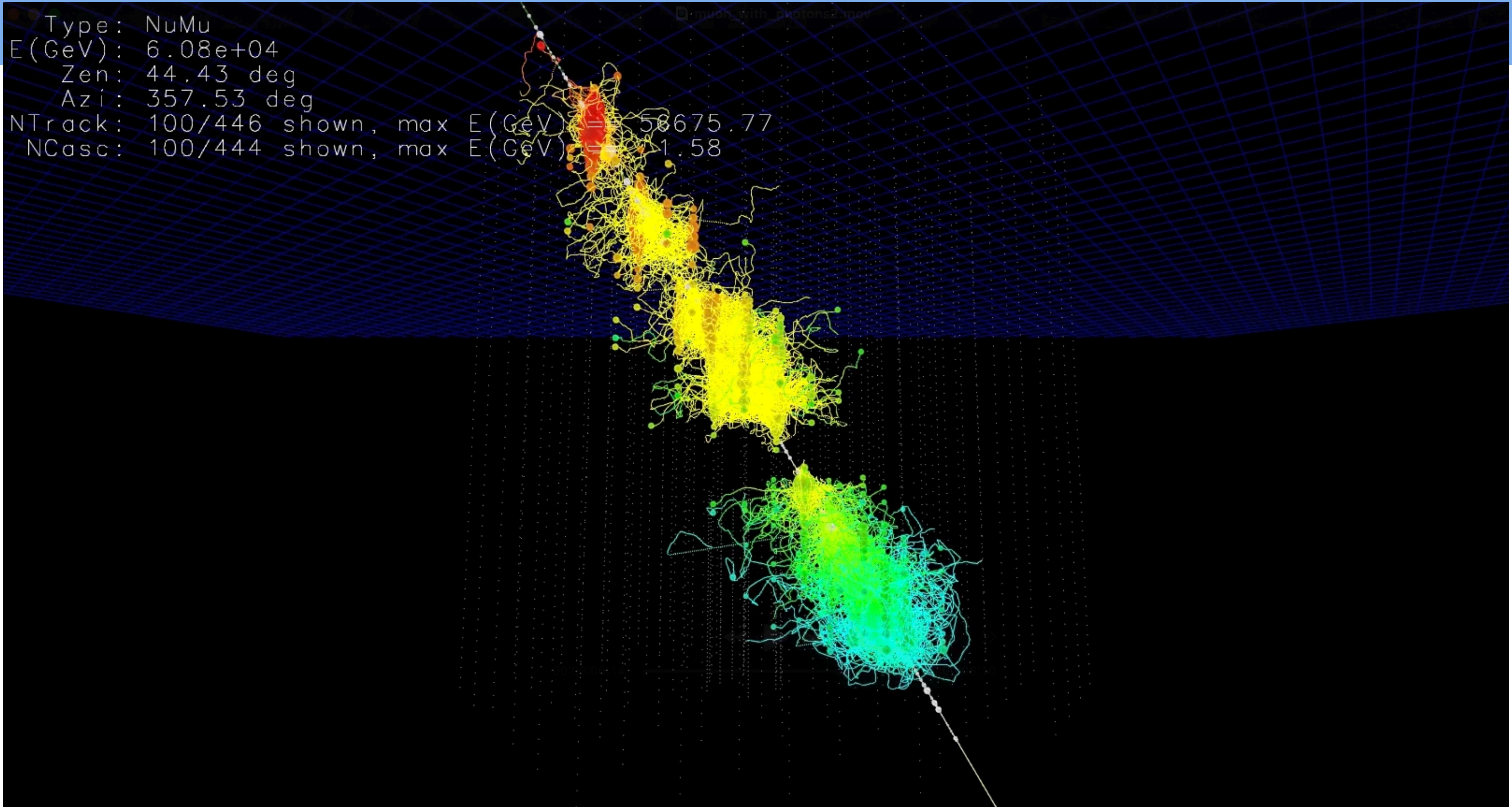
Towards improving efficiency of machine learning techniques in neutrino telescopes

Workshop on Machine Learning for Analysis of High-Energy Cosmic Particles

Felix Yu (Harvard University) | January 28, 2025


Type: NuMu
E(GeV): 6.08e+04
Zen: 44.43 deg
Azi: 357.53 deg
NTrack: 100/446 shown, max E(GeV) = 58675.77
NCasc: 100/444 shown, max E(GeV) = 1.58

Motivation

- Two major challenges:
  - **Spatial**: Neutrino telescope data is extremely sparse and of large-scale
  - **Temporal**: Events can span over thousands of nano-seconds, but fine timing resolution is important for many analyses

- Resolving these issues can lead towards development of ML reconstructions that are **efficient** and **flexible**, without sacrificing too much **performance**.

Outline

- Improve reconstruction speed
  - Efficient ML reconstruction of direction/energy using sparse submanifold CNNs (SSCNN)
- Representation learning
  - Learning effective and compact representations of neutrino telescope events
- Future plans for application in IceCube (WIP)
  - Fast ML at early levels in data pipeline

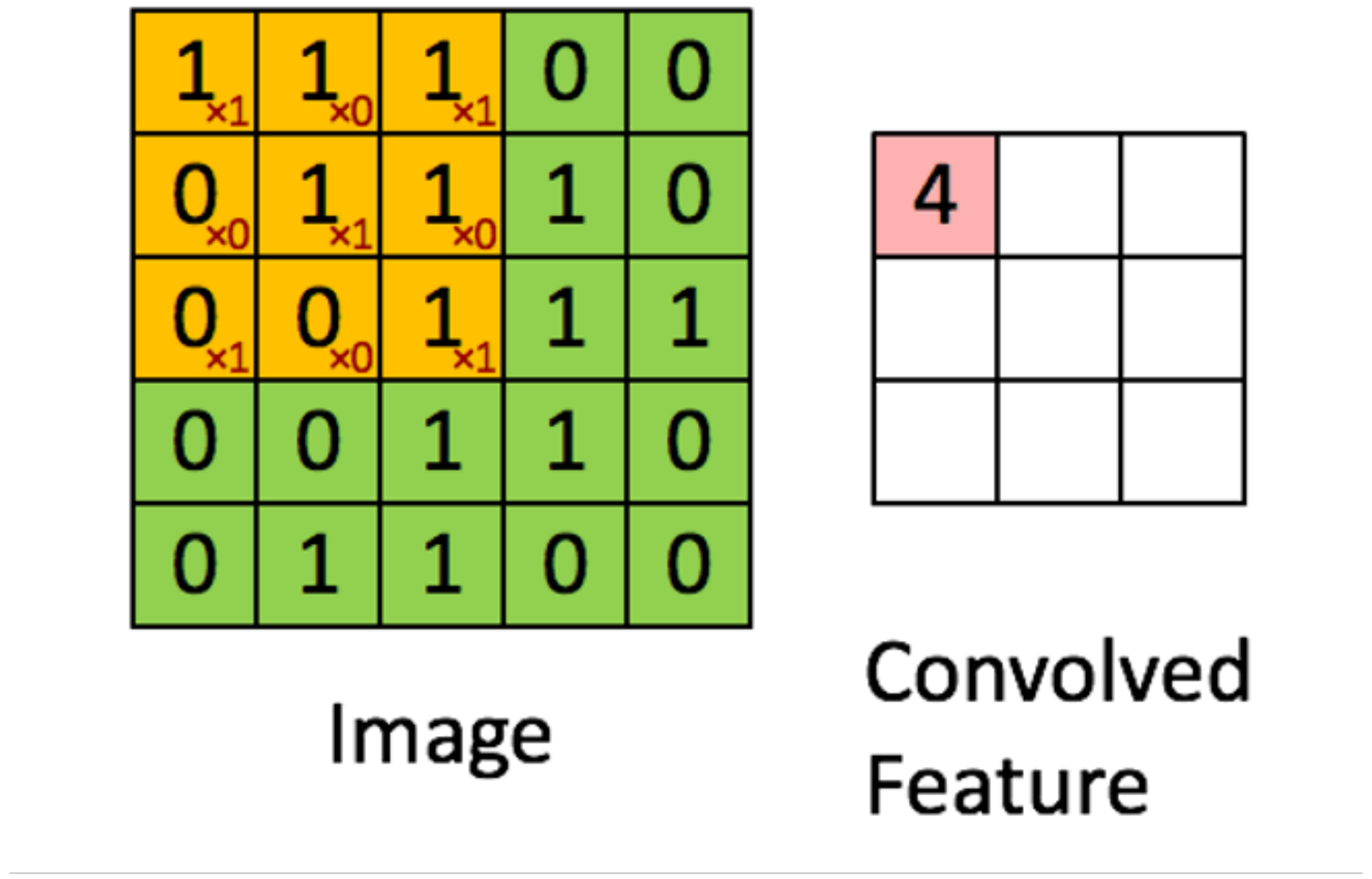
Two major challenges:

