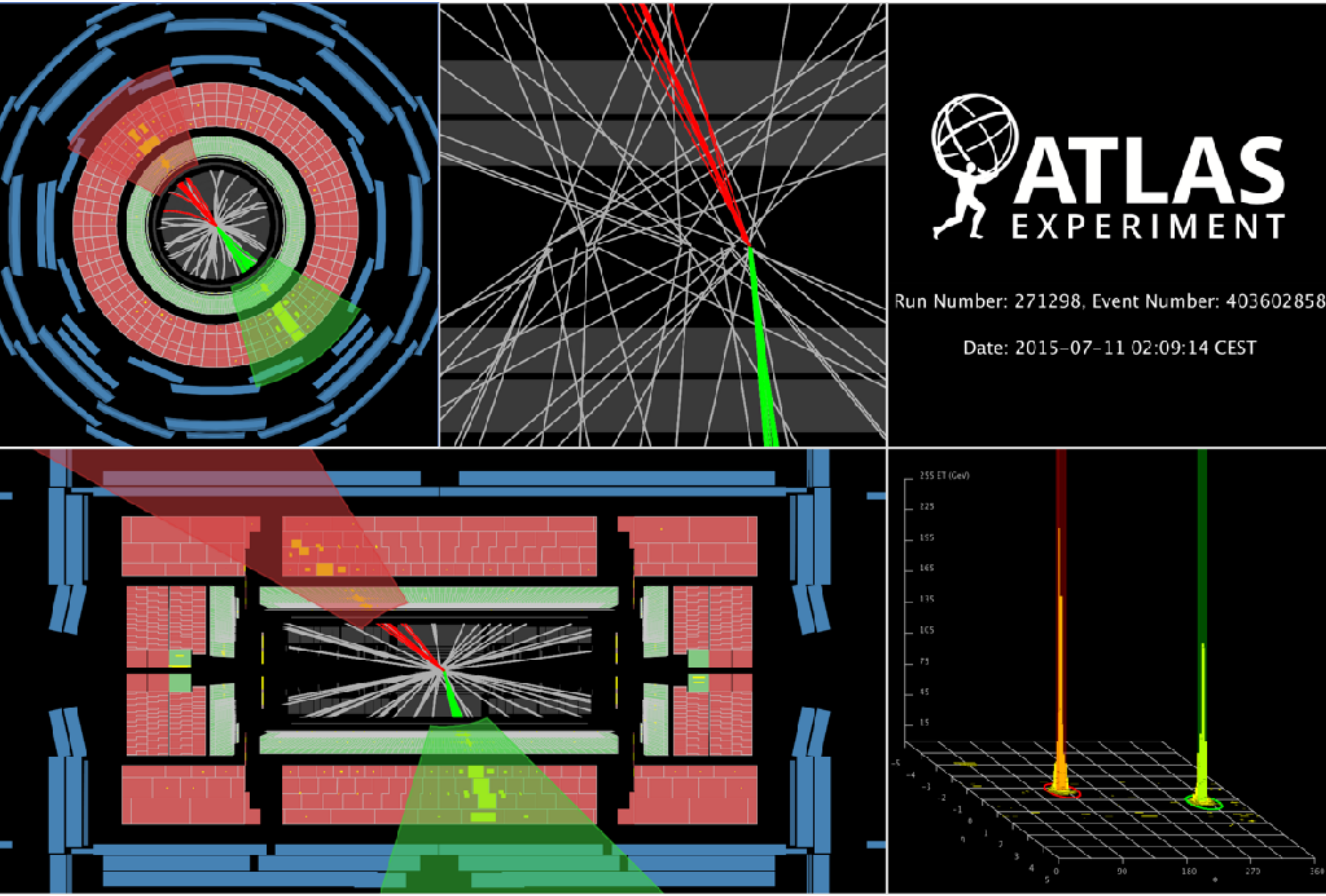
Figure 3: **Manipulating latent codes on 3D Faces:** We show the effect of the learned continuous latent factors on the outputs as their values vary from -1 to 1. In (a), we show that one of the continuous latent codes consistently captures the azimuth of the face across different shapes; in (b), the continuous code captures elevation; in (c), the continuous code captures the orientation of lighting; and finally in (d), the continuous code learns to interpolate between wide and narrow faces while preserving other visual features. For each factor, we present the representation that most resembles prior supervised results [7] out of 5 random runs to provide direct comparison.

https://arxiv.org/pdf/1606.03657.pdf

Why go Deep?

- ***Better Algorithms***
  - DNN-based classification/regression generally **out perform** hand crafted algorithms.
  - In some cases, it may provide a ***solution*** where ***algorithm approach doesn't exist or fails***.
  - ***Unsupervised learning***: make sense of complicated data that we don't understand or expect.
- ***Easier Algorithm Development***: ***Feature Learning*** instead of *Feature Engineering*
  - Reduce time physicists spend writing developing algorithms, ***saving time and cost***. (e.g. ATLAS > $250M spent software)
  - Quickly perform performance ***optimization*** or ***systematic studies***.
- ***Faster Algorithms***
  - After training, DNN inference is often *faster* than sophisticated algorithmic approach.
  - DNN can ***encapsulate expensive computations***, e.g. Matrix Element Method.
  - ***Generative Models*** enable fast simulations.
  - ***Already parallelized*** and optimized for GPUs/HPCs.
  - ***Neuromorphic*** processors.

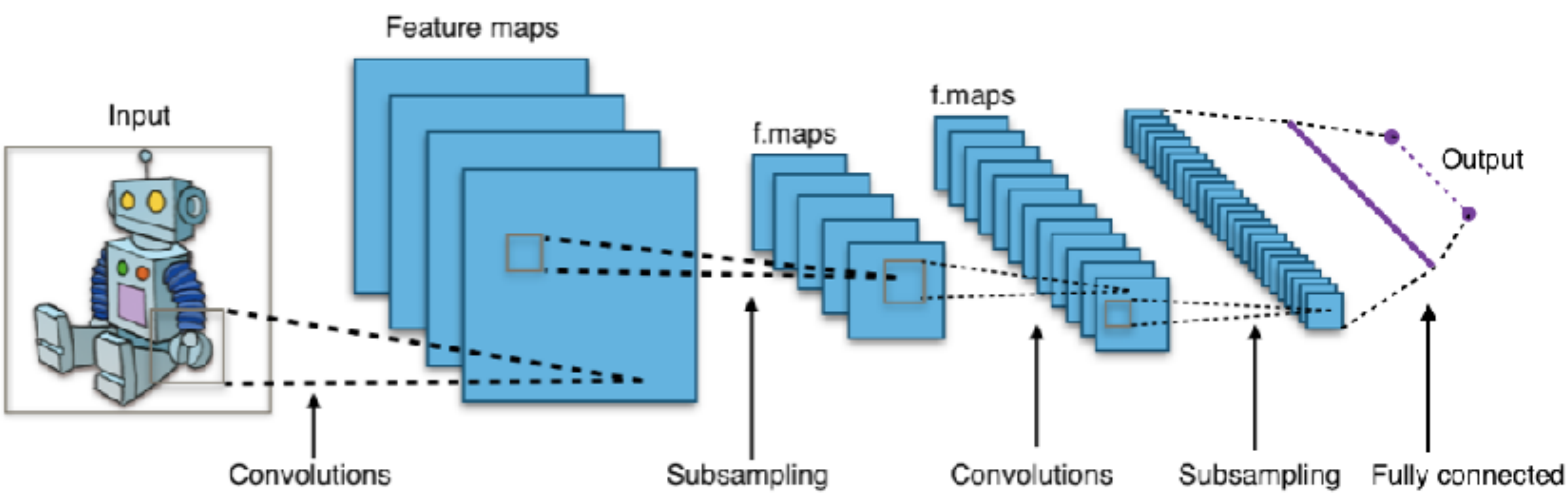

PREVIOUS SETS
1.12 woman
-0.28 in
1.23 white
1.45 dress
0.06 standing
-0.13 with
3.58 tennis
1.81 racket
0.06 two
0.05 people
-0.14 in
0.30 green
-0.09 behind
-0.14 her

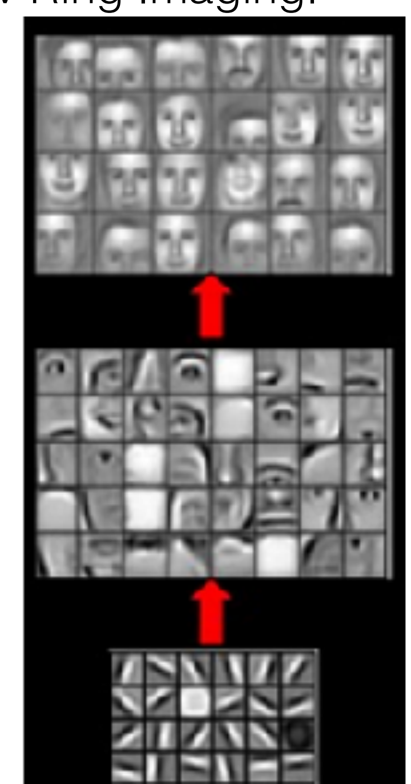

ATLAS
EXPERIMENT
Run Number: 271298, Event Number: 403602858
Date: 2015-07-11 02:09:14 CEST

Examples of DL in HEP

Feature Learning

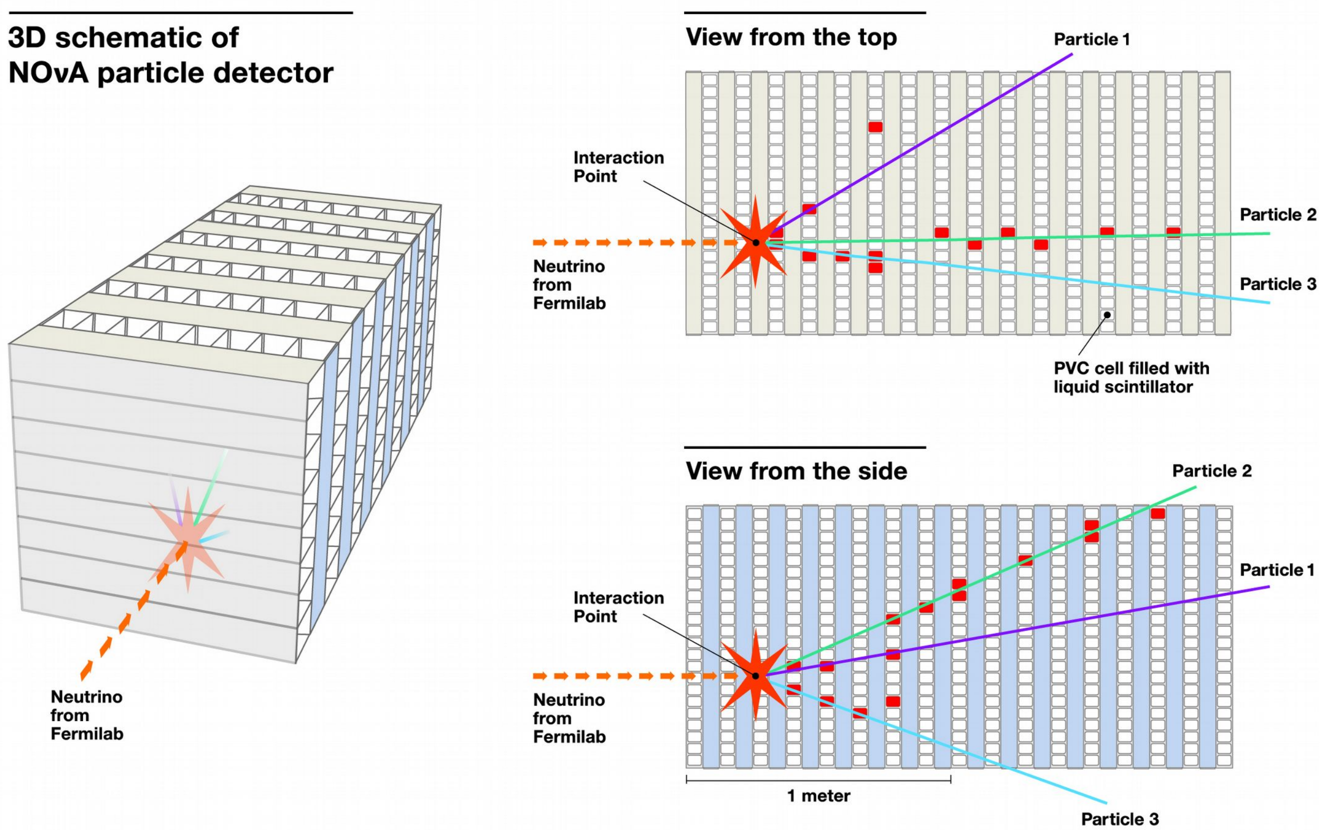
- ***Feature Engineering***: e.g. Event Reconstruction ~ Feature Extraction, Pattern Recognition, Fitting, …
- Deep Neutral Networks can ***Learn Features*** from ***raw data***.
- Example: ***Convolutional Neural Networks*** - Inspired by visual cortex
    - ***Input***: Raw data… for example 1D = Audio, 2D = Images, 3D = Video
    - ***Convolutions*** ~ learned feature detectors
    - ***Feature Maps***
    - ***Pooling*** - dimension reduction / invariance
    - ***Stack***: Deeper layers recognize higher level concepts.
- Over the past few years, CNNs have lead to ***exponential improvement / superhuman performance on Image classification*** challenges. Current best > 150 layers.
- Obvious HEP application: ***"Imaging" Detectors*** such as TPCs, High Granularity Calorimeters, or Cherenkov Ring Imaging.





Neutrino Physics

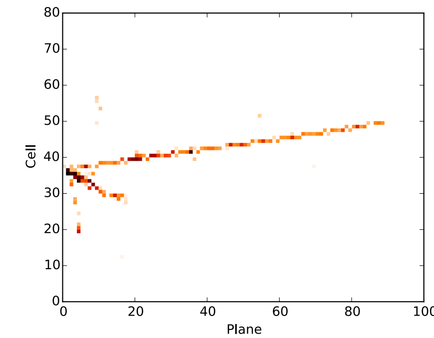
- Core Physics requires just measuring ***neutrino flavor and energy***.
- Generally clean (low multiplicity) and high granularity.
- ***First HEP CNN application: <u>Nova</u>*** using Siamese Inception CNN.



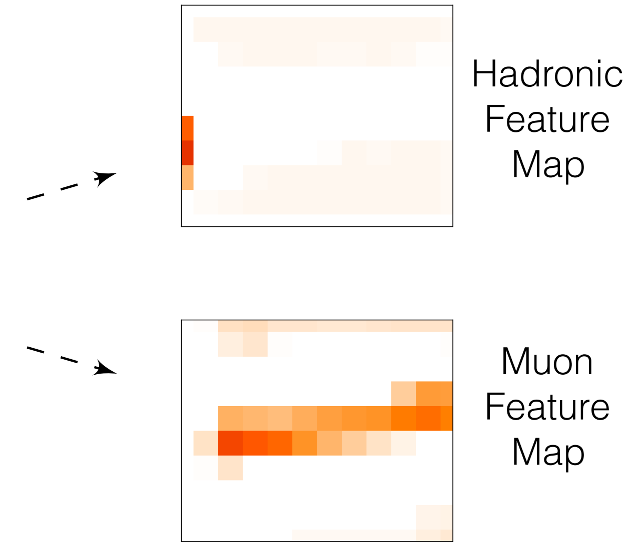
	CVN Selection Value	ν_e sig	Tot bkg	NC	ν_μ CC	Beam ν_e	Signal Efficiency	Purity
Contained Events	–	88.4	509.0	344.8	132.1	32.1	–	14.8%
$s/\sqrt{b}$ opt	0.94	43.4	6.7	2.1	0.4	4.3	49.1%	86.6%
$s/\sqrt{s+b}$ opt	0.72	58.8	18.6	10.3	2.1	6.1	66.4%	76.0%

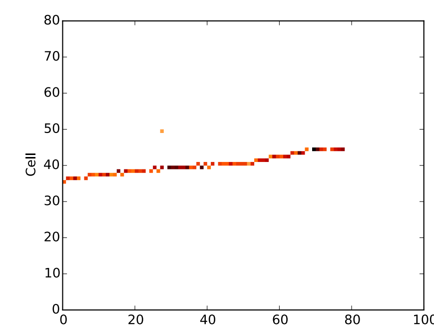
	CVN Selection Value	ν_μ sig	Tot bkg	NC	Appeared ν_e	Beam ν_e	Signal Efficiency	Purity
Contained Events	–	355.5	1269.8	1099.7	135.7	34.4	–	21.9%
$s/\sqrt{b}$ opt	0.99	61.8	0.1	0.1	0.0	0.0	17.4%	99.9%
$s/\sqrt{s+b}$ opt	0.45	206.8	7.6	6.8	0.7	0.1	58.2%	96.4%