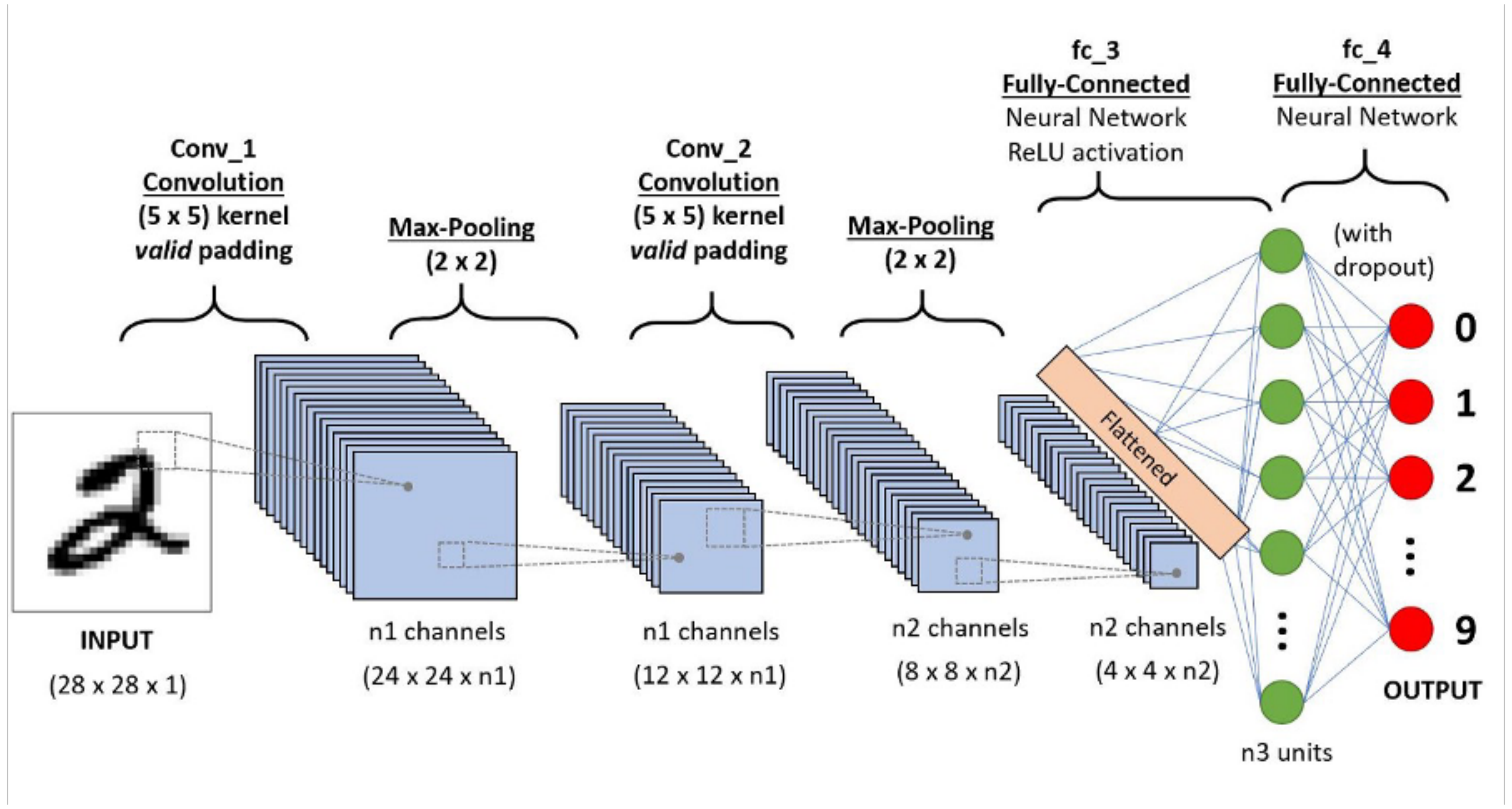
Spatial: Neutrino telescope data is extremely sparse and of large-scale

Temporal: Events can span over thousands of nano-seconds, but fine timing resolution is important for certain analyses

Convolutional Neural Networks

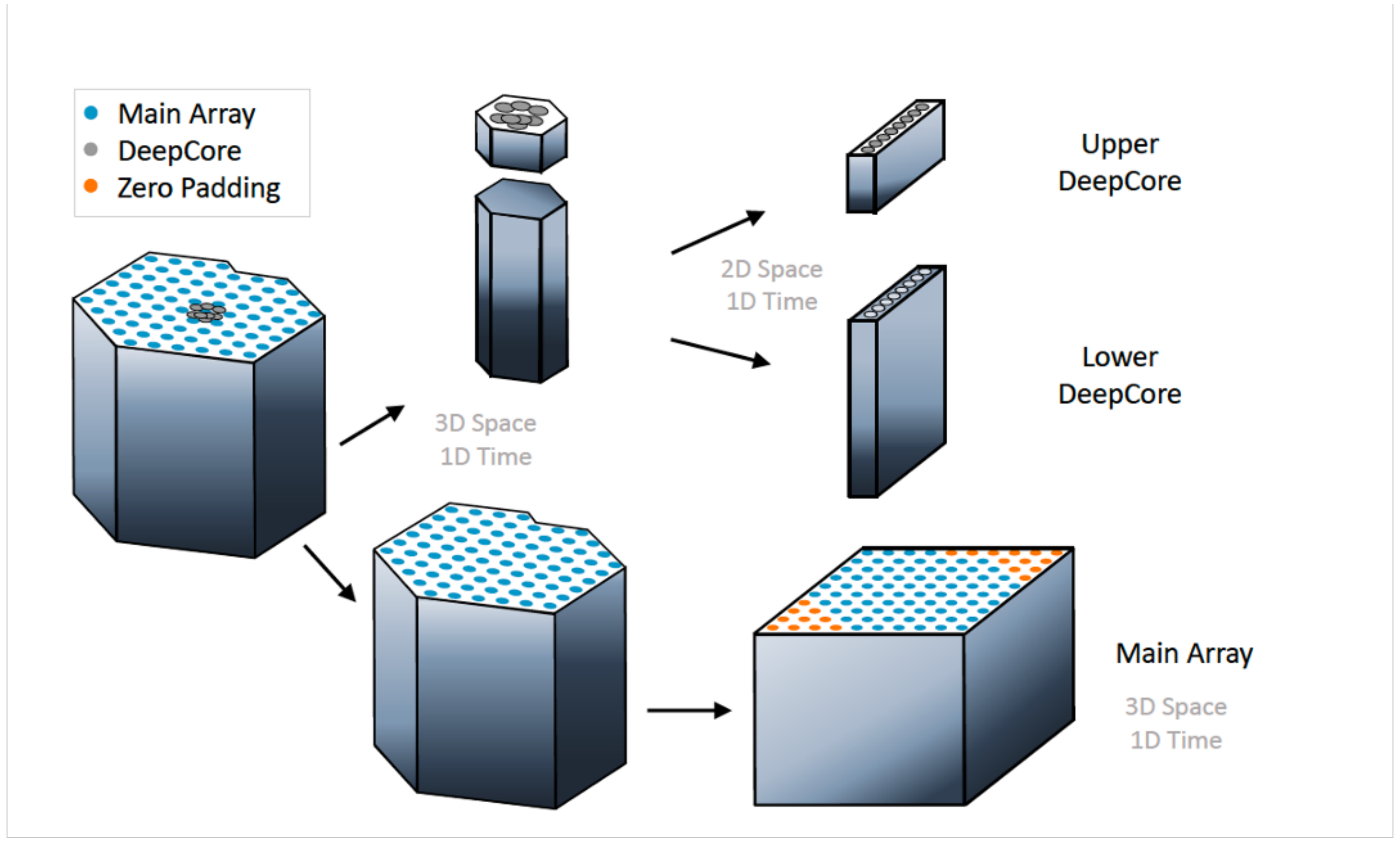
- Convolutional neural networks (CNNs) are a staple for image-like data
- Convolutions excel at feature extraction, which is done using kernels, and stacking layers to form a network