- ***40% Better Electron Efficiency for same background.***

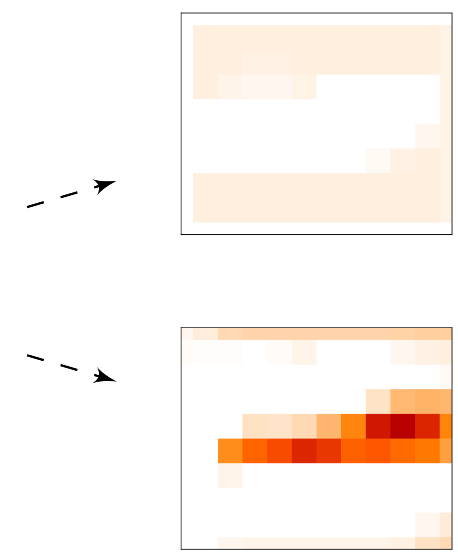


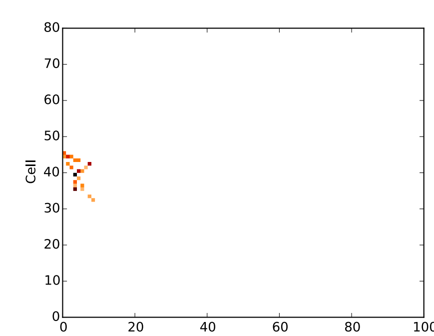
Muon Neutrino
DIS CC



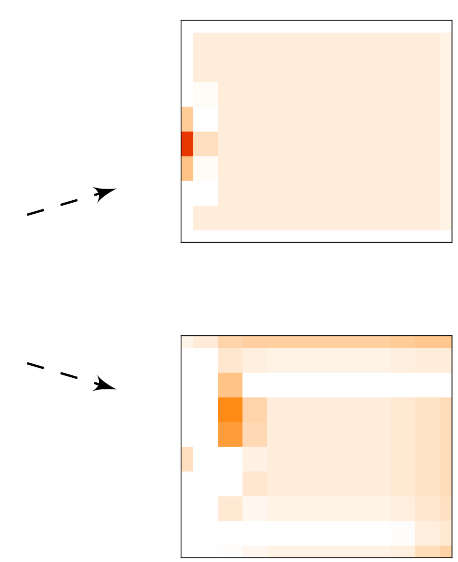


Muon Neutrino
QE CC



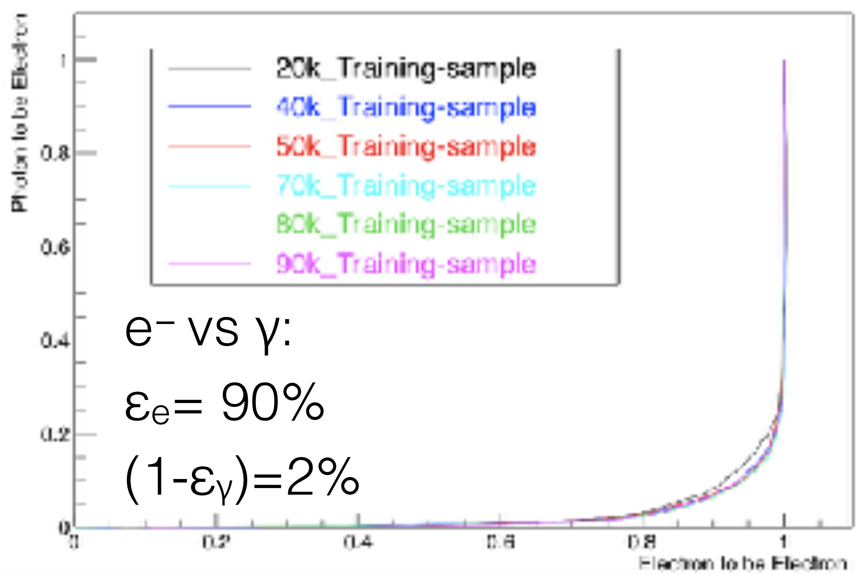


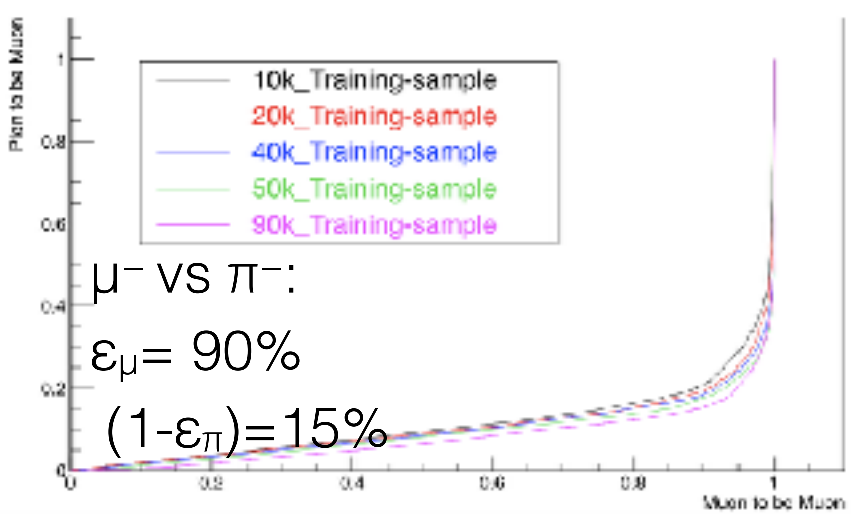
Muon Neutrino
NC

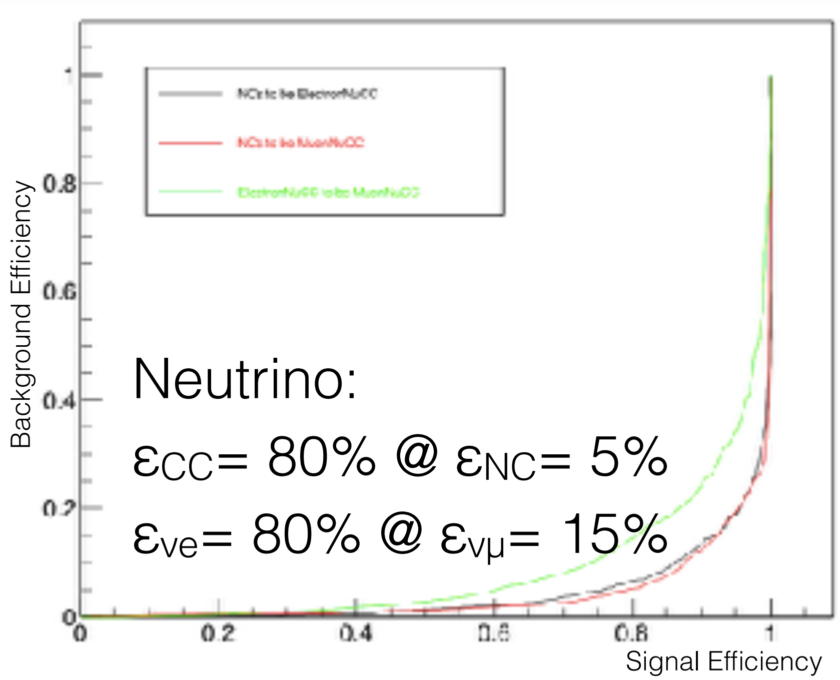


<u>http://arxiv.org/pdf/1604.01444.pdf</u>

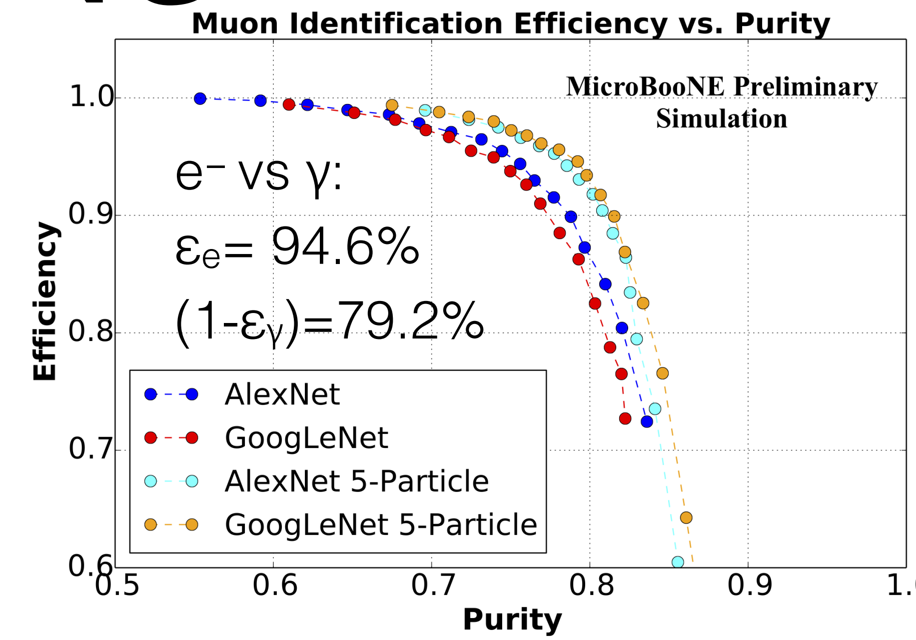
LArTPC CNNs

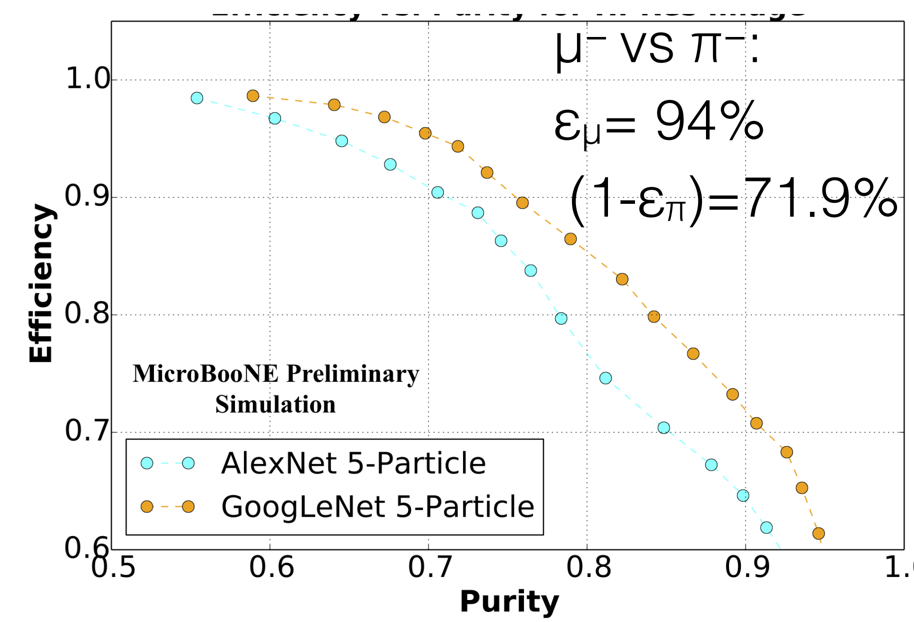


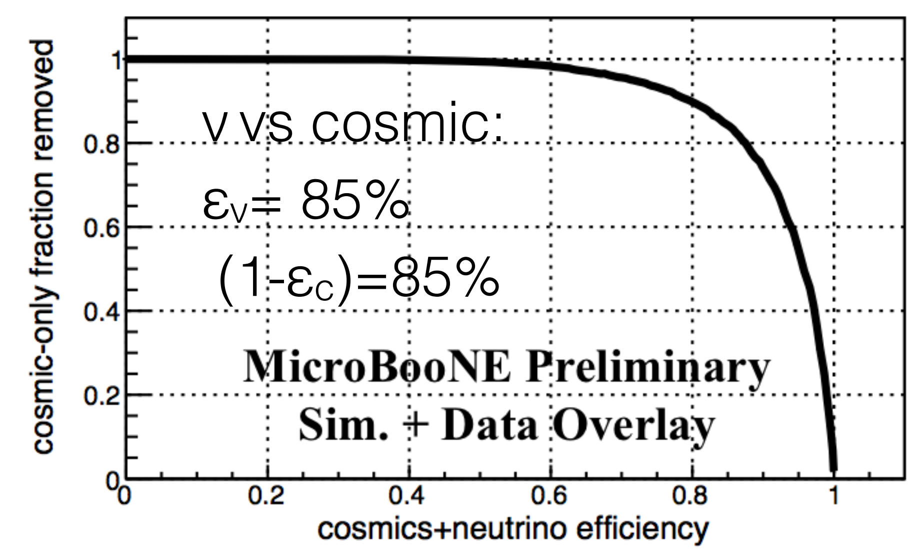




- First Studies Based on ***out-of-box CNN***
- ***MicroBooNE*** Note and ***LArIAT*** (independent by AF):
  - ***Particle ID***: MicroBooNE and LArIAT
  - ***Neutrino ID***: LArIAT
  - ***Cosmic Rejection***: MicroBooNE
  - ***Energy Regression***: LArIAT (in progress)
- Observations: LArTPC easier than image classification. ***Shallower networks sufficient***.

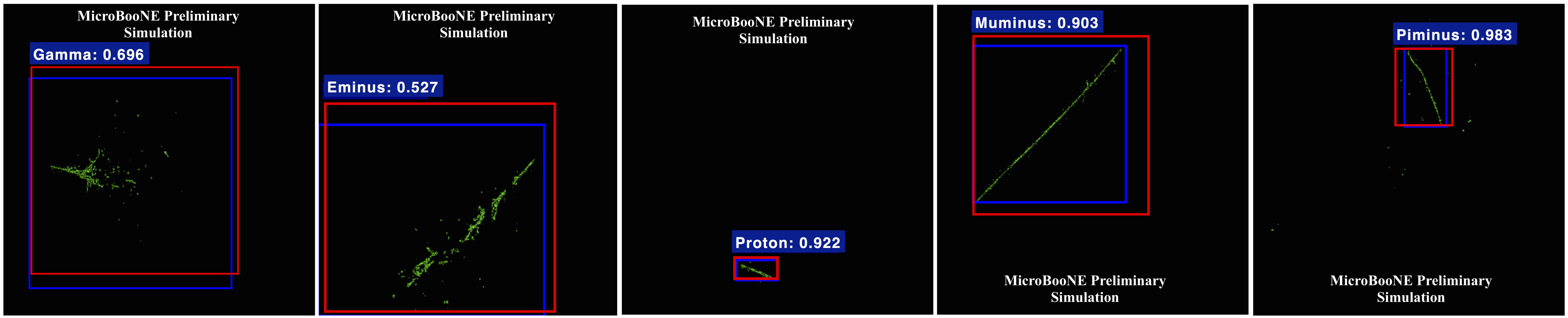


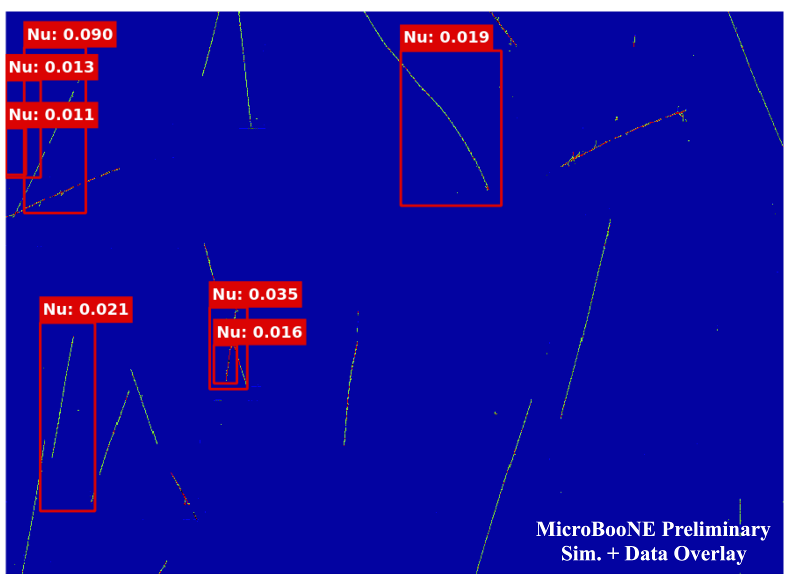




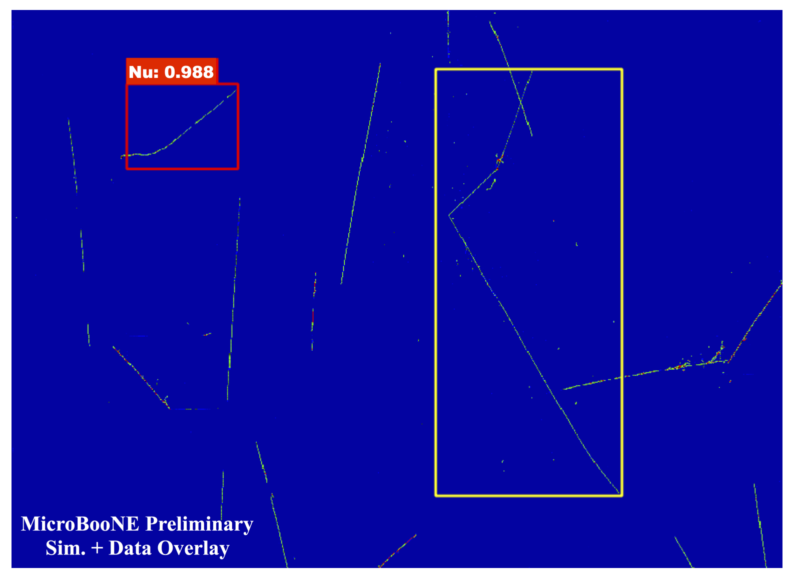
Semantic Segmentation

MicroBooNE

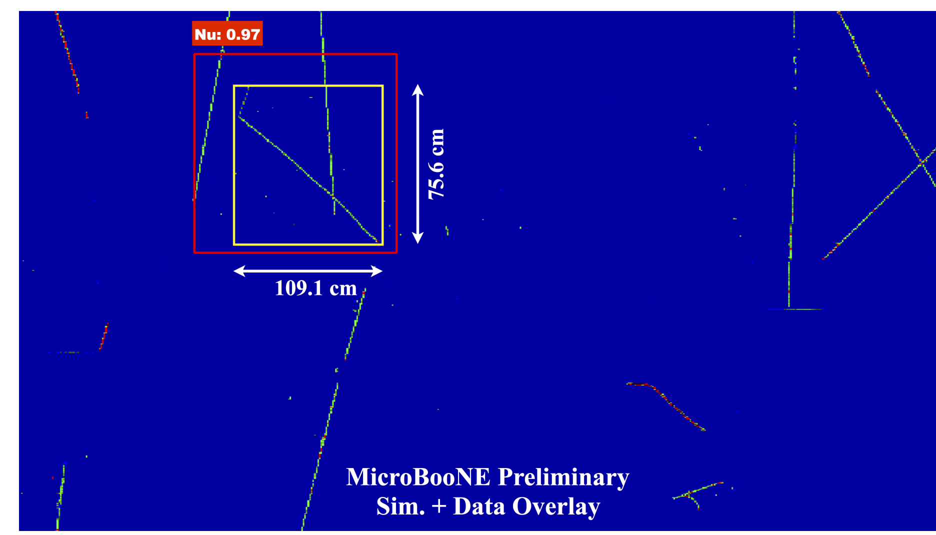




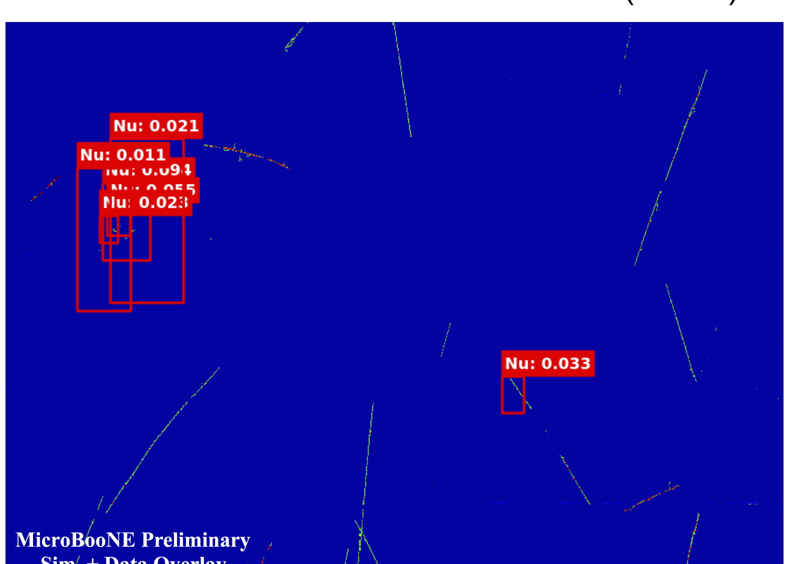
Low score for real Cosmics (data)

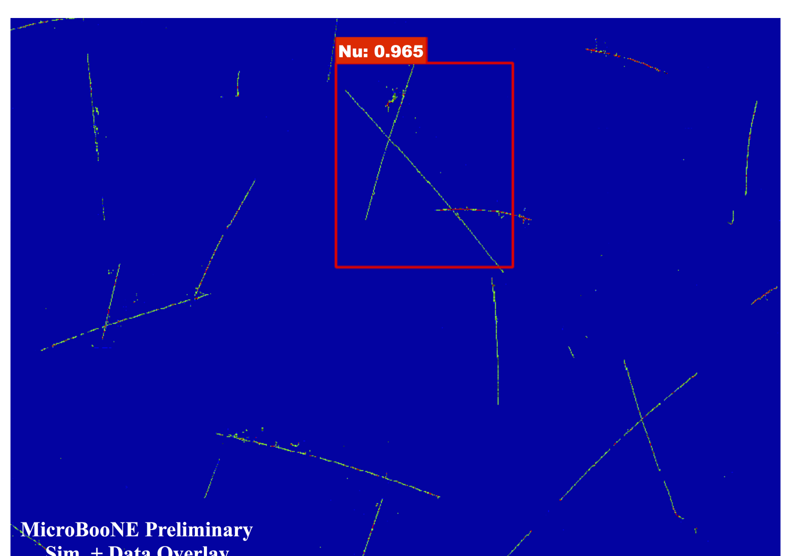


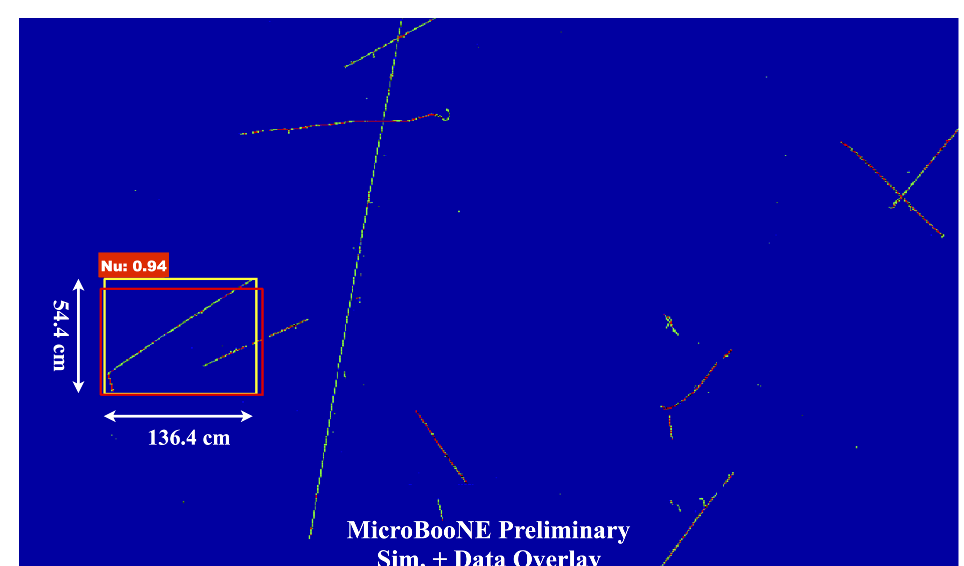
Reasonable Mistakes?



High Score for overlaid neutrinos (sim)



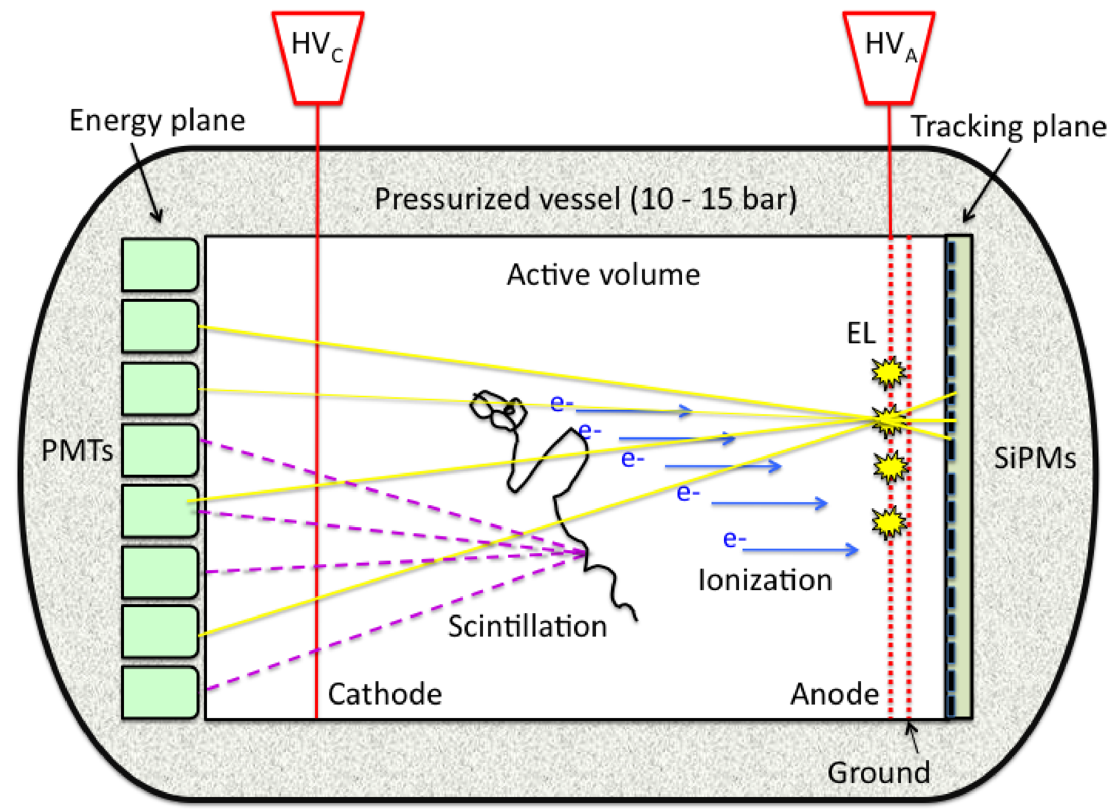


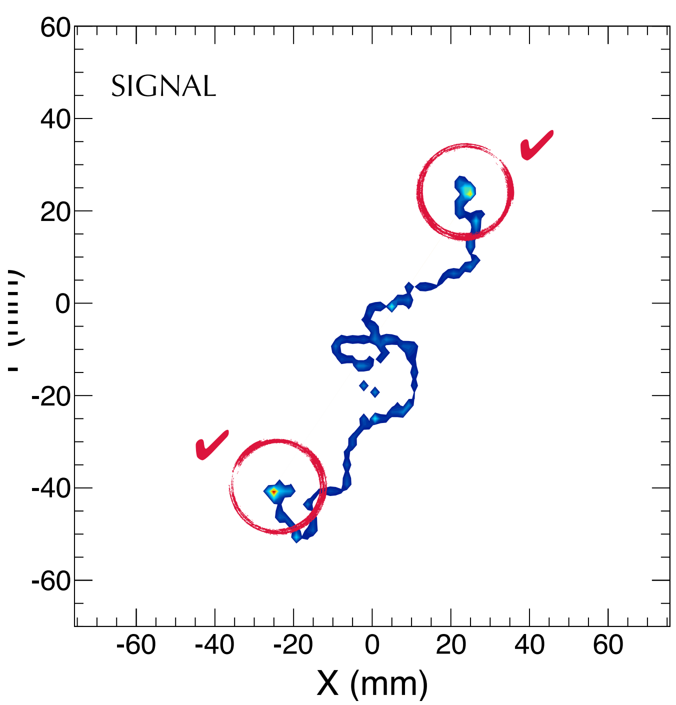


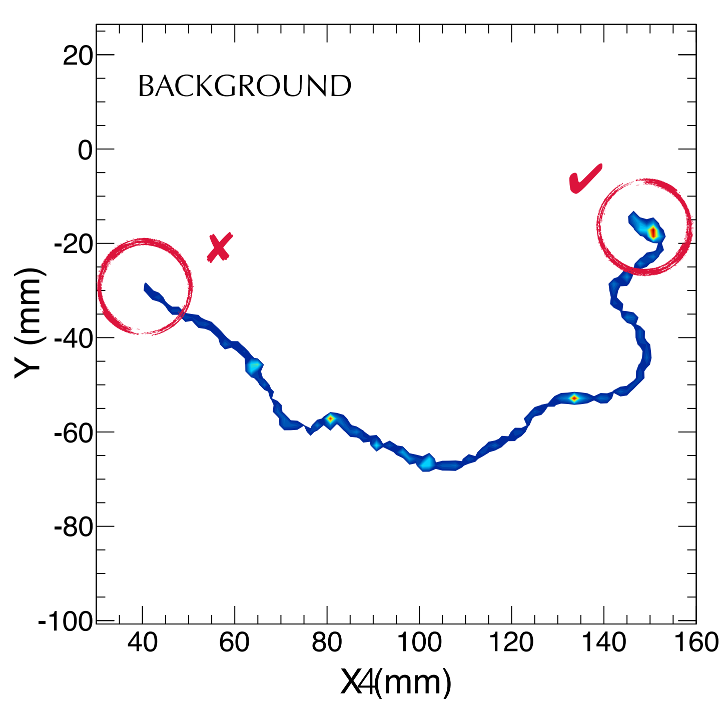
NEXT Experiment

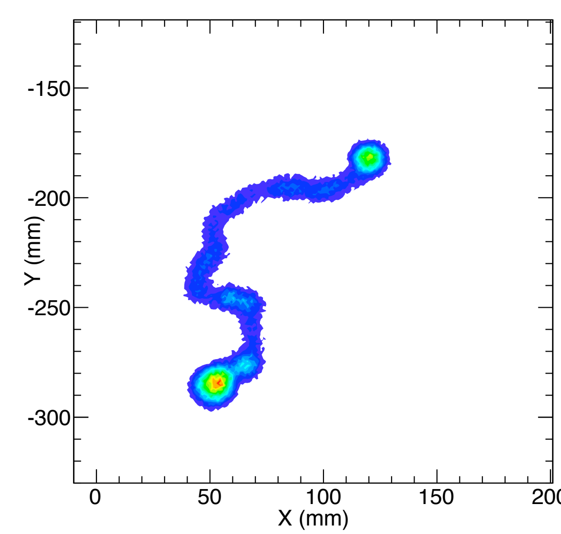
(J. Renner, J.J. Gomez, …, AF)

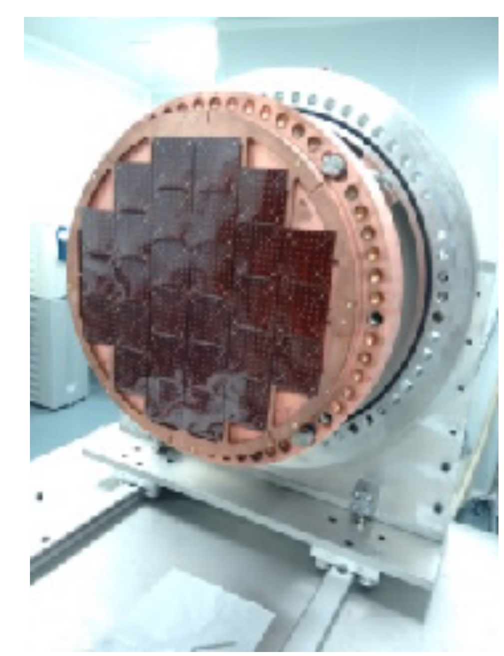
- ***Neutrinoless Double Beta Decay*** using Gas TPC/SiPMs
- Signal: 2 Electrons. Bkg: 1 Electron.
- Hard to distinguish due to ***multiple scattering***.
- ***3D readout***… candidate for 3D Conv Nets.
- Just a handful of signal events will lead to ***noble prize***
  - Can we trust a DNN at this level?







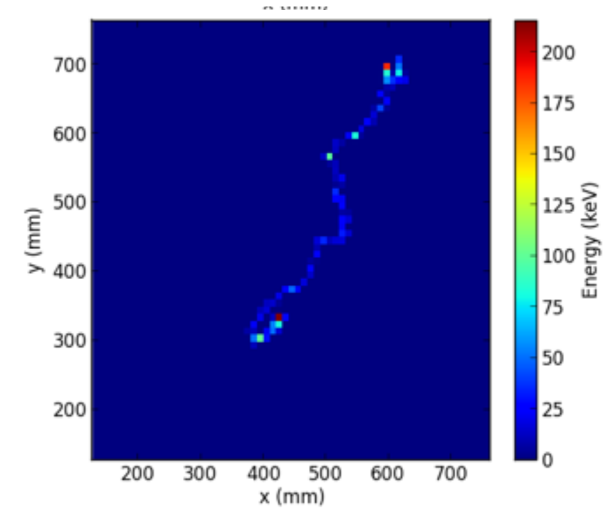


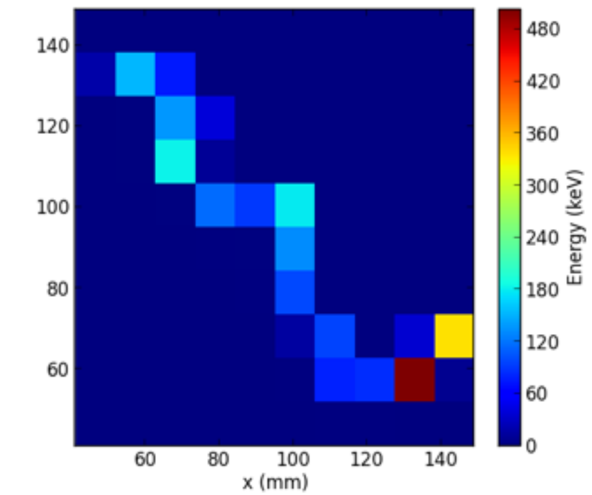


NEXT Detector Optimization

- Idea 1: use DNNs to ***optimize detector***.
  - Simulate data at different resolutions
  - Use DNN to quickly/easily assess best performance for given resolution.

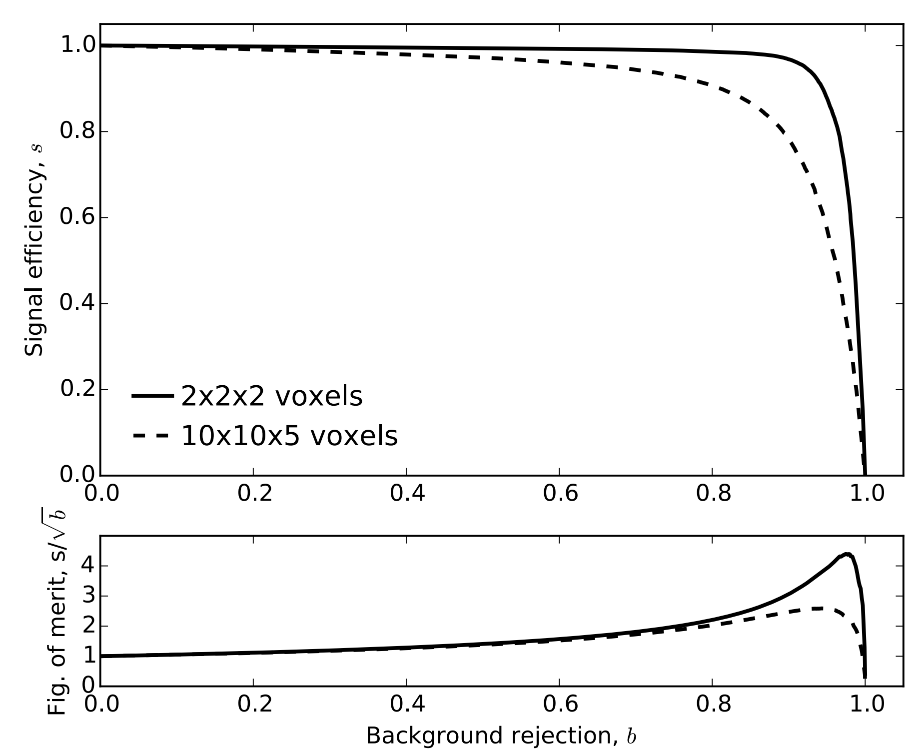
Analysis	Signal eff. (%)	B.G. accepted (%)
DNN analysis (2 x 2 x 2 voxels)	86.2	4.7
Conventional analysis (2 x 2 x 2 voxels)	86.2	7.6
DNN analysis (10 x 10 x 5 voxels)	76.6	9.4
Conventional analysis (10 x 10 x 5 voxels)	76.6	11.0





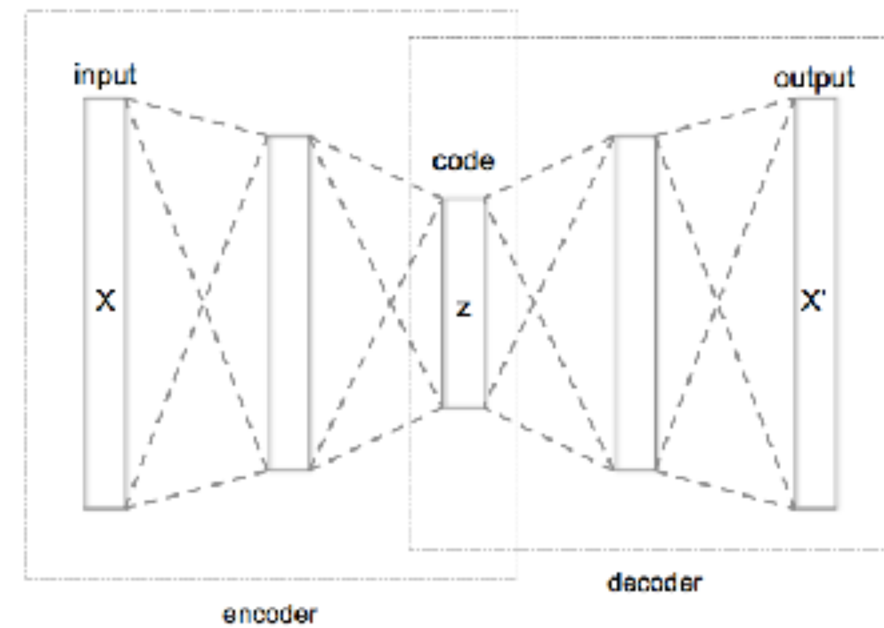
- Idea 2: ***systematically study*** the relative importance of various physics/detector effects.
  - Start with simplified simulation. Use DNN to assess performance.
  - Turn on effects one-by-one.

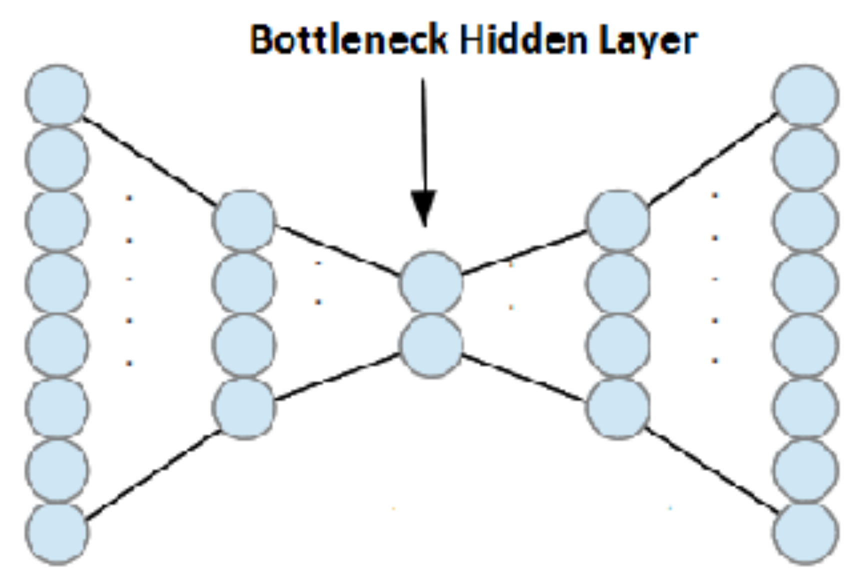
2x2x2 voxels	Run description	Avg. accuracy (%)
	Toy MC, ideal	99.8
	Toy MC, realistic $0\nu\beta\beta$ distribution	98.9
	Xe box GEANT4, no secondaries, no E-fluctuations	98.3
	Xe box GEANT4, no secondaries, no E-fluctuations, no brem.	98.3
	Toy MC, realistic $0\nu\beta\beta$ distribution, double multiple scattering	97.8
	Xe box GEANT4, no secondaries	94.6
	Xe box GEANT4, no E-fluctuations	93.0
	Xe box, no brem.	92.4
	Xe box, all physics	92.1
	NEXT-100 GEANT4	91.6
10x10x5 voxels		
	NEXT-100 GEANT4	84.5



Semi-supervised Learning

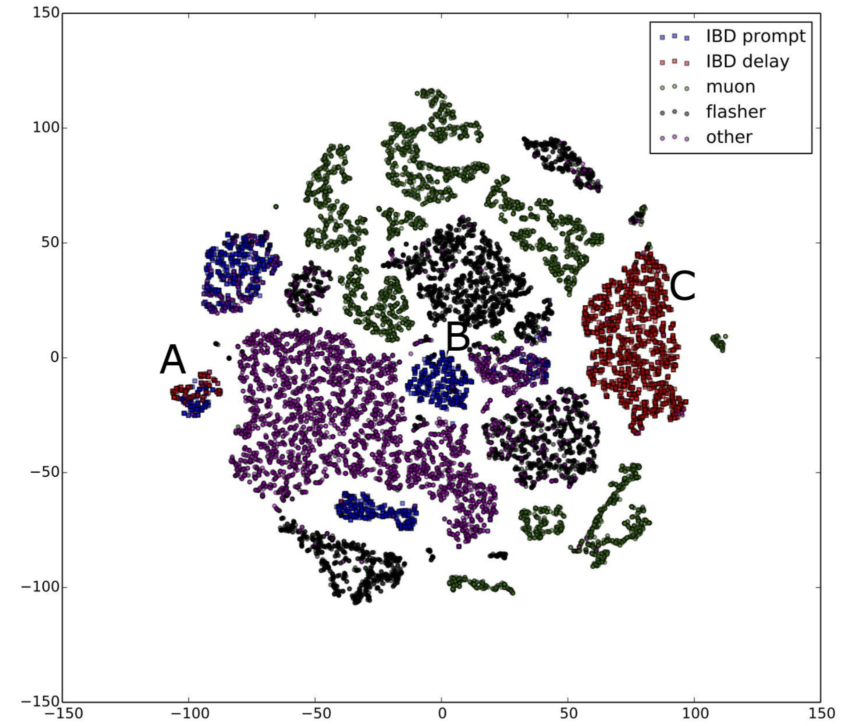
- Basic idea: Train network to ***reproduce the input***.
- Example: ***Auto-encoders***
  - ***De-noising auto-encoders***: add noise to input only.
  - ***Sparse auto-encoders***:
    - ***Sparse latent (code) representation*** can be exploited for ***Compression, Clustering, Similarity testing,*** …
  - ***Anomaly Detection***
    - Reconstruction Error
    - Outliers in latent space
  - ***Transfer Learning***
    - Small labeled training sample?
      - Train auto-encoder on large unlabeled dataset (e.g. data).
      - Train in latent space on small labeled data. (e.g. rare signal MC).
- Easily think of a dozen applications.



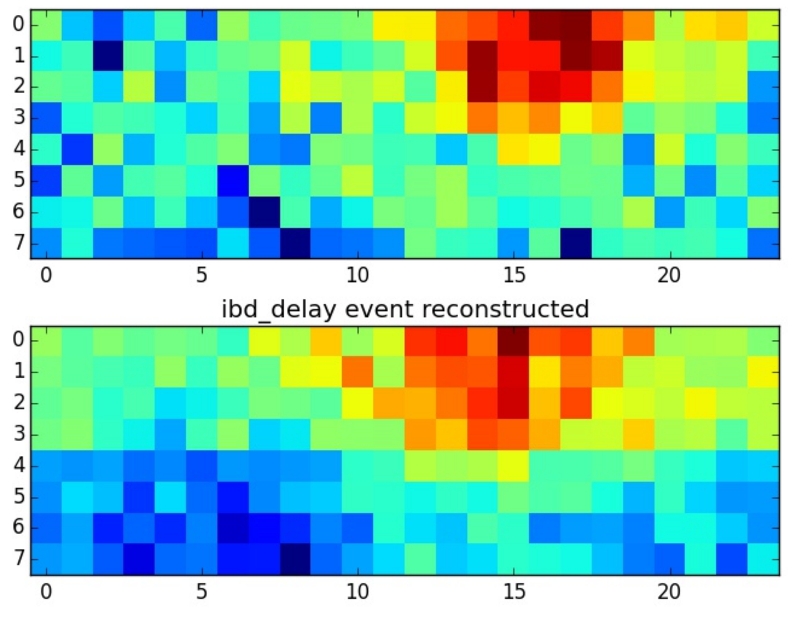


Learning Representations

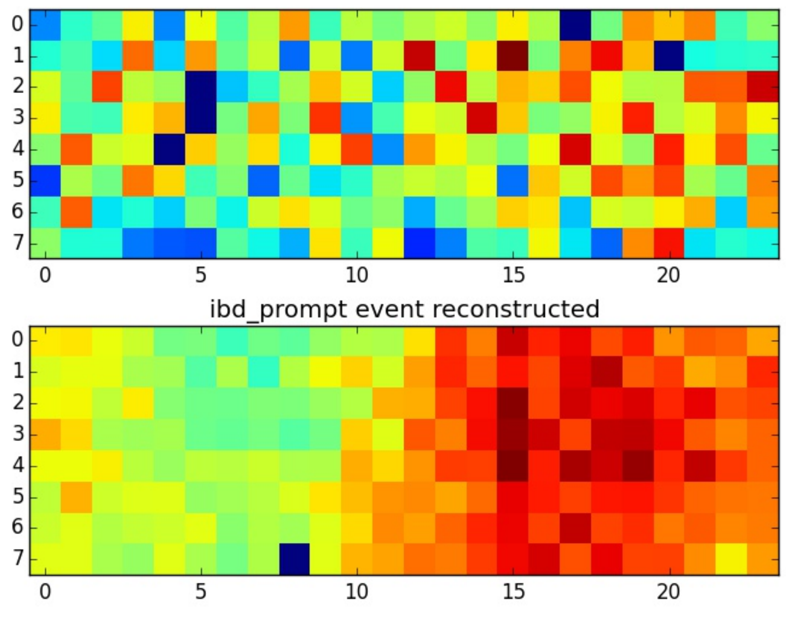
- Example: ***Daya Bay Experiment*** (Evan Racah, et al)
- Input: 8 x 24 PMT unrolled cylinder. ***Real Data (no simulation)***
- 2 Studies:
    - ***Supervised CNN Classifier***
        - Labels from standard analysis: Prompt/Delayed Inverse Beta Decay, Muon, Flasher, Other.
    - ***Convolutional Auto-encoder*** (semi-supervised)
        - Clearly separates muon and IBD delay ***without any physics knowledge***.
        - Potentially could have ID'ed problematic data (e.g. flashers) much earlier.