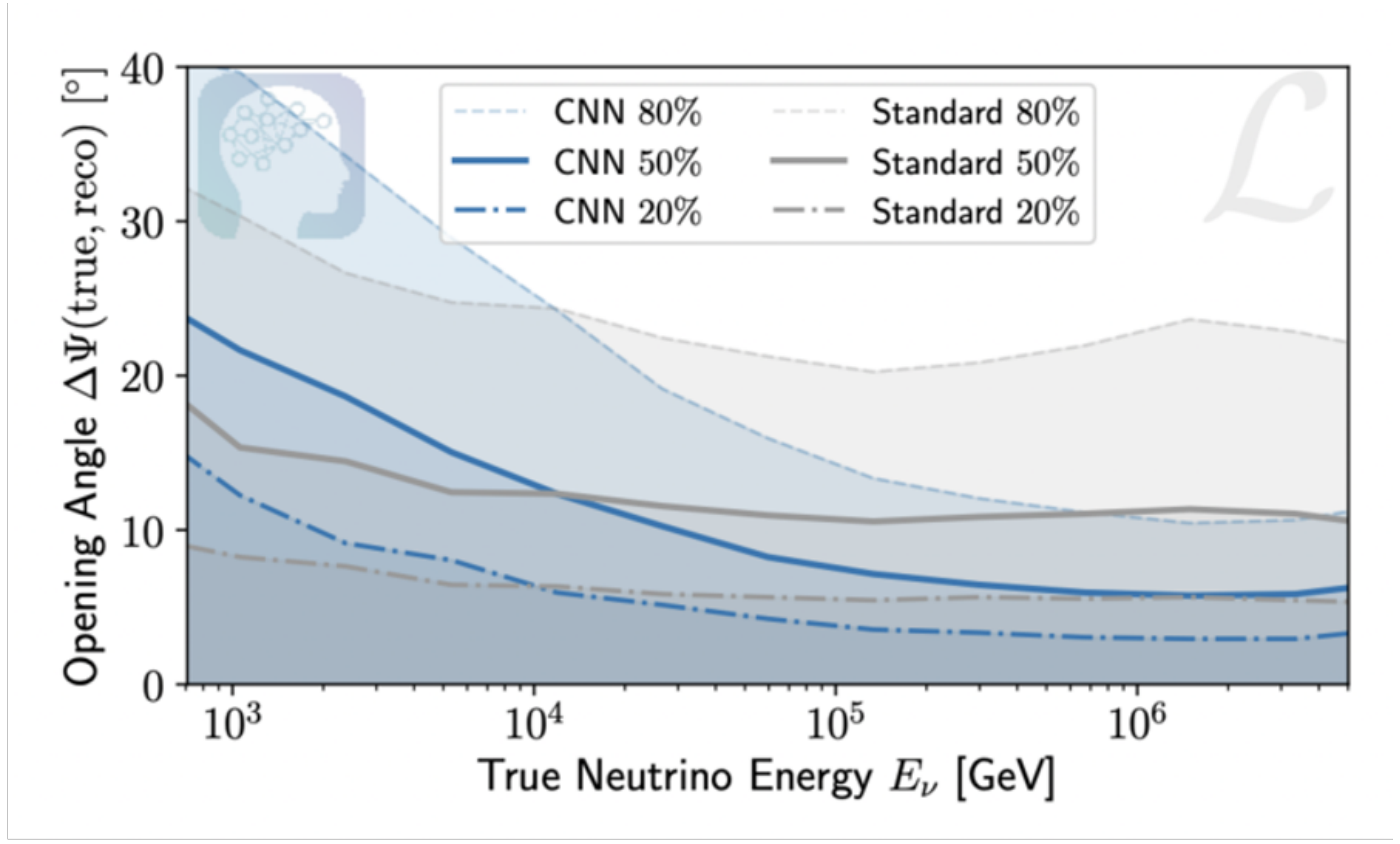




CNNs in IceCube

CNNs have seen action in neutrino telescopes like IceCube



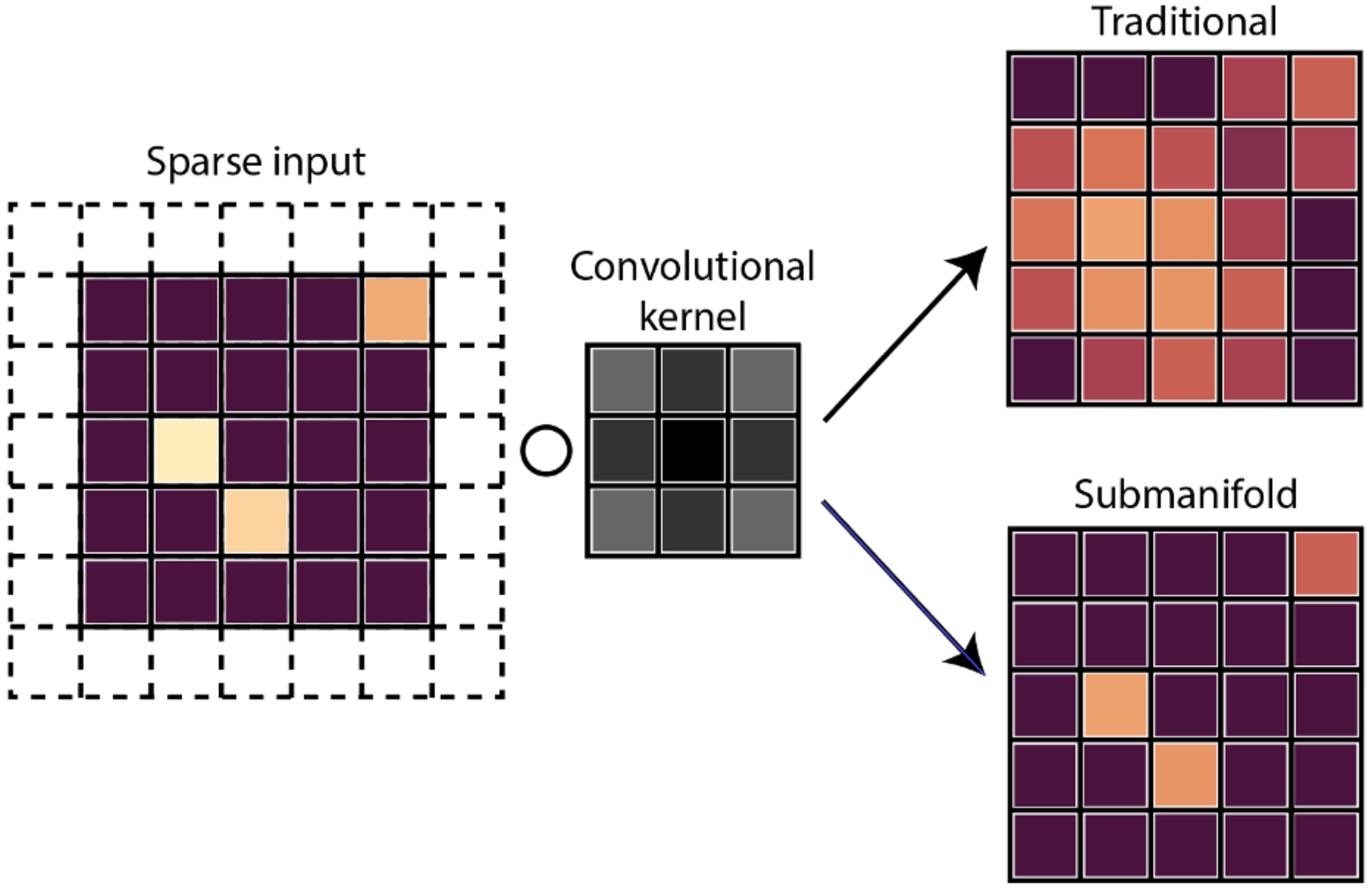


Each optical module serves as a "pixel"

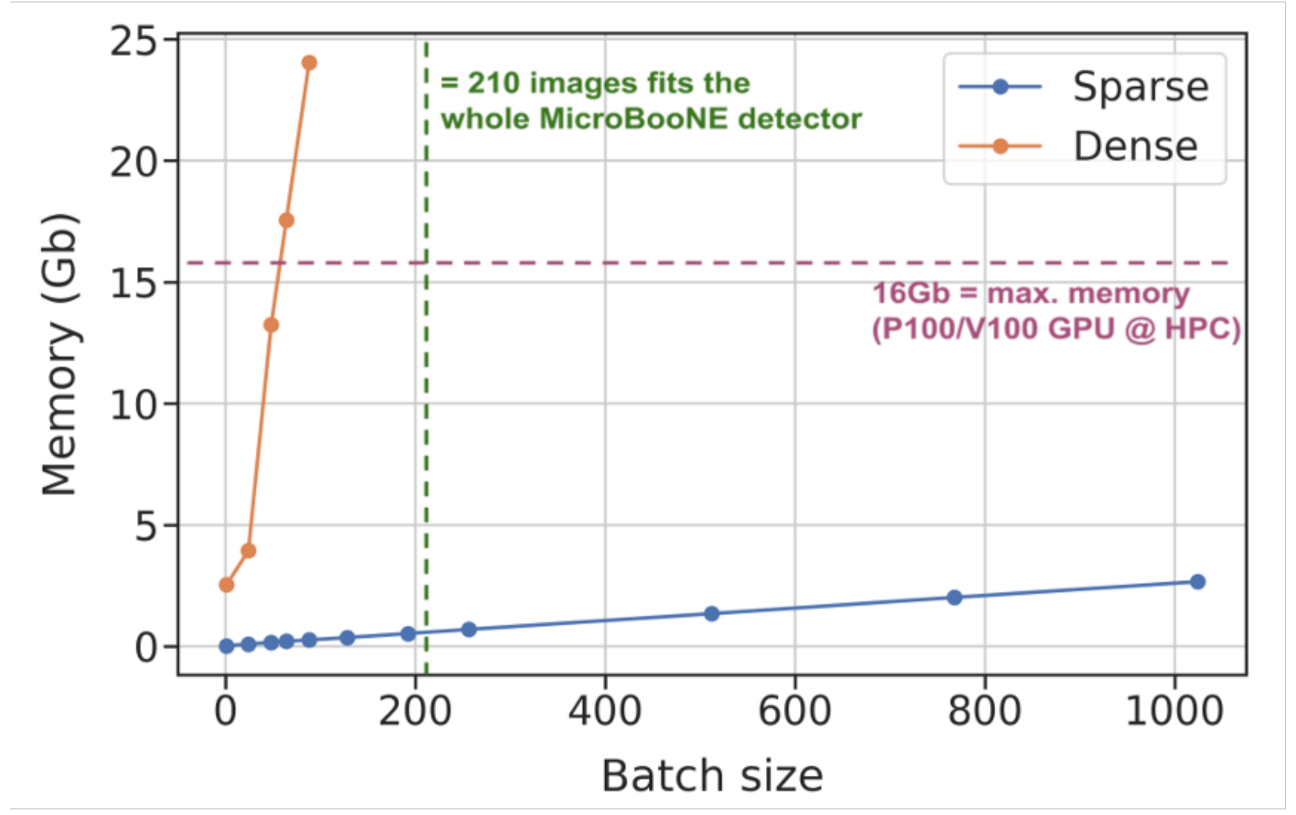
Sparsity problem!