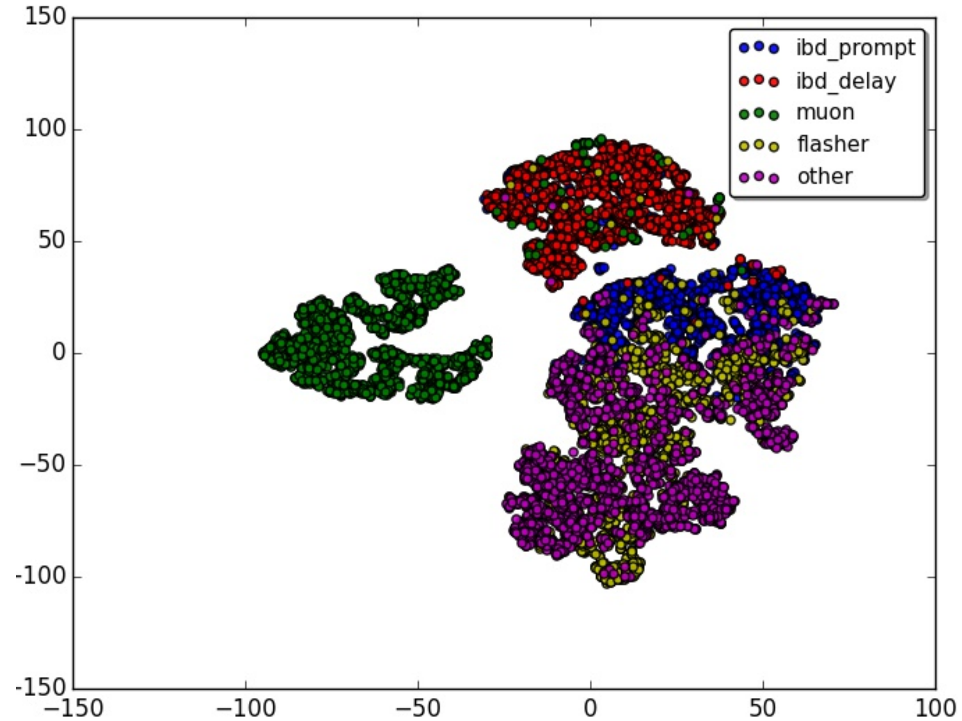
t-SNE reduction of 26-dim representation of the last fully connected layer.



(a) Example of an "IBD delay" event



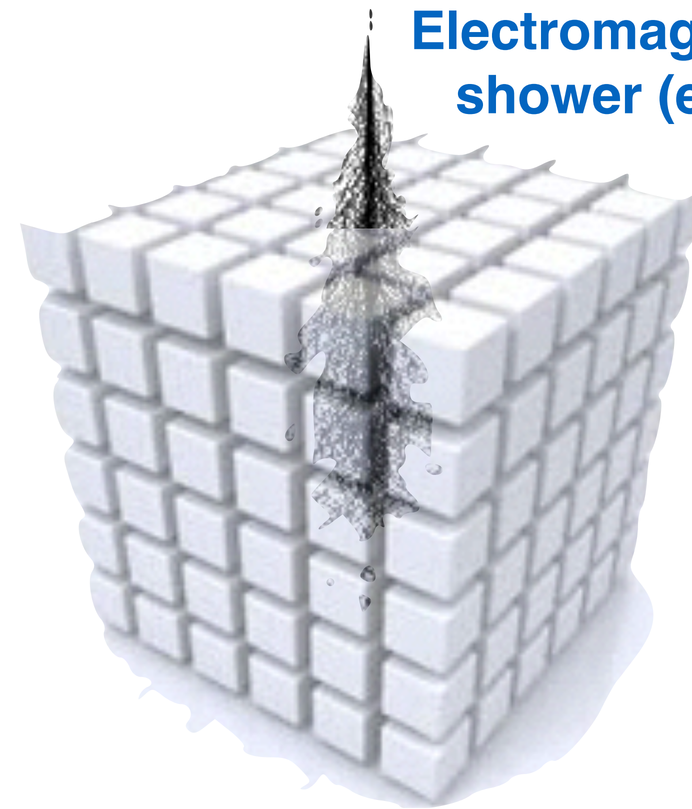
(b) Example of an "IBD prompt" event



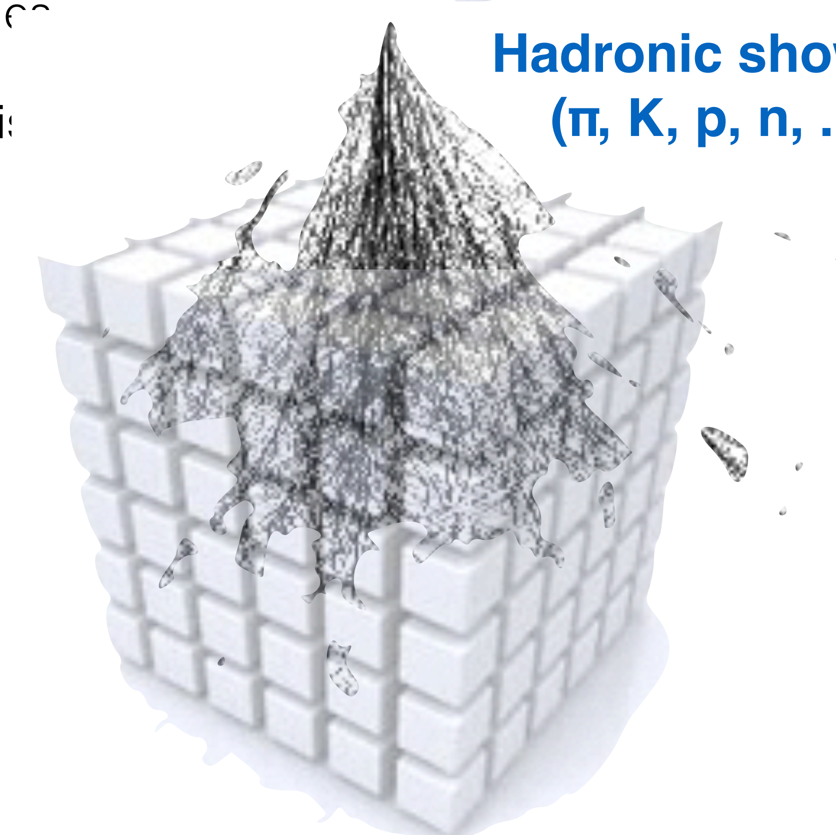
t-SNE reduction of 10 parameter latent representation.

Generative Models

- ***Likelihood Approximation relies simulation***
  - Most ***computationally expensive*** step, so any speedup has huge impact.
  - More generally, ***simulation based on data*** would be a powerful tool.
    - For example, we can build a Hadronization model purely from data.
- DNNs Generative Models enable building simulations purely from examples.
  - ***Generative Adversarial Nets*** (Goodfellow, et. al. arxiv:1406.2661). Simultaneously train 2 Networks:
    - ***Discriminator*** (D) that tries to distinguish output and real exampl
    - ***Generator*** (G) that generate the output that is difficult to distingui
  - ***Variational Auto-encoders***:
    - Learn a ***latent variable probabilistic model*** of the input dataset.
    - ***Sample latent space*** and use ***decoder to generate data***.
- ***Particle showering*** is ***slowest*** part of the micro-physics simulation…
  - Various techniques for fast showering (e.g. shower template libraries) are common.
  - DNN Generative Models are being pursued inside the experiments (K. Cranmer, G. Louppe, …) for this task…



Electromagnetic shower (e, γ)



Hadronic shower (π, K, p, n, ..)

Learning Particle Physics by Example: Location-Aware Generative Adversarial Networks for Physics Synthesis

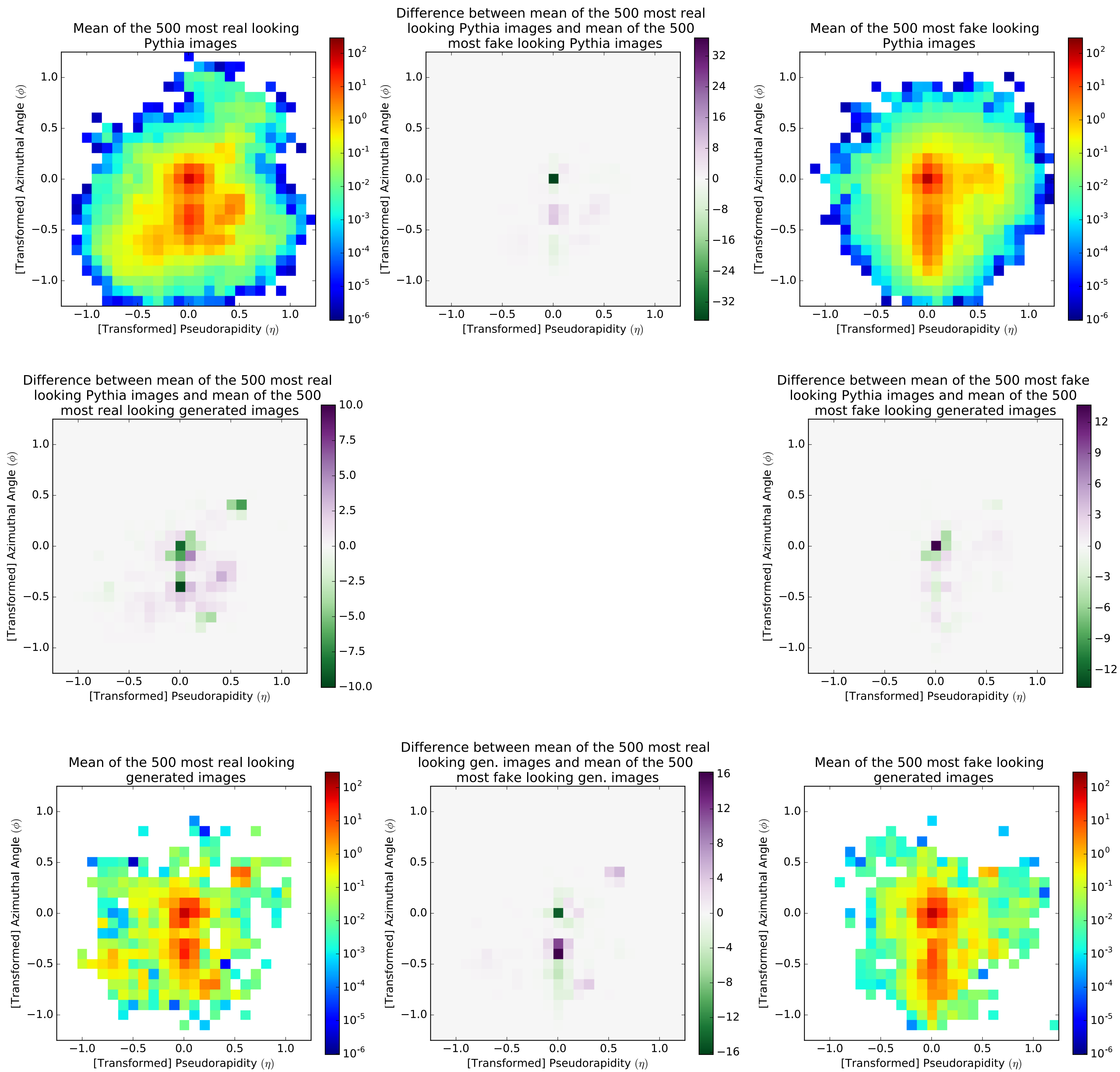
Luke de Oliveira[a], **Michela Paganini**[a,b], **and Benjamin Nachman**[a]

[a] *Lawrence Berkeley National Laboratory, 1 Cyclotron Rd, Berkeley, CA, 94720, USA*
[b] *Department of Physics, Yale University, New Haven, CT 06520, USA*

E-mail: lukedeoliveira@lbl.gov, michela.paganini@yale.edu, bnachman@cern.ch

ABSTRACT: We provide a bridge between generative modeling in the Machine Learning community and simulated physical processes in High Energy Particle Physics by applying a novel Generative Adversarial Network (GAN) architecture to the production of *jet images* – 2D representations of energy depositions from particles interacting with a calorimeter. We propose a simple architecture, the Location-Aware Generative Adversarial Network, that learns to produce realistic radiation patterns from simulated high energy particle collisions. The pixel intensities of GAN-generated images faithfully span over many orders of magnitude and exhibit the desired low-dimensional physical properties (*i.e.*, jet mass, n-subjettiness, etc.). We shed light on limitations, and provide a novel empirical validation of image quality and validity of GAN-produced simulations of the natural world. This work provides a base for further explorations of GANs for use in faster simulation in High Energy Particle Physics.

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



Mean of the 500 most real looking Pythia images
Difference between mean of the 500 most real looking Pythia images and mean of the 500 most fake looking Pythia images
Mean of the 500 most fake looking Pythia images
Difference between mean of the 500 most real looking Pythia images and mean of the 500 most real looking generated images
Difference between mean of the 500 most fake looking Pythia images and mean of the 500 most fake looking generated images
Mean of the 500 most real looking generated images
Difference between mean of the 500 most real looking gen. images and mean of the 500 most fake looking gen. images
Mean of the 500 most fake looking generated images
[Transformed] Azimuthal Angle (ϕ)
[Transformed] Pseudorapidity (η)
−1.0 −0.5 0.0 0.5 1.0
10^{2} 10^{1} 10^{0} 10^{-1} 10^{-2} 10^{-3} 10^{-4} 10^{-5} 10^{-6}
32 24 16 8 0 −8 −16 −24 −32
10.0 7.5 5.0 2.5 0.0 −2.5 −5.0 −7.5 −10.0
12 9 6 3 0 −3 −6 −9 −12
16 12 8 4 0 −4 −8 −12 −16

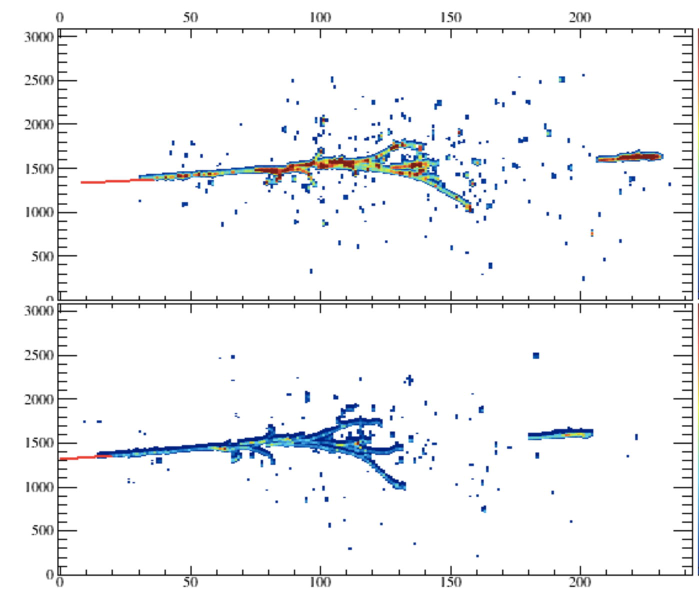
Plans…

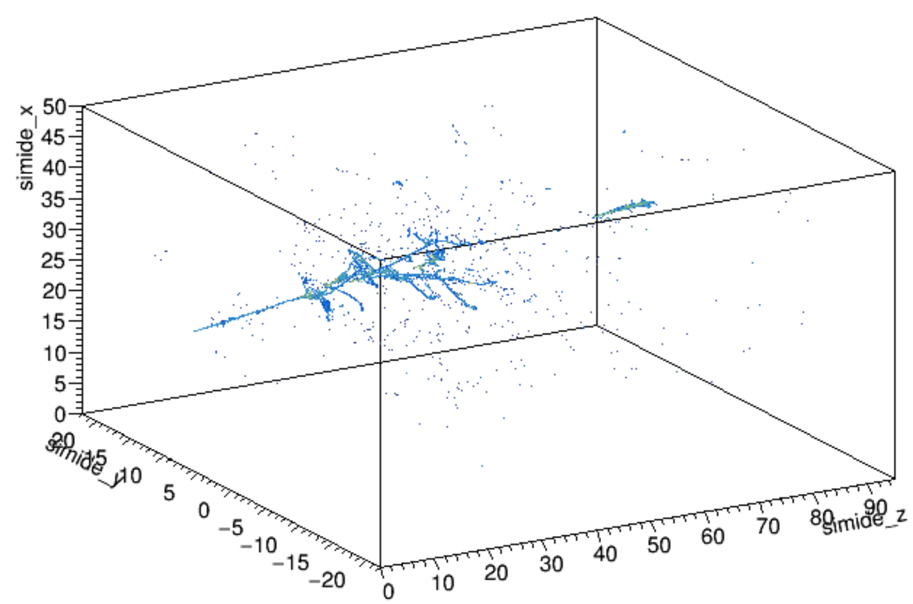
Public Datasets

- ***Biggest obstacles*** to DNN research is ***Data accessibility***.
    - Detector level studies require ***CPU intensive simulations***.
    - DNNs require large training sets with ***full level of detail*** (i.e. not 4-vectors).
    - Experiments have such samples, but they are not easily accessible and ***not public***.
    - ***Difficult to collaborate*** with DL community or other experiments.
- ***Public datasets*** *(Unveiling next week at DS@HEP Workshop at Fermilab)*:
    - We provide data, tools (e.g. fast data read), fully setup problems. Goal is build working groups around each dataset.
    - ***LArTPC*** (Sepideh Shahsavarani, AF): LArIAT detector. 1 M of every particle species (including neutrinos).
        - Challenges: Particle/Neutrino Classification and Energy Reco, Noise Suppression, 2D->3D.
    - ***Calorimetry*** (Maurizio Pierini, Jean-Roch Vlimant, Nikita Smirnov, AF): LCD Calorimeter.
        - Challenges: PID/Energy Reco. Simulation.
    - ***Tracking***
        - Simple 2D tracking data shown at Connecting the Dots will be used for DS@HEP.
        - *TrackingML/ACTS* (David Rousseau, Andreas Salzberger, …) HL-LHC like detector/environment.
    - ***CMS Jets***: Full Reco Simulated Jets for boosted object and jet ID

LArTPC 2D to 3D

- LArTPC wire readout necessary due to heat load.
  - Full Pixelized readout would give ~ N^2 datapoint/time slice
  - Wire readout give ~2N datapoint/time
- Information loss is “recovered” in reconstruction by assuming particle interaction topologies (track, shower, …)
- Tomographic approach (Wirecell) “resolves” ambiguities through costly Markov Chain MC
- Perhaps a DNN can learn the topologies and infer a 3D image
  - Imagine an Auto-encoder like setup with
    - Input: 2x (or 3x) 2D images
    - Output: 3D image.





Science Fiction?

- Imagine in next 10 years DNN lives up to the hype…
  - We've proven DNNs gets us better, faster, easier software… and hardware.
  - Industry investment in DNNs has yielded significant gain over Moore's Law
    - Custom DL/neuromorphic chips and HPCs
    - Software Frameworks
    - Cloud Services
    - Consultants:
      - Data Scientists: DL reduces need for domain-specific expertise (e.g. in biology now).
      - Data Engineers: low level optimization, deployment, operation…
  - Actually, all of these already exist!

- Large portions of HEP code replaced by deep neutral network architecture and weights.
  - HEP Software Frameworks built on top of DL Frameworks.
  - To DL systems, our computing looks like everyone else's… e.g. other sciences.
- Optimization, deployment, operations handled by professional Data Engineers.
- Trigger implemented in custom inference systems built from commodity hardware.
- Computation performed on DL Clouds and scientific HPCs.
- DNNs designed and trained in collaboration with professional Data Scientists.
- HEP PhDs trained/funded by industry to apply DL to HEP and then transition to industry.

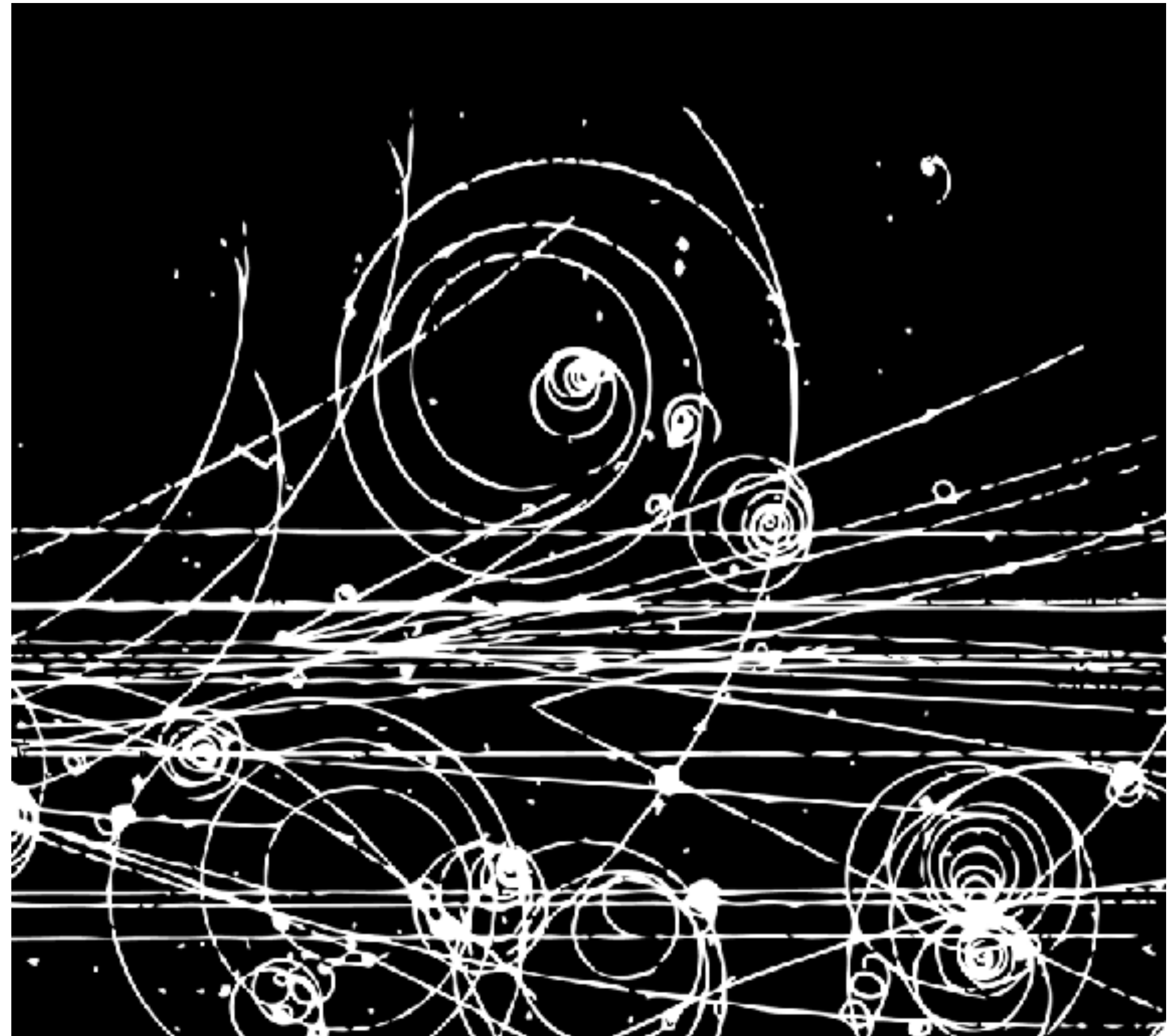
Final Thoughts

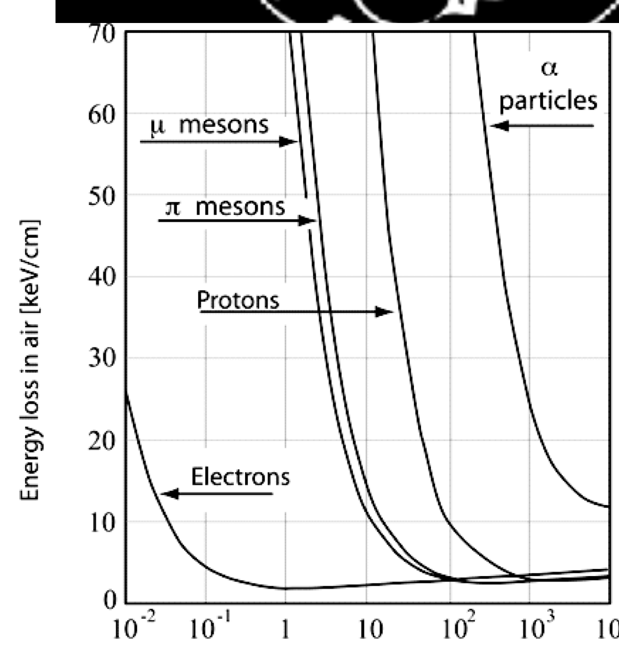
- For decades HEP had the biggest datasets… was overtaken by industry mid-2000's… may be the biggest datasets again in the next decade.
- Computing for HL-LHC will be prohibitively expensive unless we find some clever techniques.
  - Deep Learning and Neuromorphic processors are a promising solution.
- Deep Learning can change how science is done.
  - Improve performance. Save time and money.
  - Mitigate stalling of Moore's law.
  - Use most recent hardware.
  - Allow scientists to focus on concepts rather than implementation.Deep Learning serves as a tool to optimize designs and traditional techniques, indirectly improving our measurements.
- If we want to be ready for the DL revolution in 10 years, we need to do R&D now.
- We can't forget that DL can complicated things:
  - Systematics. Data/MC agreement.
  - Generate large independent training and calibration samples.
  - New complicated "release", production, and analysis cycles/work-flows.

Extra Slides

How do we “see” particles?

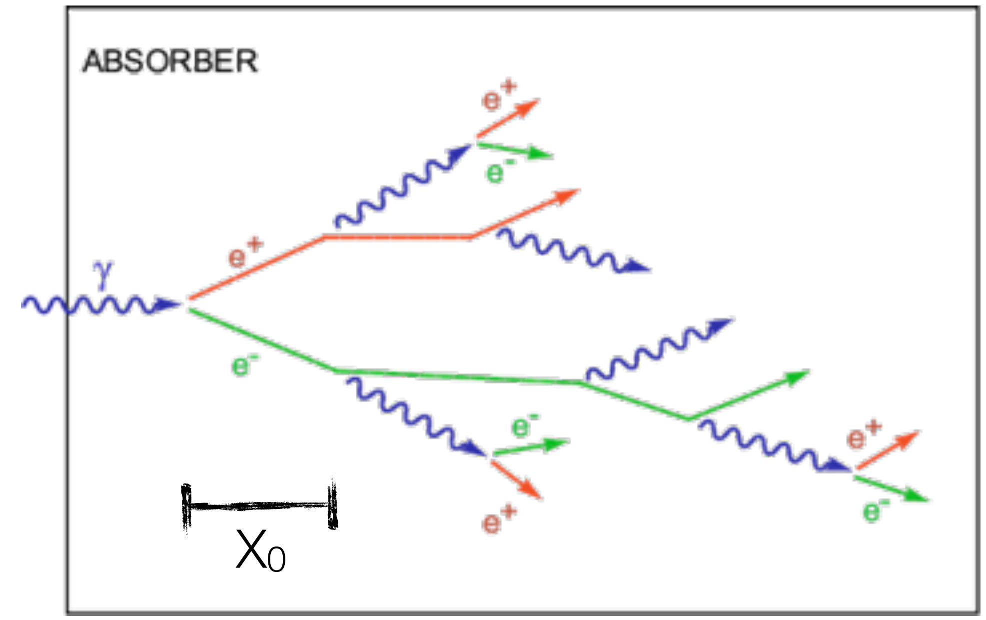
- ***Charged particles ionize media***
  - Image the ions.
    - In ***Magnetic Field*** the ***curvature*** of trajectory ***measures momentum***.
    - Momentum resolution degrades as less curvature: σ(p) ~ c p ⊕ d.
      - d due to multiple scattering.
  - Measure ***Energy Loss*** (~ # ions)
    - dE/dx = Energy Loss / Unit Length = f(m, v) = Bethe-Block Function
      - Identify the particle type
    - ***Stochastic process*** (Laudau)
  - Loose all energy → range out.
    - Range characteristic of particle type.

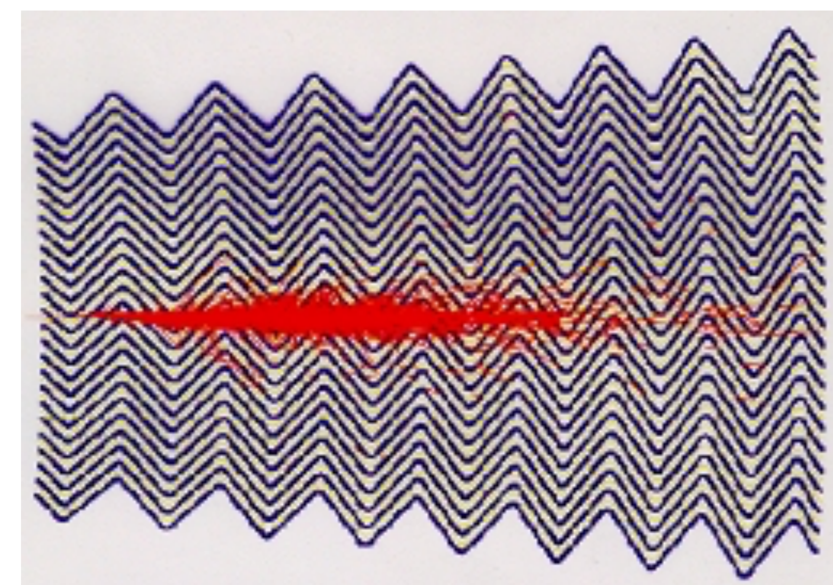


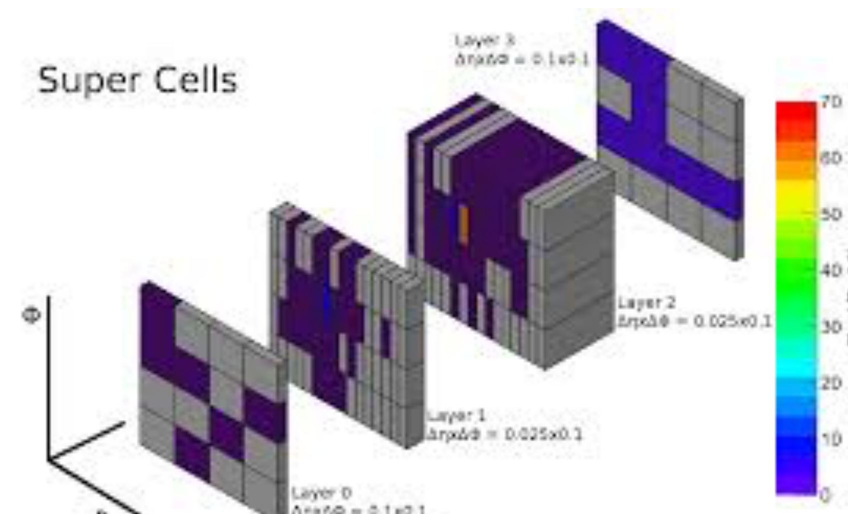


How do we “see” particles?

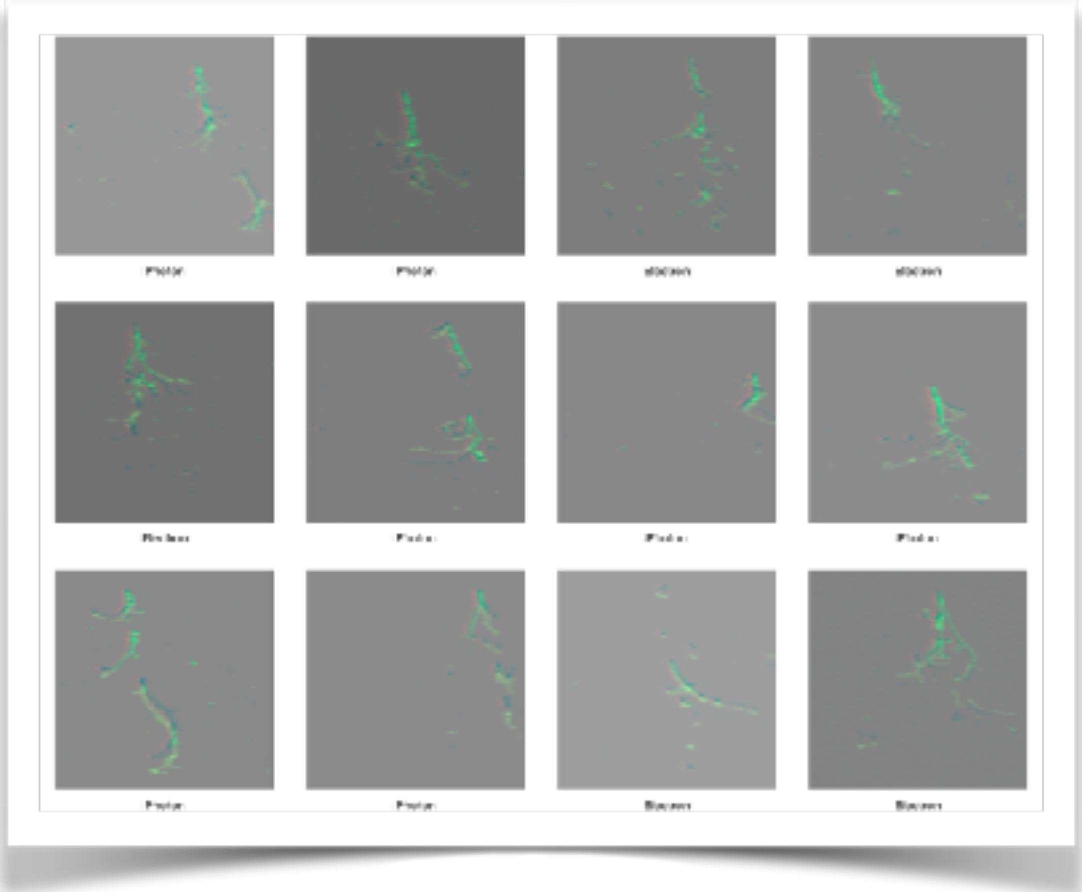
- Particles deposit their energy in a ***stochastic process*** know as ***“showering”***, secondary particles, that in turn also shower.
  - Number of secondary particles ~ Energy of initial particle.
  - Energy resolution improves with energy: $\sigma(E) / E = a/\sqrt{E} \oplus b/E \oplus c$.
    - a = sampling, b = noise, c = leakage.
  - Density and Shape of shower characteristic of type of particle.
- ***Electromagnetic calorimeter***: Low Z medium
  - ***Light particles***: electrons, photons, $\pi^0 \rightarrow \gamma\gamma$ interact with electrons in medium
- ***Hadronic calorimeters***: High Z medium
  - ***Heavy particles***: Hadrons (particles with quarks, e.g. charged pions/protons, neutrons, or jets of such particles)
    - Punch through low Z.
    - Produce secondaries through strong interactions with the nucleus in medium.
    - Unlike EM interactions, not all energy is observed.