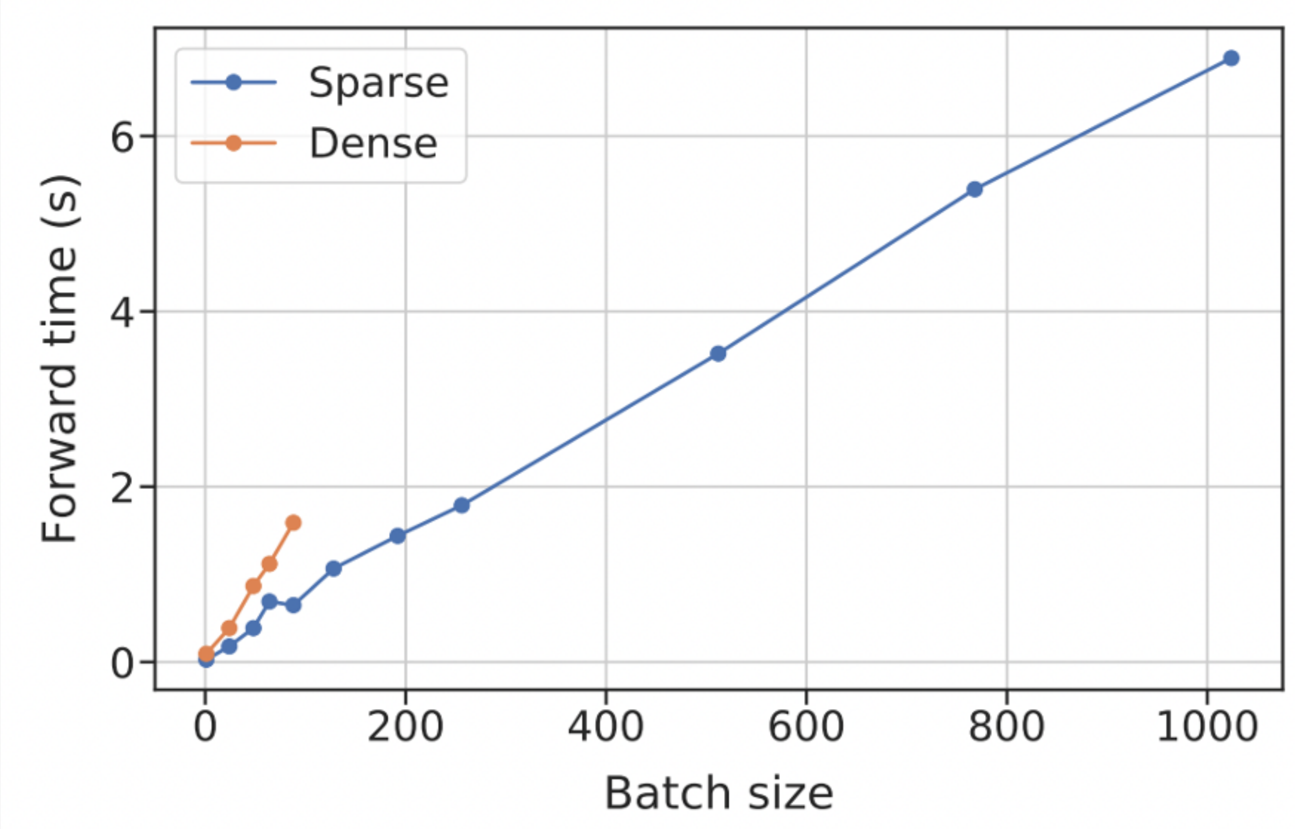
Sparse Submanifold Convolutions

- Sparse submanifold convolutions only operate on non-zero input coordinates
- Very efficient for sparse data
- Deployable for inference on CPUs!