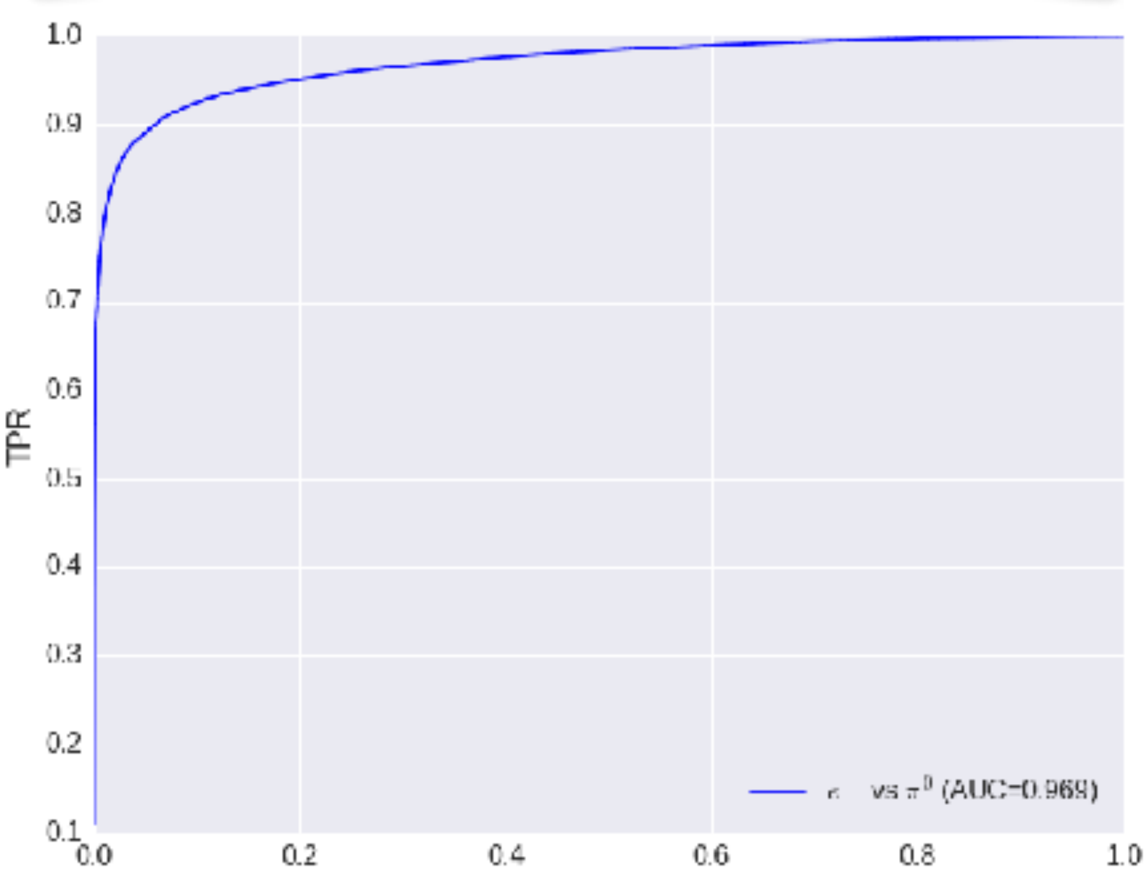






Large LArTPC Dataset

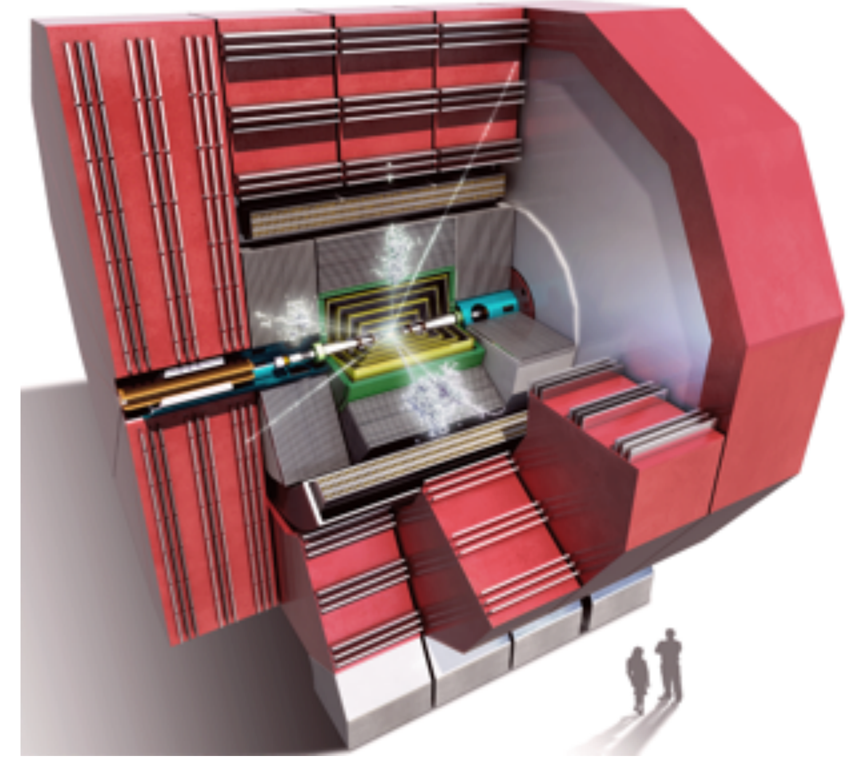




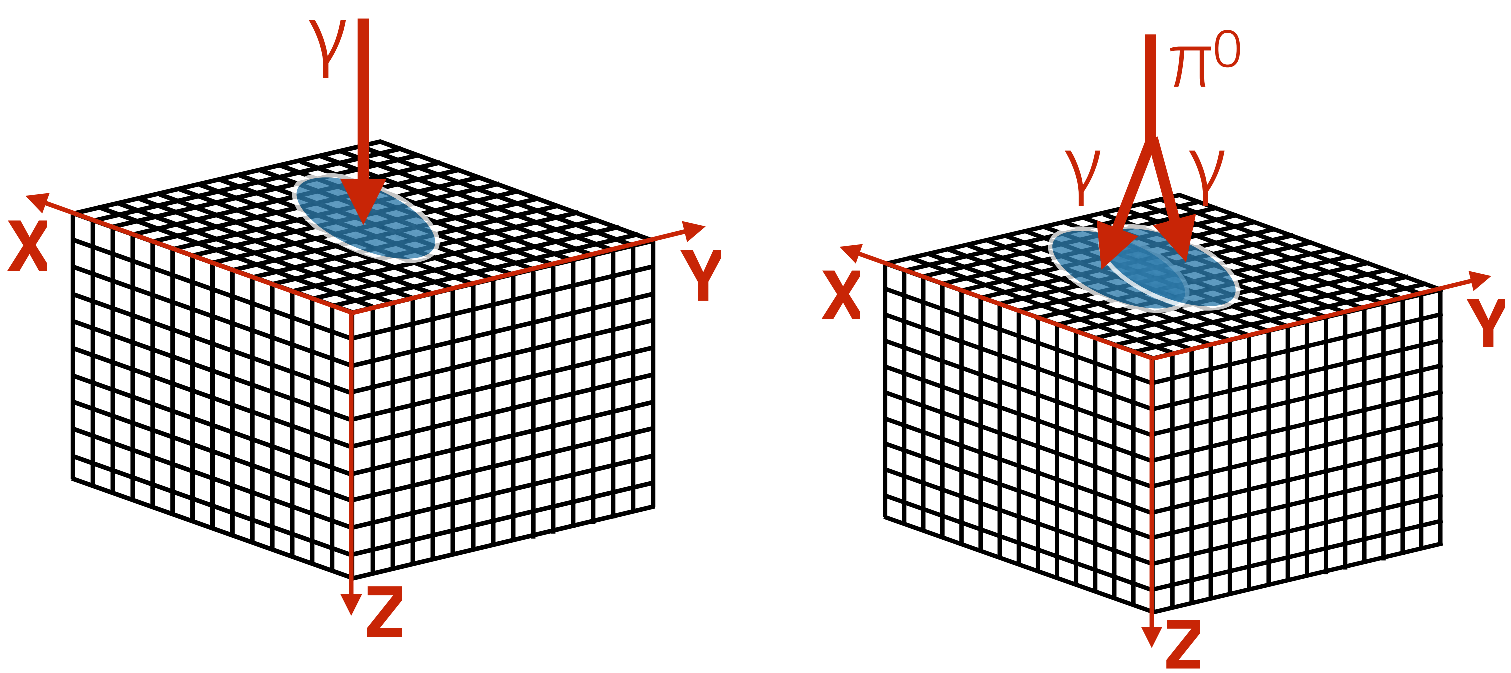
- Training samples have been at best ~100k examples…. usually much less.
- My students (S. Shahsavarani and G. Hilliard) simulated a huge sample of LArTPC events (LArIAT Detector).
  - Necessitated by Energy Regression studies.
  - 1 M of every particle species: $e^{\pm}$, $p^{\pm}$, $K^{\pm}$, $\pi^{\pm}$, π^{0}, $\mu^{\pm}$, γ, ν_e, ν_μ, ν_τ
  - Flat Energy distribution.
  - Will soon make these publicly available.
- Collaborators at UCI (P. Sadowski, et al) were able to get better performance by training for a week on this large dataset.

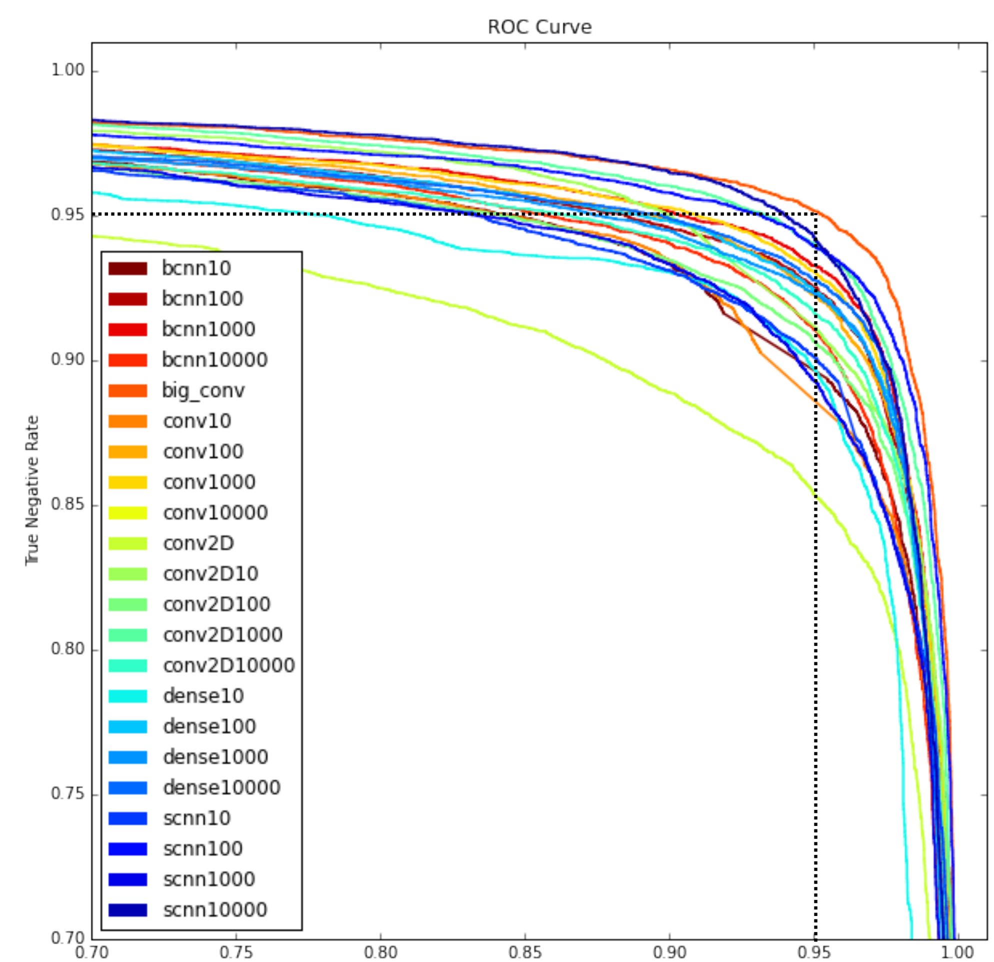
LCD Calorimeter

(Maurizio Pierini, Jean-Roch Vlimant, Nikita Smirnov, AF)



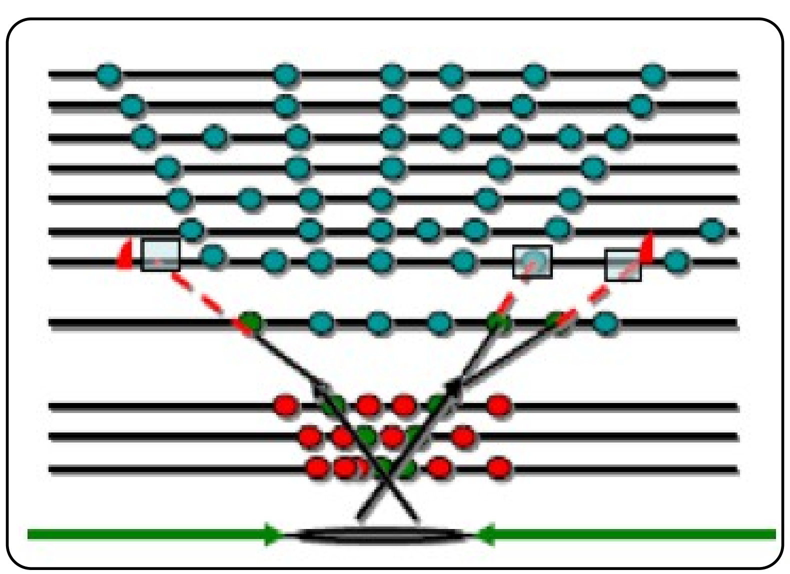
- CLIC is a proposed CERN project for a linear accelerator of electrons and positrons to TeV energies (~ LHC for protons)
    - Not a real experiment yet, so we (Maurizio Pierini, Jean-Roch Vlimant, Nikita Smirnov, AF) can simulate data and make it public.
- The LCD calorimeter is an array of absorber material and silicon sensors comprising the most granular calorimeter design available
    - Data is essentially a 3D image
- First studies, π^0 vs γ classification with various DNNs.
    - Much more to come…

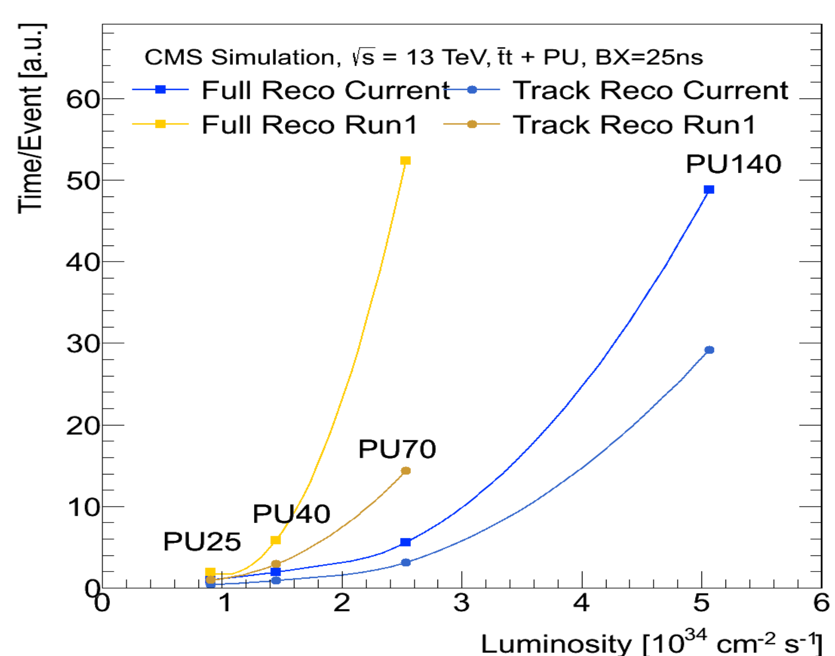


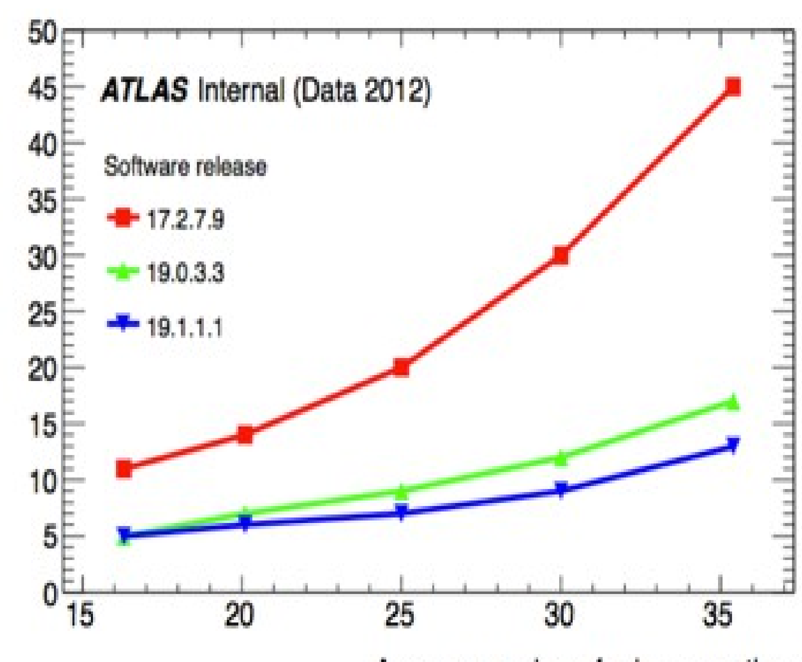


HL-LHC Tracking

- ***Tracking steps:*** hit prep, seeding, pattern recognition, track fitting, track cleaning
    - Highly optimized already for offline reconstruction for Run 2
        - ~30-50 proton collisions per beam crossing
        - 1 kHz data stream, processed offline.
- ***HL-LHC***: ~ 200 proton collisions per beam crossing
    - combinatorics cause pattern recognition time to grow exponentially
    - Busy environment requires tracking at 40 MHz for trigger
- Need Pattern Recognition that scales better with number of hits. Deep Learning?
- Again an obstacle to applying deep learning techniques is accessibility to the data.
- ***Tracking ML*** (David Rousseau, Andreas Salzberger, …, AF): Hoping to have ML community develop solutions, mirroring the HiggsML Challenge.
    - ACTS: Standalone version of ATLAS Tracking Simulation/Reconstruction developed for this challenge.

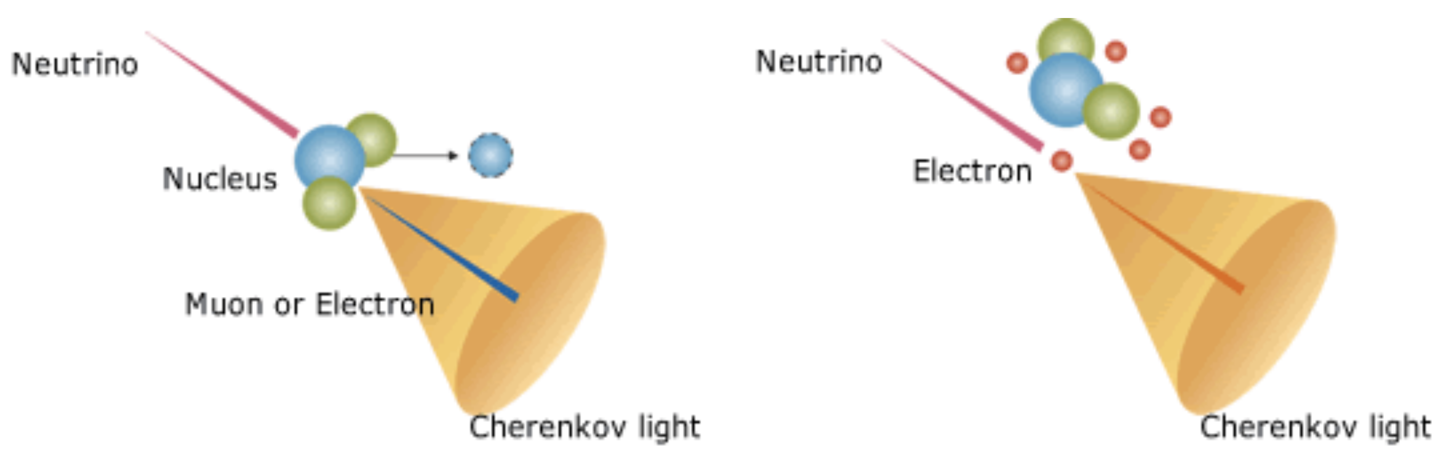




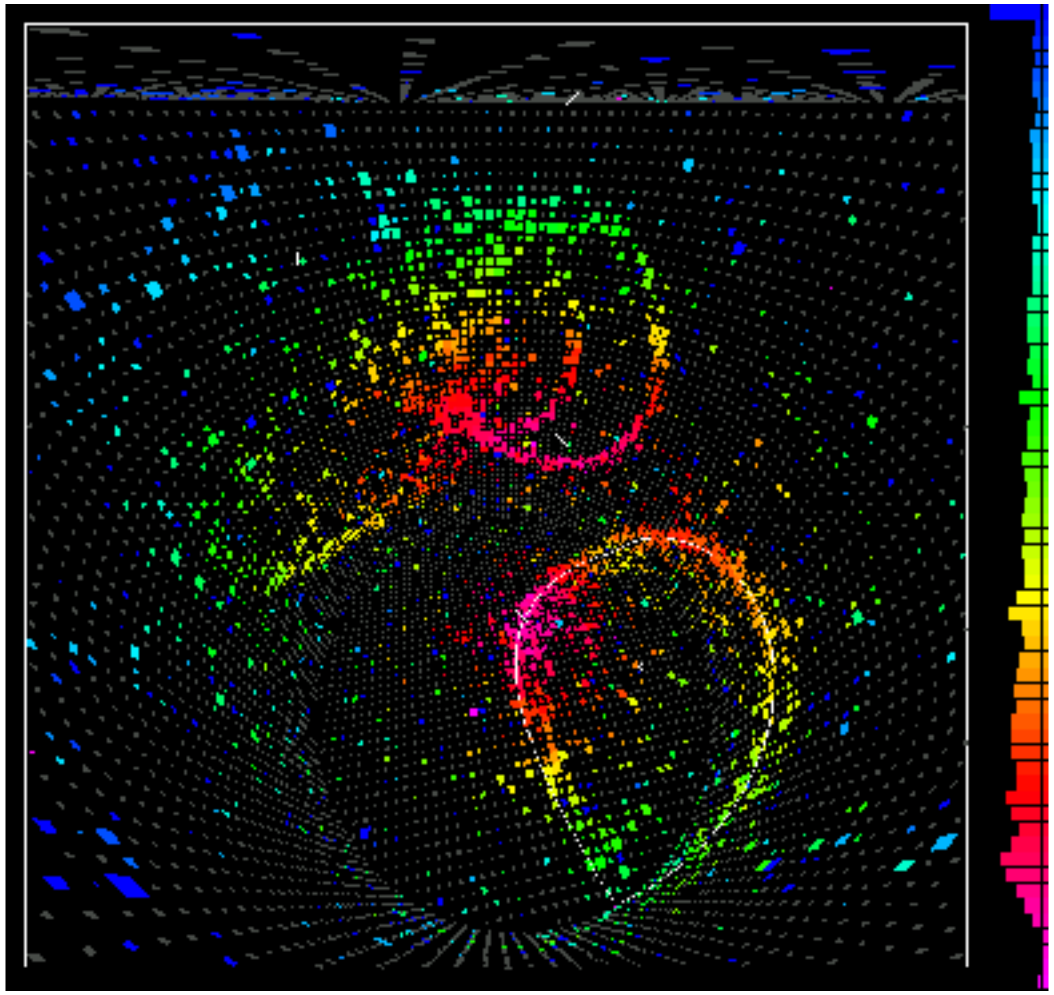


How do we “see” particles?

- Charged Particles traveling faster than speed of light in medium emit ***Cherenkov light*** (analogous to sonic boom).
  - Light emitted in cone, with angle function of speed and mass.
  - Depending on context, allow for particle identification and/or speed measurement.

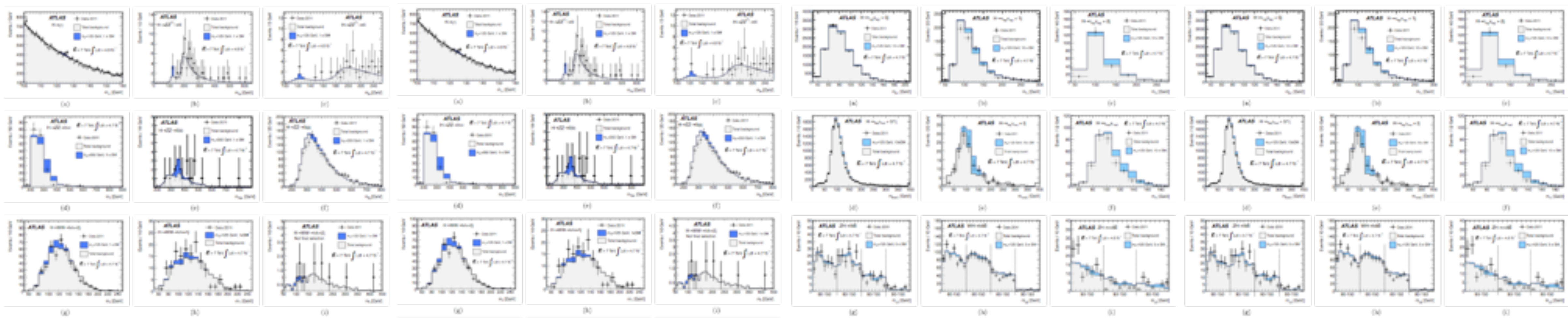


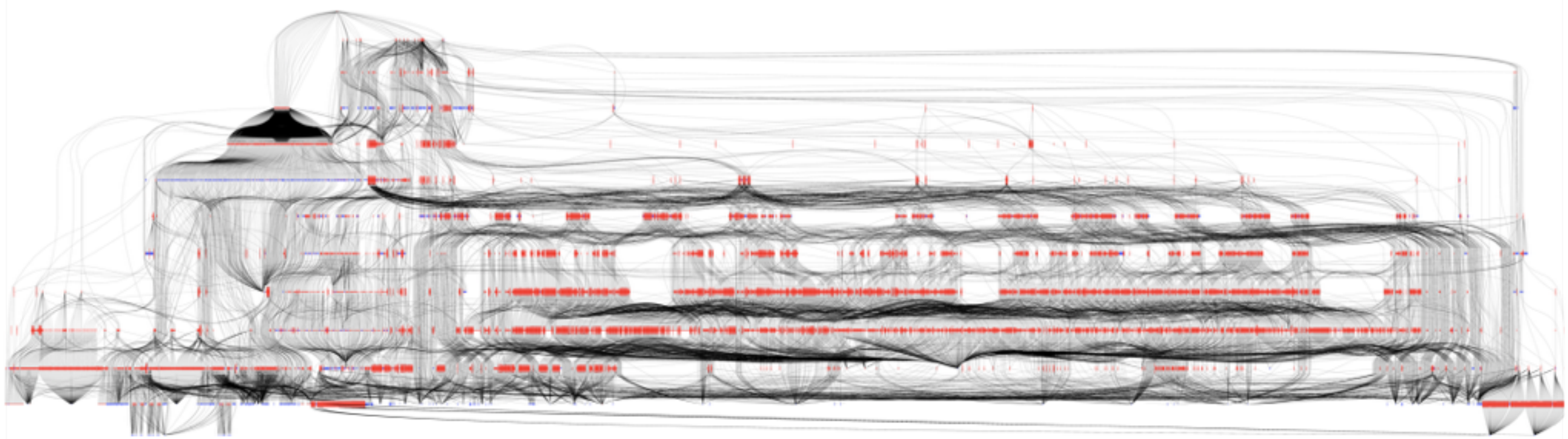
The generated charged particle emits the Cherenkov light.



Slide from Kyle Cranmer:

LIKELIHOOD-BASED COMBINATIONS





$$\mathbf{f}_{\text{tot}}(\mathcal{D}_{\text{sim}}, \mathcal{G}|\boldsymbol{\alpha}) = \prod_{c\in\text{channels}} \left[\text{Pois}(n_c|\nu_c(\boldsymbol{\alpha})) \prod_{e=1}^{n_c} f_c(x_{ce}|\boldsymbol{\alpha}) \right] \cdot \prod_{p\in\mathbb{S}} f_p(a_p|\alpha_p)$$

Data Analysis

- Objectives:
  - ***Searches*** (hypothesis testing): Likelihood Ratio Test (Neyman-Pearson lemma)
  - ***Measurements***: Maximum Likelihood Estimate
  $$\frac{P(x|H_1)}{P(x|H_0)} > k_\alpha$$
  - ***Limits*** (confidence intervals): Also based on Likelihood
- ***Likelihood***

$$p(\{x\}|\theta) = \mathrm{Pois}(n|\nu(\theta)) \prod_{e=1}^{n} p(x_e|\theta)$$

  - *n* Independent Events (*e*) with Identically Distributed Observables ({*x*})
  - Significant part of Data Analysis is ***approximating the likelihood*** as best as we can.

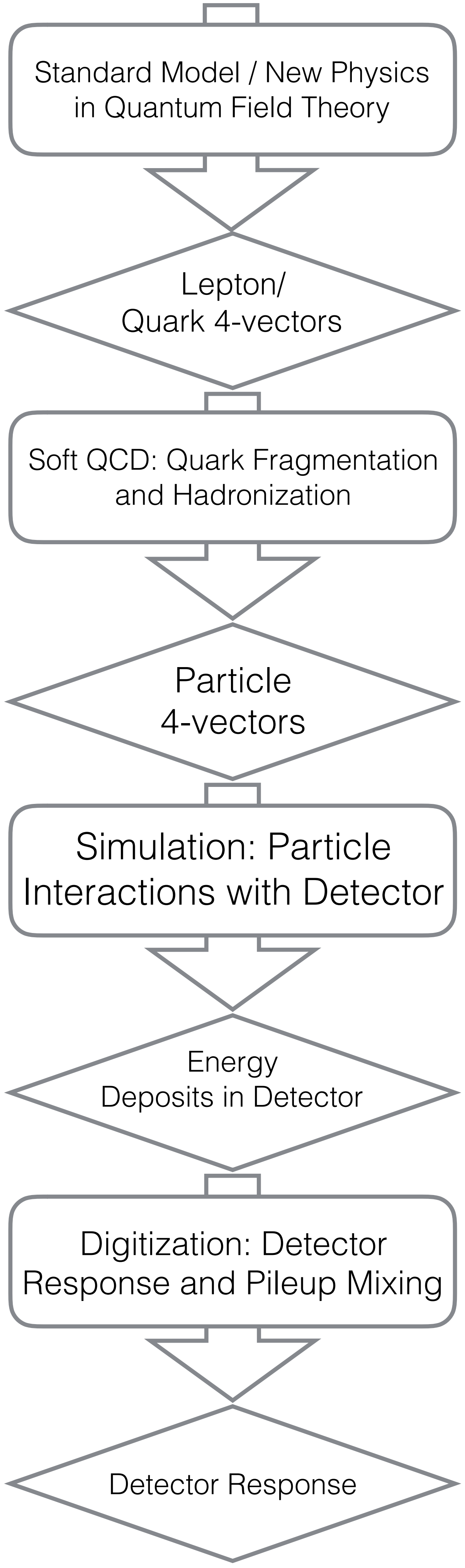
Approximating the Likelihood

- Physics is all about establishing a very precise “model” of the underlying phenomena… so ***we can model our data very well.***

- Enables ***multi-step ab-initio simulations***:

  1. ***Generation***: Standard Model and New Physics are expressed in language of Quantum Field Theory.
     - ➡ Feynman Diagrams simplify perturbative prediction of HEP interactions among the most fundamental particles (leptons, quarks)
  2. ***Hadronization***: Quarks turn to jets of particles via Quantum Chromodynamics (QCD) at energies where theory is too strong to compute perturbatively.
     - ➡ Use semi-empirical models tuned to Data.
  3. ***Simulation***: Particles interact with the Detector via stochastic processes
     - ➡ Use detailed Monte Carlo integration over the “micro-physics”
  4. ***Digitization***: Ultimately the energy deposits lead to electronic signals in the O(100 Million) channels of the detector.
     - ➡ Model using test beam data and calibrations.

- Output is fed through ***same reconstruction as real data***.



Likelihood Approximations

- Need $P(\{x_e\}|\theta)$ of an observed event (e). The better we do, the more sensitive our measurements.
- Steps 2 (Hadronization) and 3 (Simulation) can only be done in the ***forward mode***…

  ➡ ***cannot evaluate the likelihood.***

- So we simulate a lot of events and use a ***Probability Density Estimator (PDE)***, e.g. a histogram.
  - $\{x_e\}$ = {100M Detector Channels} or even { particle 4-vectors } are too high dimension.
  - Instead we derive $\{x_e\}$ = { small set of physics motivated observables } → ***Lose information.***
    - ***Isolate signal*** dominating regions of $\{x_e\}$ → ***Lose Efficiency.***
    - Sometimes use ***classifiers*** to further reduce dimensionality and improve significance
    - ***Profile the likelihood*** in 1 or 2 (ideally uncorrelated) observables.
- Alternative, try to brute force calculate via ***Matrix Element Method***:

$$\mathcal{P}(\boldsymbol{p}^{vis}|\alpha) = \frac{1}{\sigma_\alpha}\int d\Phi dx_1 dx_2 |M_\alpha(\boldsymbol{p})|^2 W(\boldsymbol{p},\boldsymbol{p}^{vis})$$

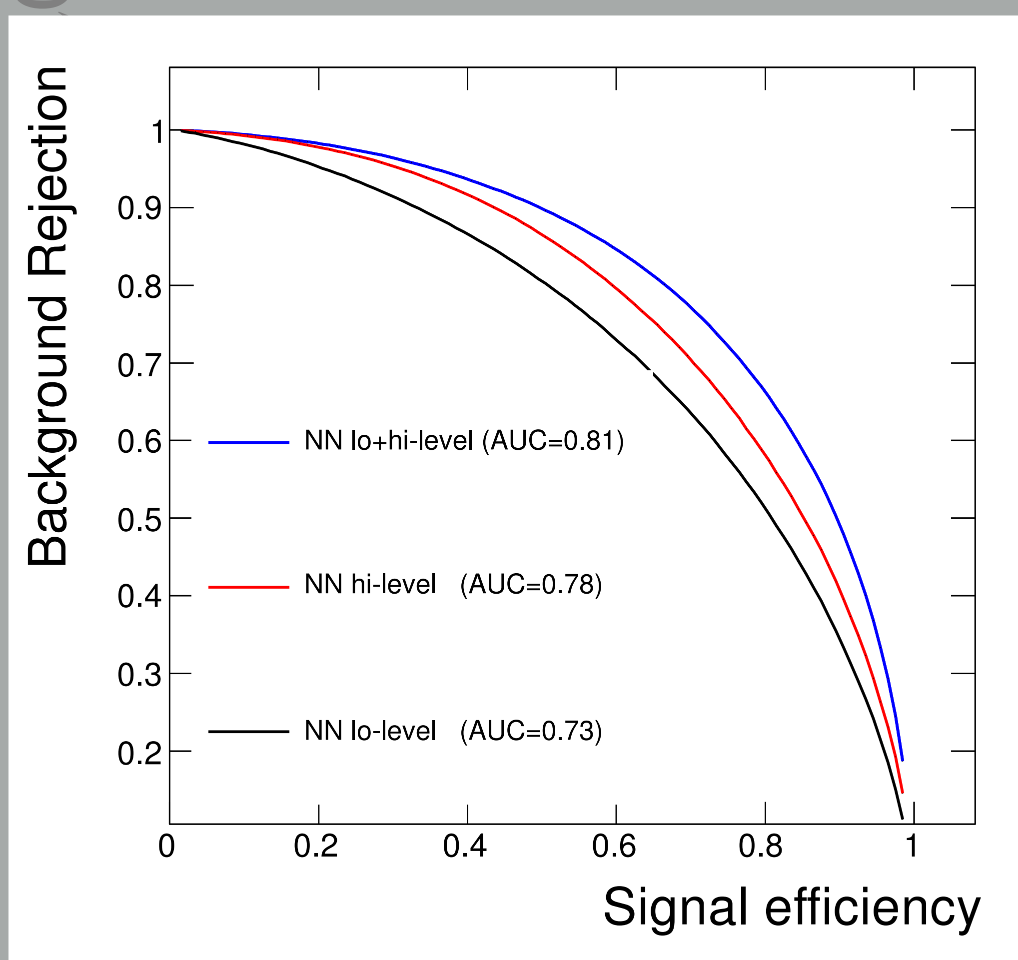
- But it's technically difficult, computationally expensive, mistreats hadronization, and avoids simulation by highly simplifying the detector response.

Searching for Exotic Particles in High-Energy Physics with Deep Learning

P. Baldi,[1] P. Sadowski,[1] and D. Whiteson[2]

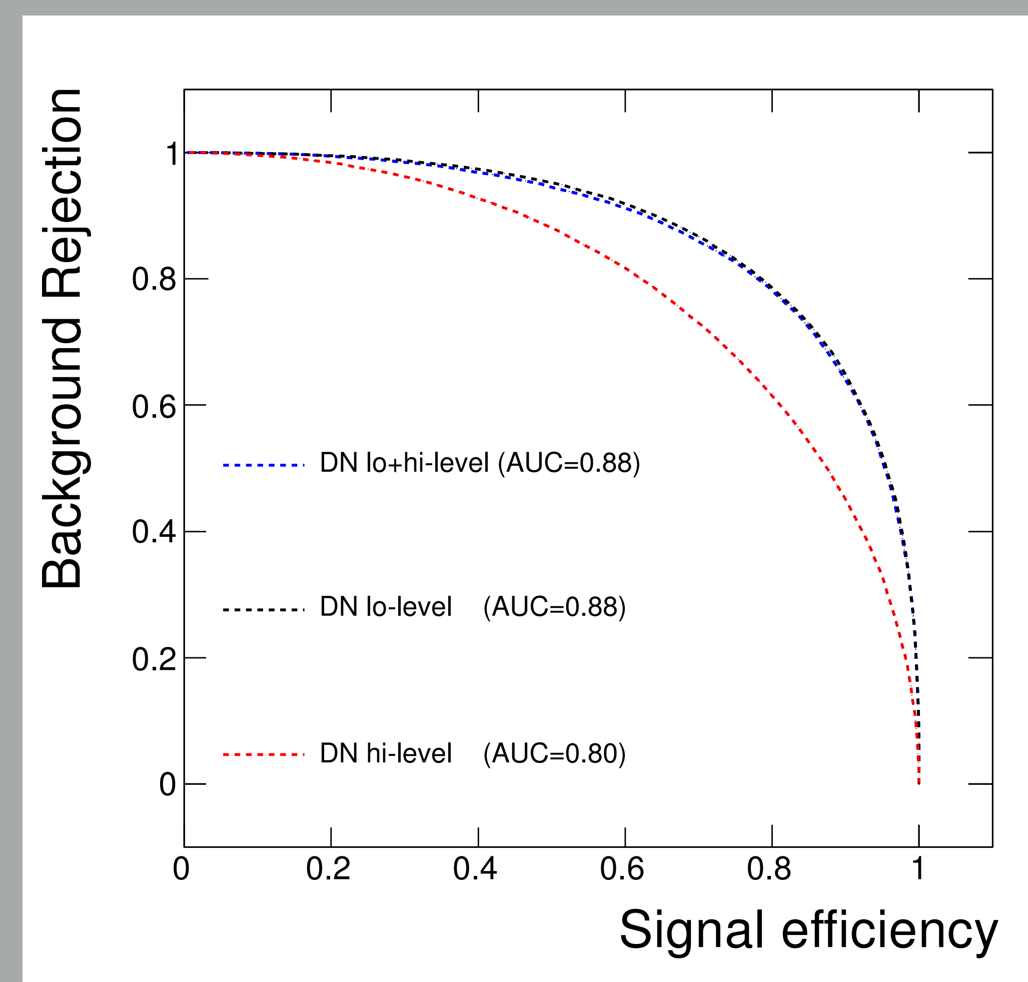
[1] *Dept. of Computer Science, UC Irvine, Irvine, CA 92617*[*]
[2] *Dept. of Physics and Astronomy, UC Irvine, Irvine, CA 92617*[†]

Collisions at high-energy particle colliders are a traditionally fruitful source of exotic particle discoveries. Finding these rare particles requires solving difficult signal-versus-background classification problems, hence machine learning approaches are often used. Standard approaches have relied on 'shallow' machine learning models that have a limited capacity to learn complex non-linear functions of the inputs, and rely on a pain-staking search through manually constructed non-linear features. Progress on this problem has slowed, as a variety of techniques have shown equivalent performance. Recent advances in the field of deep learning make it possible to learn more complex functions and better discriminate between signal and background classes. Using benchmark datasets, we show that deep learning methods need no manually constructed inputs and yet improve the classification metric by as much as 8% over the best current approaches. This demonstrates that deep learning approaches can improve the power of collider searches for exotic particles.



(a)

the study
nvestigat-
govern its
damental
ary tools
odern ac-
rotons to
nely high
l measur-



(b)

er,
3],
d.
t-
c-
in
ne
ne
p-
to
at

Boosted Object Tagging

- Decay products of ***Highly Energetic heavy particles*** (i.e. top, W, Z, H) are collimated into ***large jets that cannot be resolved.***

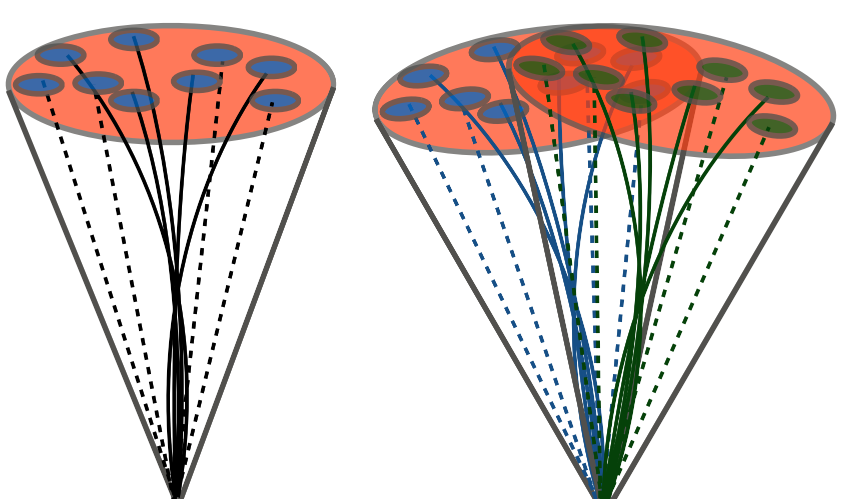
- Boosted object tagging using Deep Learning is being pursed by multiple groups in the LHC experiments… mostly the data is private, so I can't show.

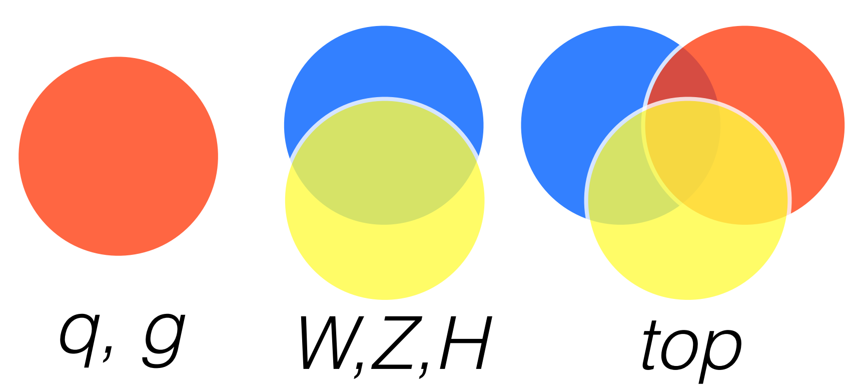
- Early study by SLAC group (M. Kagan, A. Schwartzman, et al)

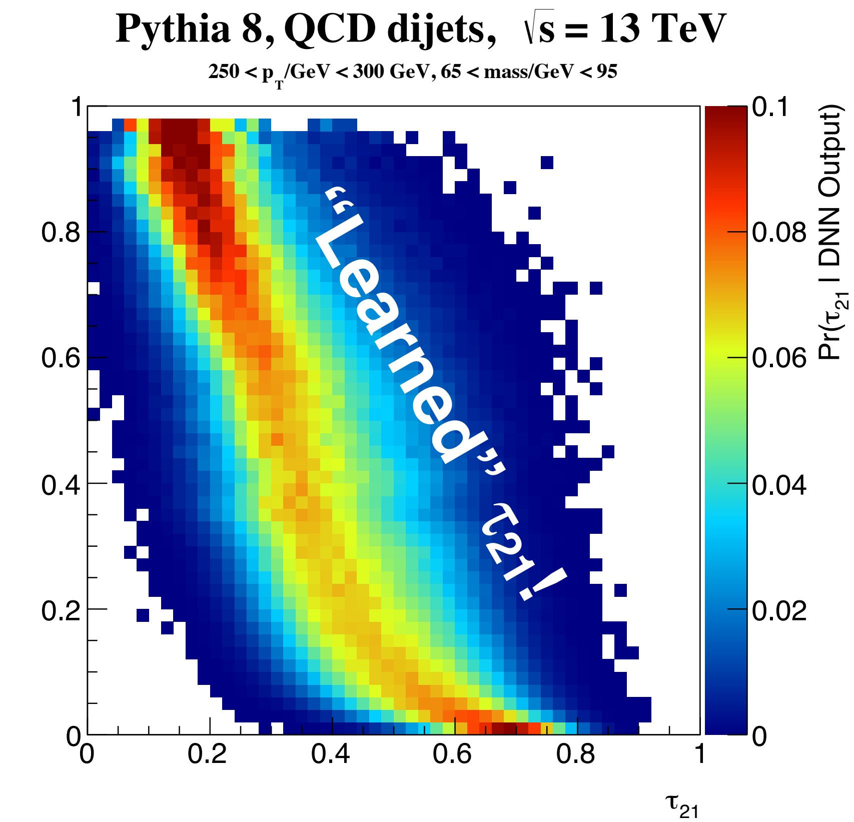
    - ***Jet Images***: Construct images of the energy deposits of jets (paper)

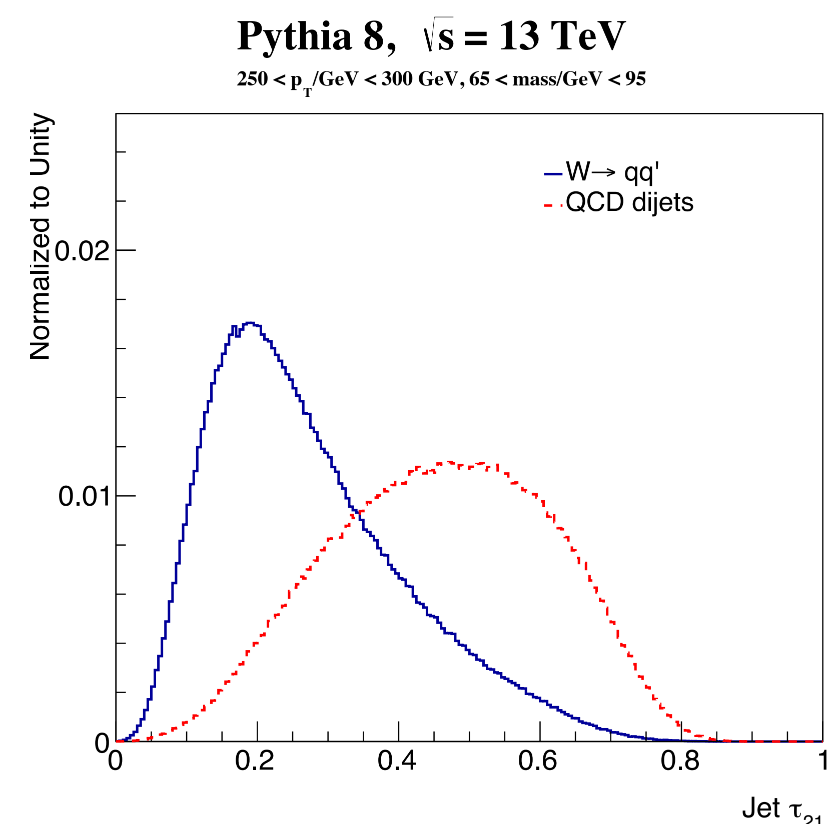
    - Use classifiers, such as CNNs…

    - By studying the features of the CNN, SLAC group got new insight into color-flow which can be used in traditional techniques.