F. J. Yu et al., [arXiv:2303.08812](https://arxiv.org/abs/2303.08812)

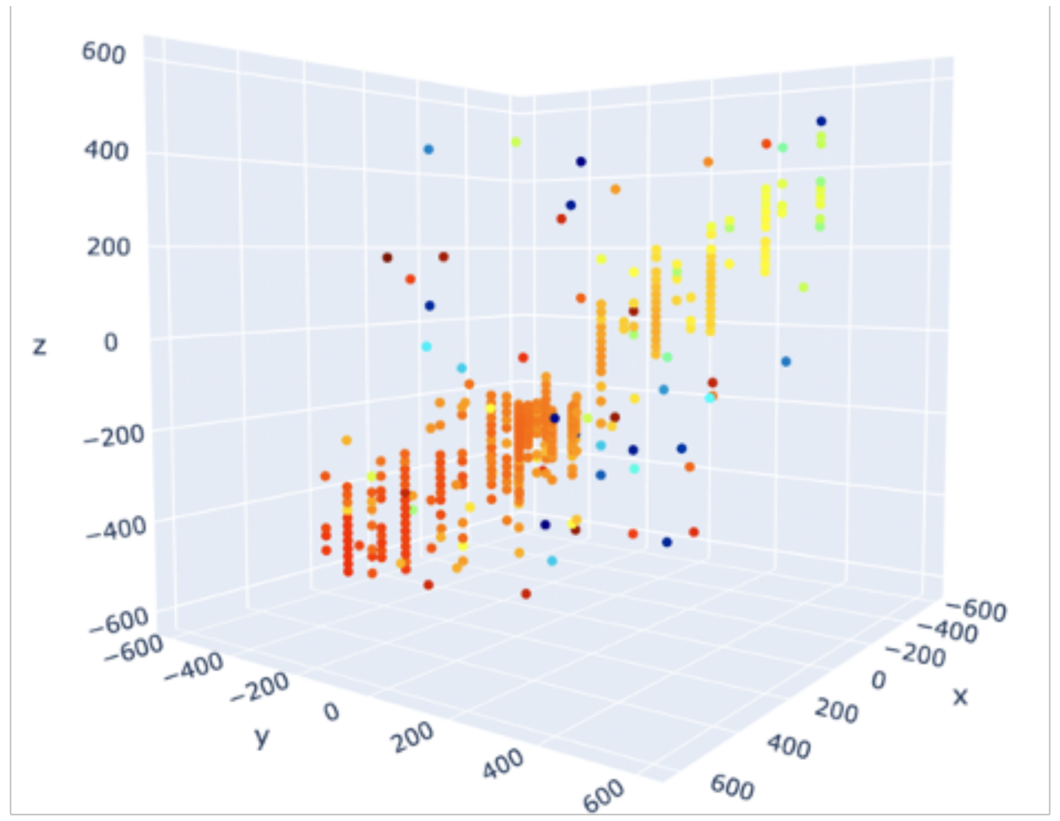




L. Domine et al., Phys. Rev. D **102**, 012005

Sparse Submanifold Convolutions

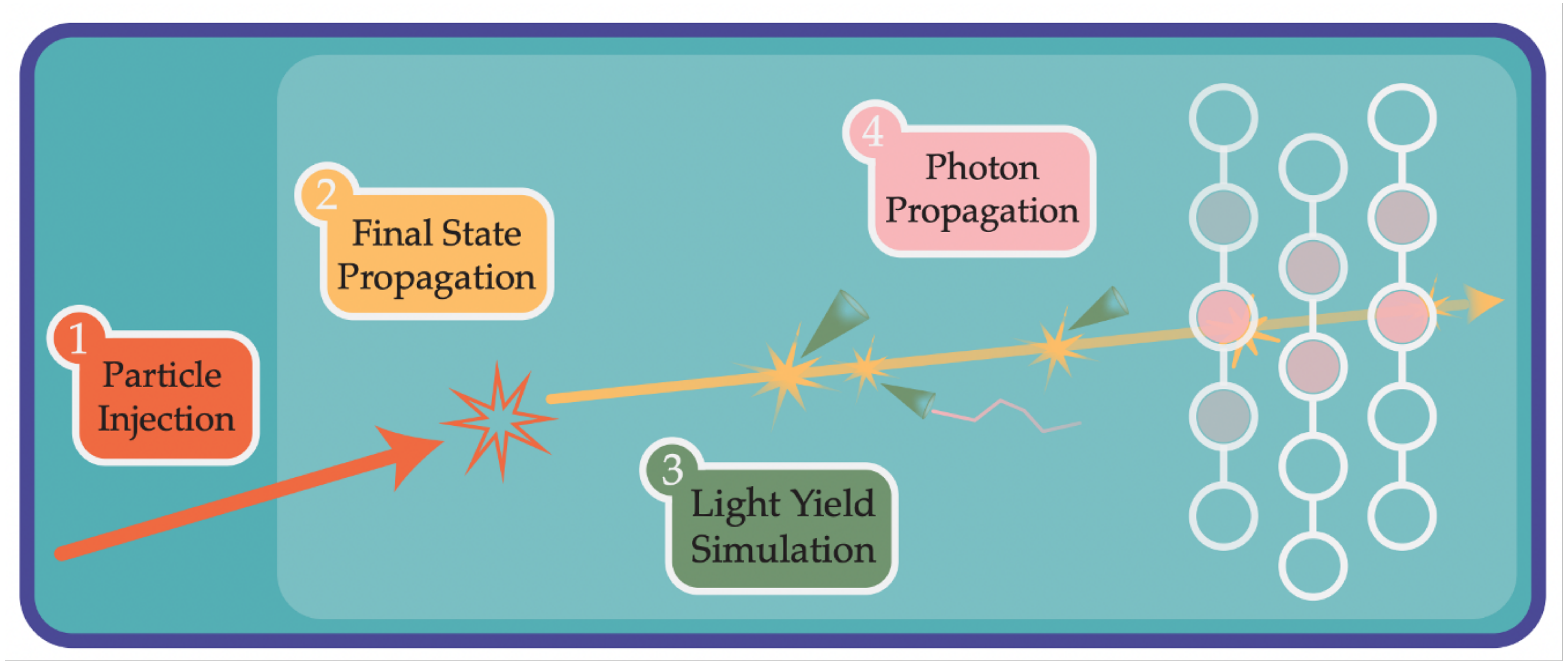
- Natural way to encode neutrino telescope data for SSCNNs is as a 4D point cloud
- Feature is the number of hits that occurred in the time bin
- **Thoughts for later**: Lots of hits/ pulses in the timing dimension! Can we think of ways to reduce this?



$$C = \begin{bmatrix} x_1 & y_1 & z_1 & t_1 \\ \vdots & \vdots & \vdots & \vdots \\ x_n & y_n & z_n & t_n \end{bmatrix}, F = \begin{bmatrix} h_1 \\ \vdots \\ h_n \end{bmatrix}$$

Proof-of-concept study with Prometheus

- We conducted a proof-of-concept study using open-source neutrino telescope simulation software Prometheus [1]
- It can simulate events for any detector configuration, we specifically used IceCube-like parameters



[1] J. Lazar et al. arXiv:2304.14526

Energy & Angular Reconstructions

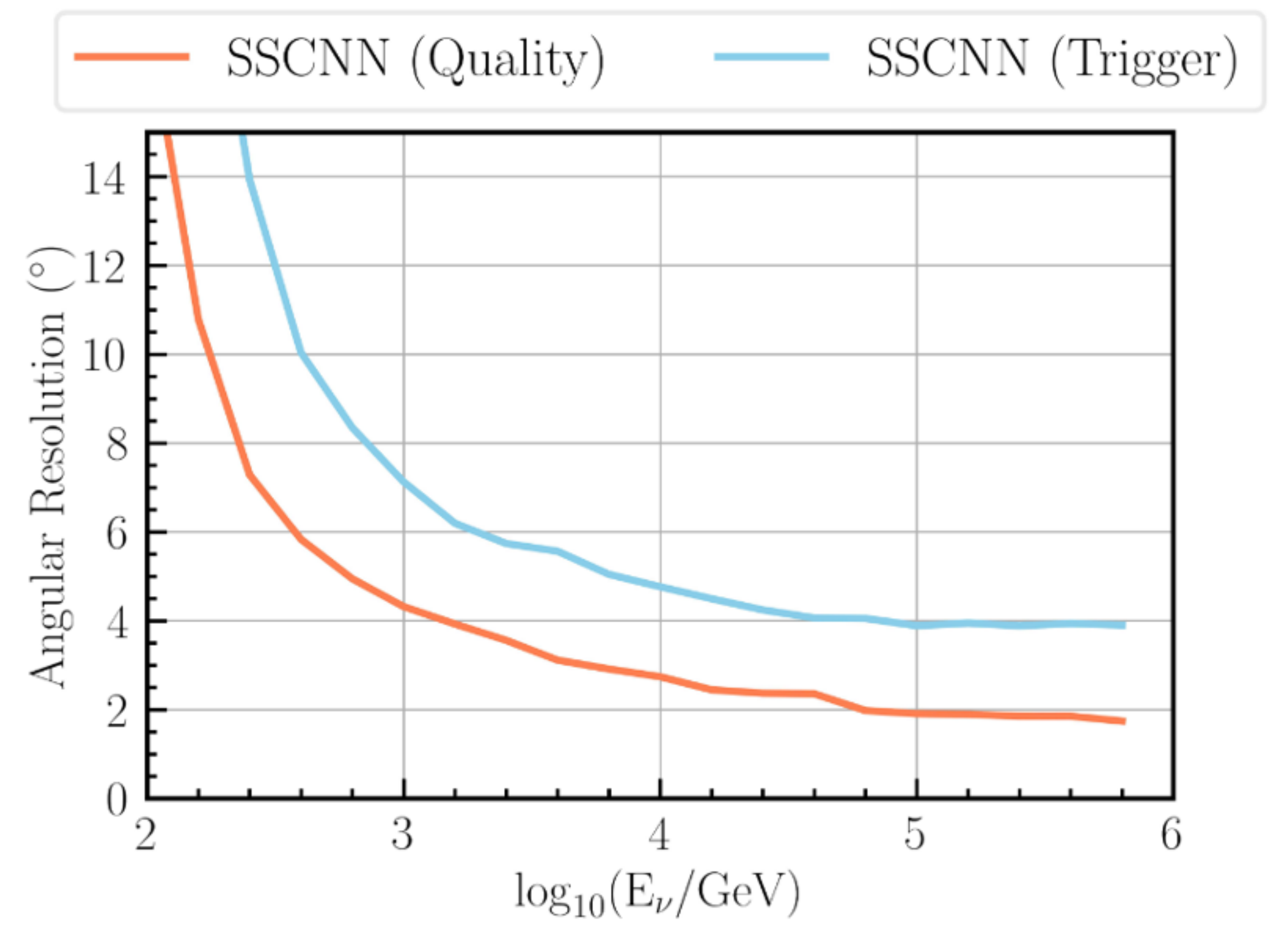
Train a 4D SSCNN to do energy and angular reconstruction on Prometheus (IceCube-like) events