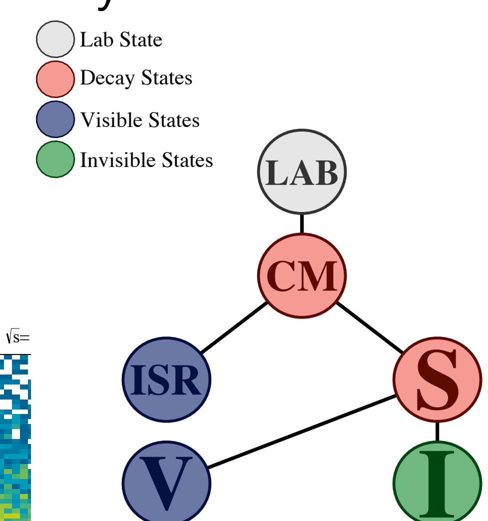






Simple Example

- ***Chris Rogan*** (while grad student on CMS experiment) invented a set of observables (known as ***Razor***) for separating SUSY events from their backgrounds. (https://arxiv.org/abs/1006.2727)
  - My group brought the technique to the ATLAS experiment… 3 searches/papers in LHC Run 1.
  - Chris moved to ATLAS when he moved to Harvard… but we never had chance to directly work together (though my postdoc/student do).
  - We just finished (for ICHEP 2016) a search based on "***Recursive Jigsaw***", a new set of observables Chris developed with Paul Jackson.
    - An ***uncorrelated basis of observables*** based on successive boosts into decay frames of the particles.



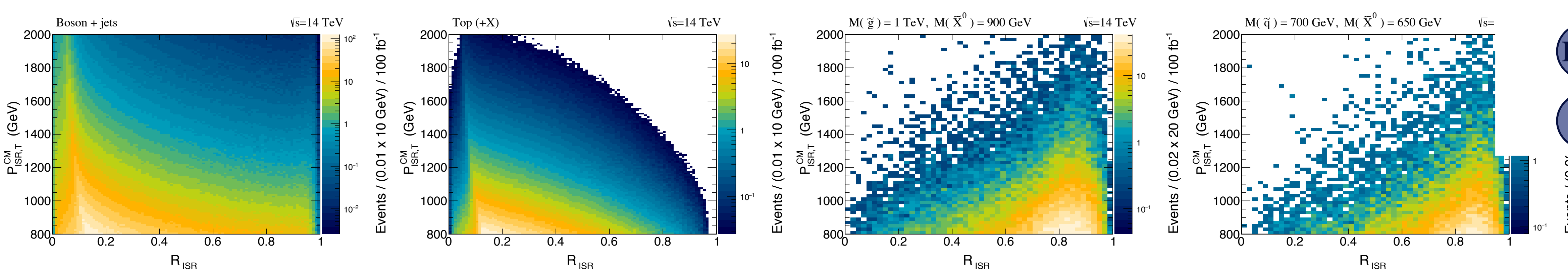
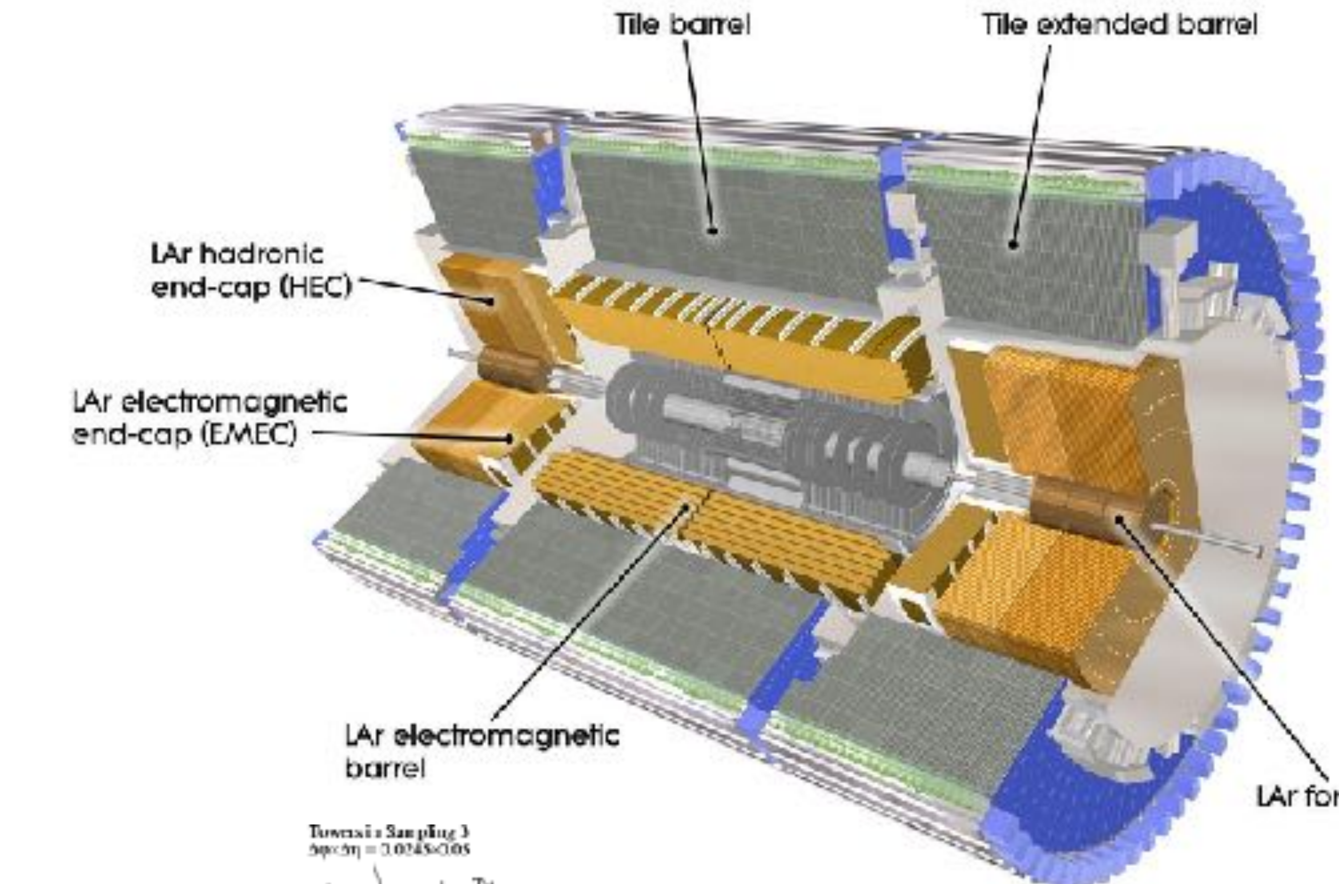


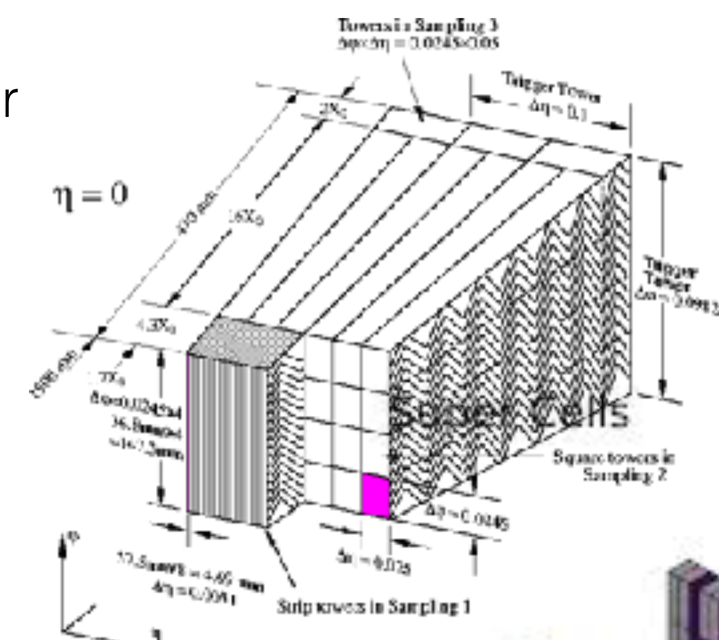
FIG. 4. Distribution of the $p^{\mathrm{CM}}_{\mathrm{ISR},T}$ as a function of R_{ISR} for (from left to right) boson+jets and top+X backgrounds, gluino and squark pair-production signal samples.

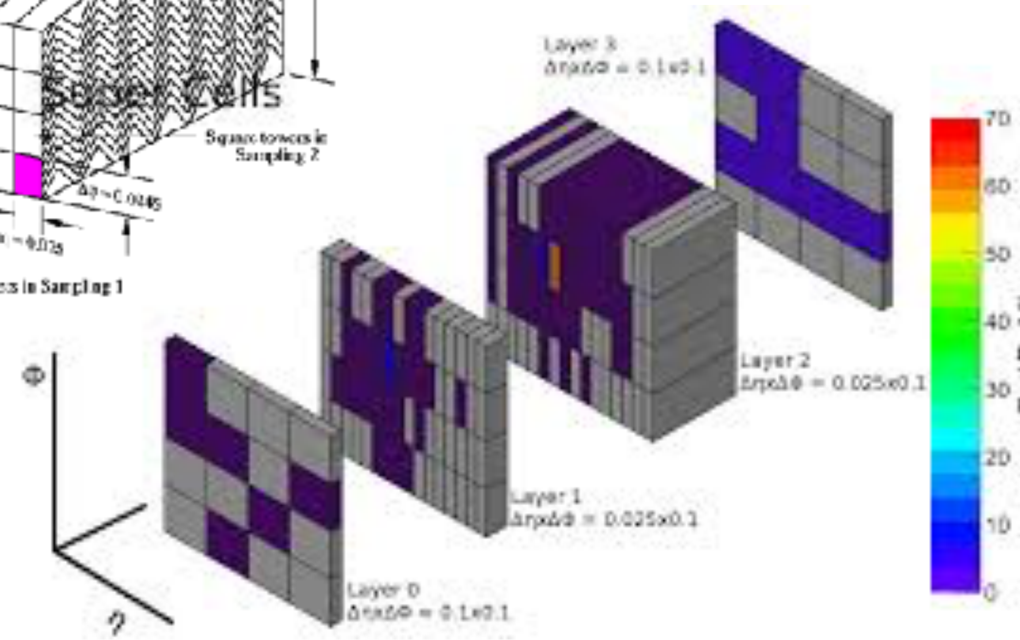
http://arxiv.org/pdf/1607.08307.pdf

ATLAS Calorimeter

- ***Ideally suited*** for "***imaging***" ~ 64 x 36 x 7 3D Image
  - 200K Calorimeter cells measure energy deposits.
  - Interesting Challenges: non-uniform granularity, cylindrical geometry.
- ***High impact***:
  - ***Improve Identification and energy resolution*** make the peaks stand out.
  - Turn DNN into generative model for ***fast shower simulation***.
- ***High potential***: we don't use all information so room for improvement
  - *e/gamma:* take full advantage of the high granularity and accordion structure
  - *hadronic calibration:* take full advantage of longitudinal sampling and other handles
  - *particle flow:* correlate with tracks (and vertex) for hadronic calibration, taus, jet-tagging, boosted objects…
- ***Problem: Private Data…***







DNN+HEP Software Needs (1/4)

1. *Inference in HEP Frameworks:*
   - Need optimized and validated inference implementation.
     - Nova uses Caffe in art.
     - LArSoft/ATLAS using handwritten C++.
     - TMVA has similar new DNN implementation (w/ GPU support)
   - DNN weights can be Gigabytes, likely need
     - Condition DB-like systems storage.
     - Memory sharing between processes/threads.
   - I can imagine a DL service similar to ATLAS APE GPU service:
     - Processes are client of server(s) that talk to backends/accelerators.
     - No reason for every experiment to reinvent the wheel here…

DNN+HEP Software Needs (2/4)

2. *Training systems*:

- Training DNNs efficiently generally requires GPUs (or other future accelerators).
- Hyper-parameter scans critical part of DNN development workflow.
  - Great use of GPUs on HPCs.
  - Google and other clouds specifically target DL.
- Today's training samples can already be 10s of Terabytes, requiring massive parallelism.
  - *Data Parallelism*: Bottlenecked by gradient syncing between GPUs or systems. Lots of Engineering in Industry already.
  - *Model Parallelism*: Less sync'ing but only makes sense for large enough model.
  - No more embarrassingly parallel. Must provision large number of machines.
- As DNNs become essential, training them becomes part of software releases, simulation, reco,… cycle.
  - New simulation/reco can require regenerating large training sets (various conditions) and running long training before using reco.
  - Somewhat analogous to calibration on express streams.
- I can imagine Workflow and Data Management systems designed for DL training workflows on any available resource.

DNN+HEP Software Needs (3/4)

3. Opportunistic Data Generation/Processing:

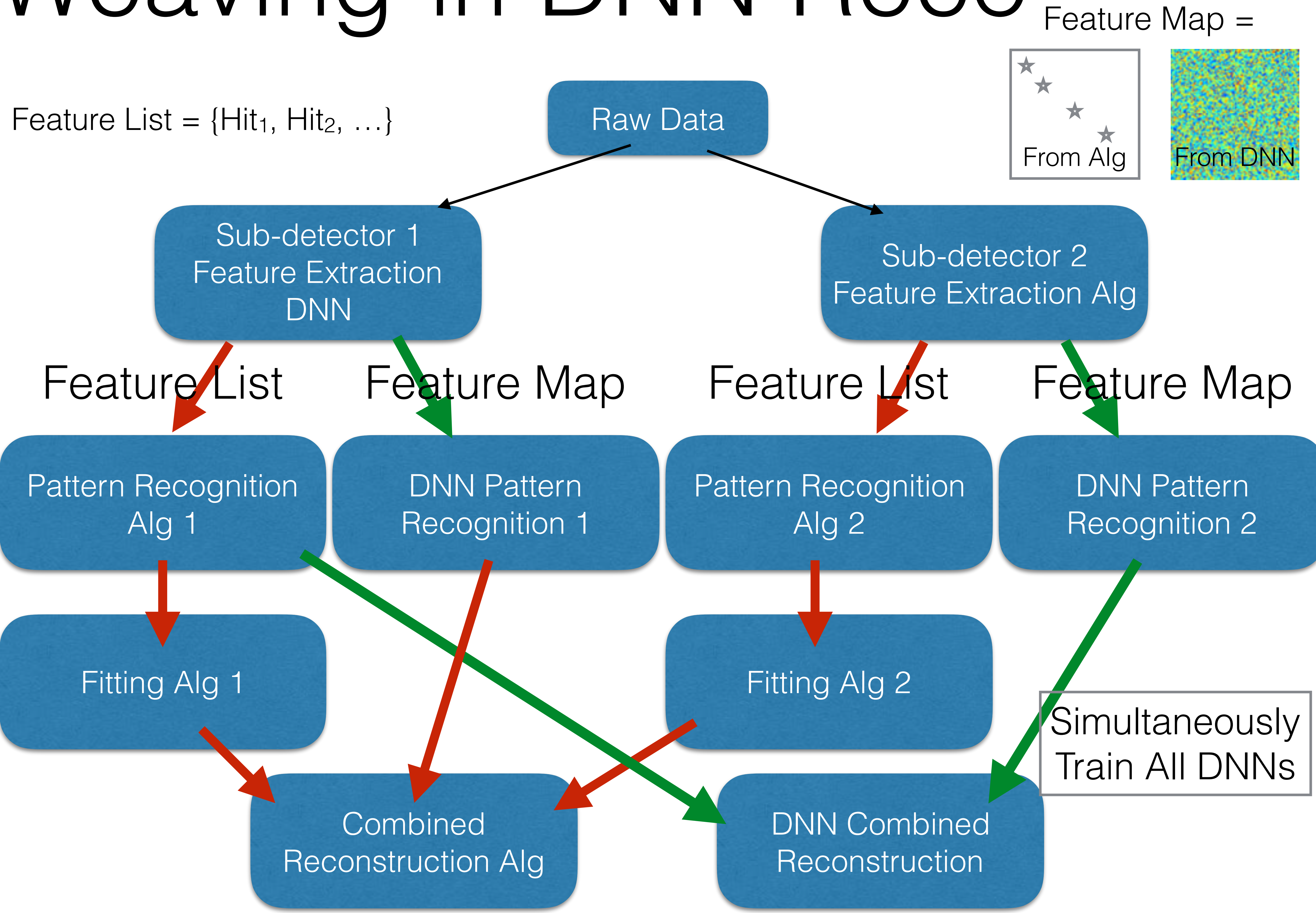
- DL generally requires huge independent training samples.
  - Probably need to resort to Data Augmentation, Fast MC, etc… when possible.
  - But the data is private, making collaboration and rapid publication difficult.
- Collaboration with Machine Learning experts and among experiments require public data sets.
  - Publicly available simulation and reconstruction (for base-line). (see: Journal of Brief Ideas.)
  - Reconstruction DNNs will likely require Geant4. (i.e. CPU intensive)
  - No dedicated resources, so rely on opportunistic CPU.
  - Need to store and distribute large data-sets.
- I can image WMS/DDM systems allowing users to opportunistically run docker containers on any system, and centrally collecting samples for everyone.

DNN+HEP Software Needs (4/4)

4. *Event Processing within Deep Learning Frameworks*

- DL will potentially become integral to our software and trigger
  - We may replace code with weights.
  - DL integrated into HEP frameworks. Not just an external. (example next slide)
- Many-core/FPGA/neuro-morphic accelerators may prolong Moore's law
  - Experiments like DUNE will run for 30 years and must keep up with emerging tech.
  - Frameworks must [automatically] optimize and place computations on a variety of hardware.
  - May need to distribute processing of individual events across cluster (like HEP trigger)
    - Use network hardware for primitive operations during transfers.
    - Partially process on specialized machines (specific accelerators, HPC, massive memory, …)
- Threading in GaudiHive, CMS FW, art, … use data flow programming model (graphs), like many DL systems.
- Industry will highly optimize DL systems and provide services around them.

Weaving-in DNN Reco



R&D Proposal

- *Premise*: We need new frameworks to take advantage of DL and emerging architectures.
    - ➡ Build HEP Framework on top of a DL Framework.
- If we envision new frameworks need to do R&D now, ver 1.0 by 2020, deployed by 2025.
- R&D Proposal (can we do traditional HEP Reco in DL Framework?):
    - Build HEP Reco on top of Google's OpenSource TensorFlow
        - General computation system, based on Directed Acyclic Graphs.
            - Framework for Automatic optimizations (like Theano), though currently primitive.
        - Supports all architectures and distributes computation across GPUs and clusters.
        - Build a HEP Framework in python (like Keras) with C++ wrapped in TF ops.
    - 3 project ideas:
        - First steps of LArTPC reco: deconvolution, hit finding, …
        - Online Sparsification and compression of LArTPC data for protoDUNEs.
        - ATLAS GPU Trigger Demonstrator: Wrap the existing GPU/CPU kernels in TensorFlow Ops