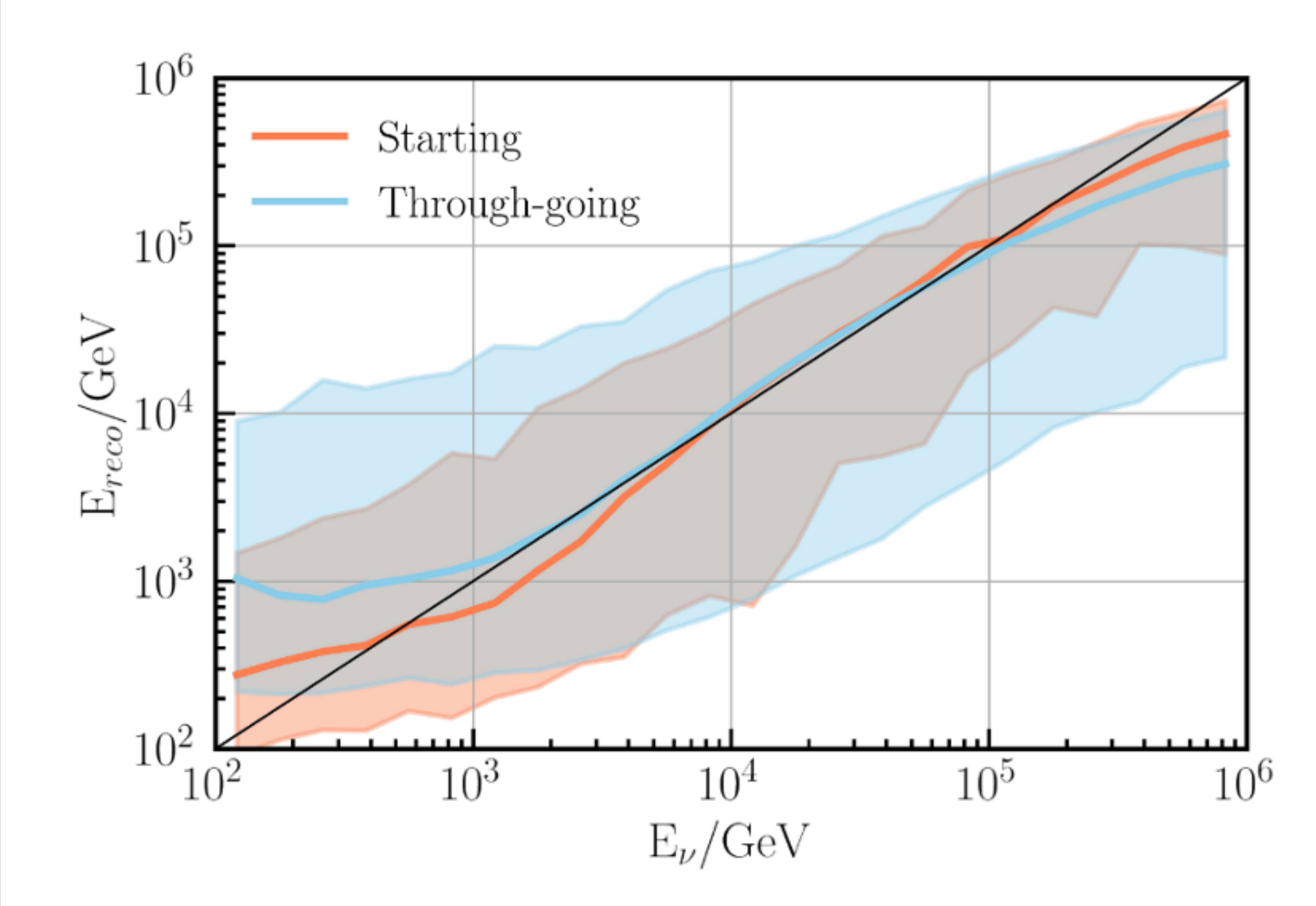
Large GPU batching (memory-efficient) allows for **sub-ms** per-event average runtime

Per-event average runtime

SSCNN Angular (GPU)	**0.101 ± 0.003 ms**
SSCNN Energy (GPU)	**0.103 ± 0.008 ms**
SSCNN Angular (CPU)	**37.7 ± 53.4 ms**
SSCNN Energy (CPU)	**30.6 ± 48.9 ms**
Likelihood Angular (CPU)	36 ± 152 ms
Likelihood Energy (CPU)	6.58 ± 23 ms

< 100ms per-event average runtime on CPU sequentially (batch size of 1)





Outline

- Improve reconstruction speed
  - Efficient ML reconstruction of direction/energy using sparse submanifold CNNs (SSCNN)
- Representation learning
  - Learning effective and compact representations of neutrino telescope events
- Future plans for application in IceCube (WIP)
  - Fast ML at early levels in data pipeline

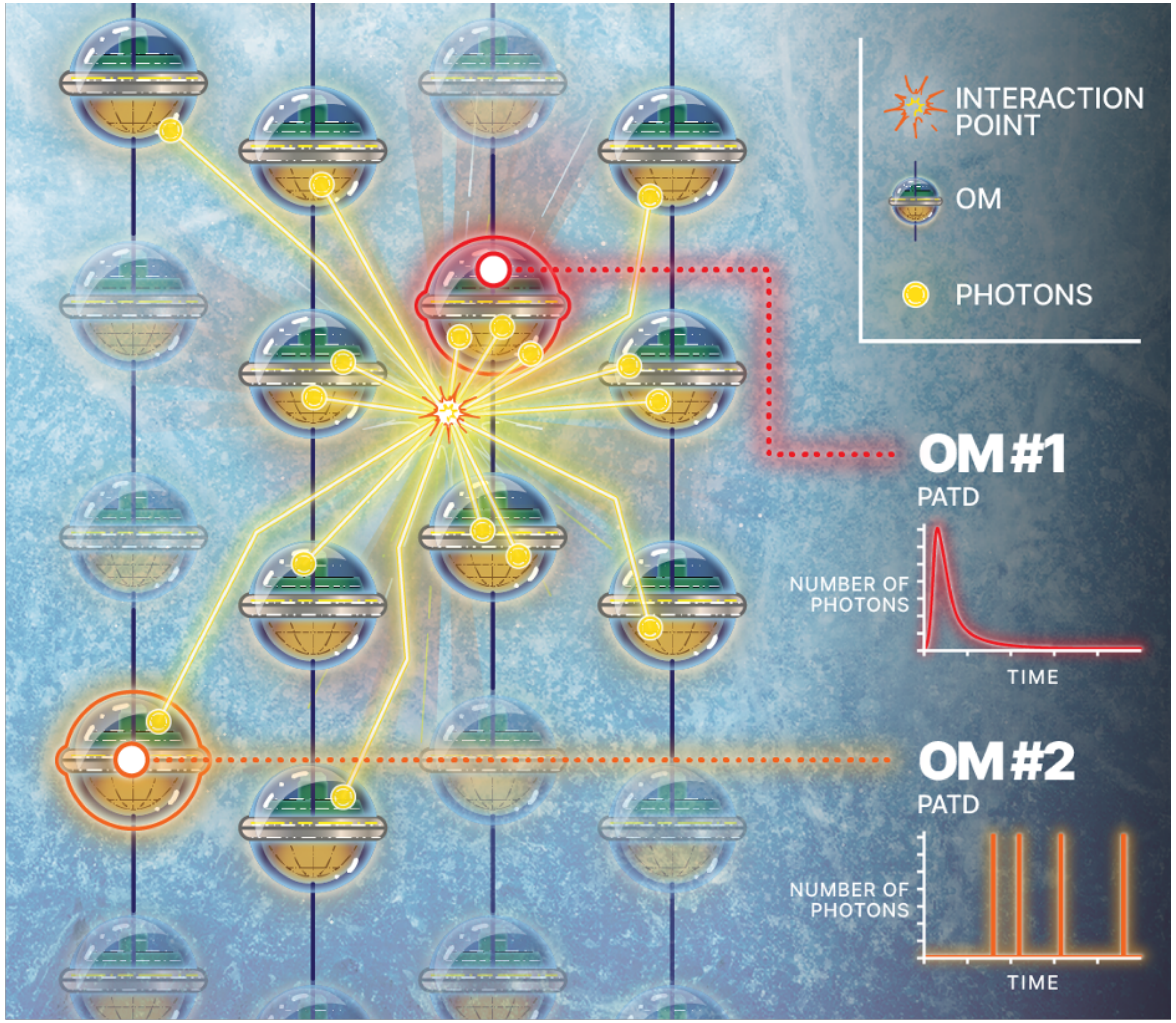
Two major challenges:

Spatial: Neutrino telescope data is extremely sparse and of large-scale

Temporal: Events can span over thousands of nano-seconds, but fine timing resolution is important for certain analyses

Learning Representations of Events

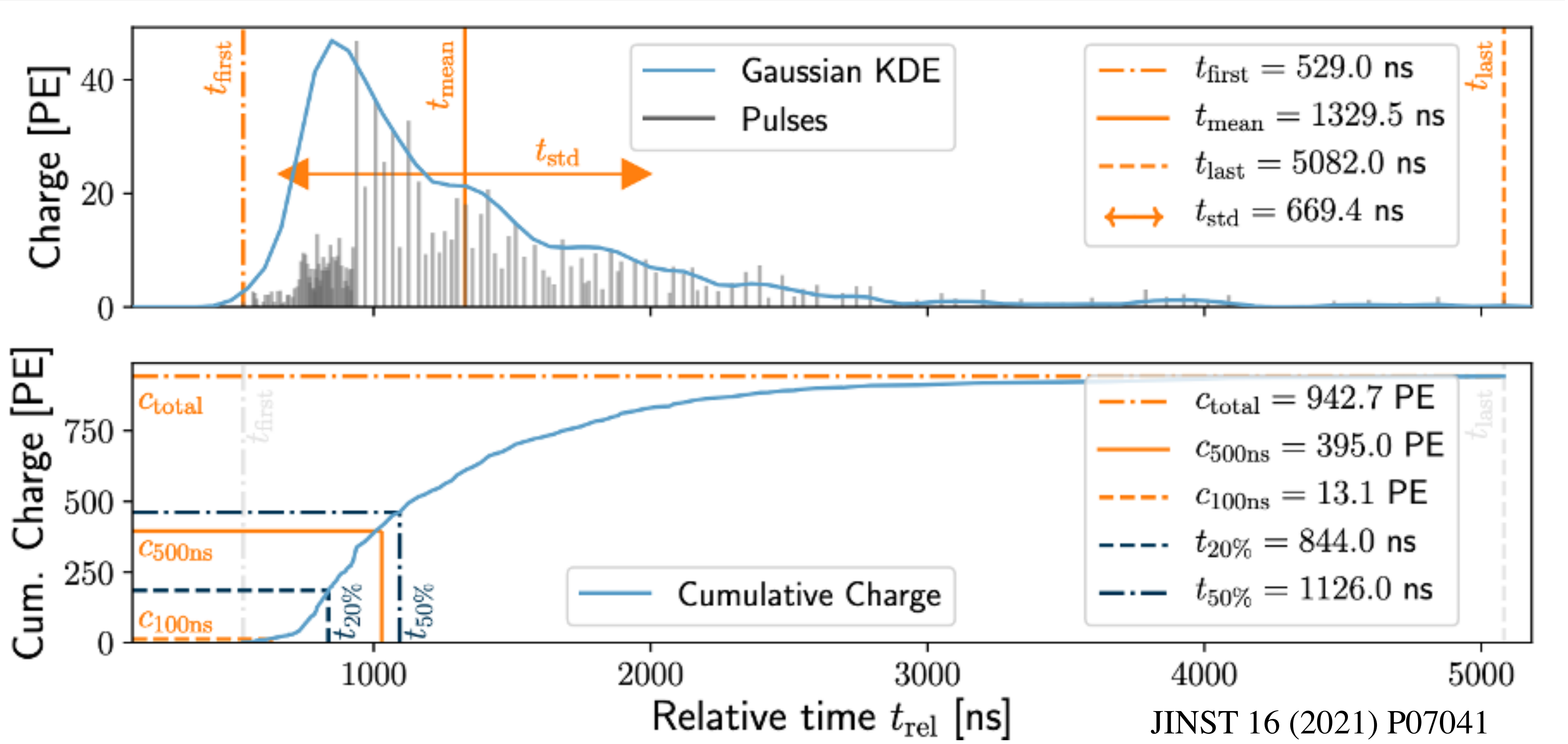
- We can view an event as a set of optical modules (OM) that saw light, and the series of "pulses" associated with that OM
- Computationally intensive to process all hits/pulses in a 4D manner (hundreds to thousands per OM at high energies)
- **Idea**: compress/summarize each OMs timing information into a fixed-size parameterization (reducing the problem from 4D -> 3D)



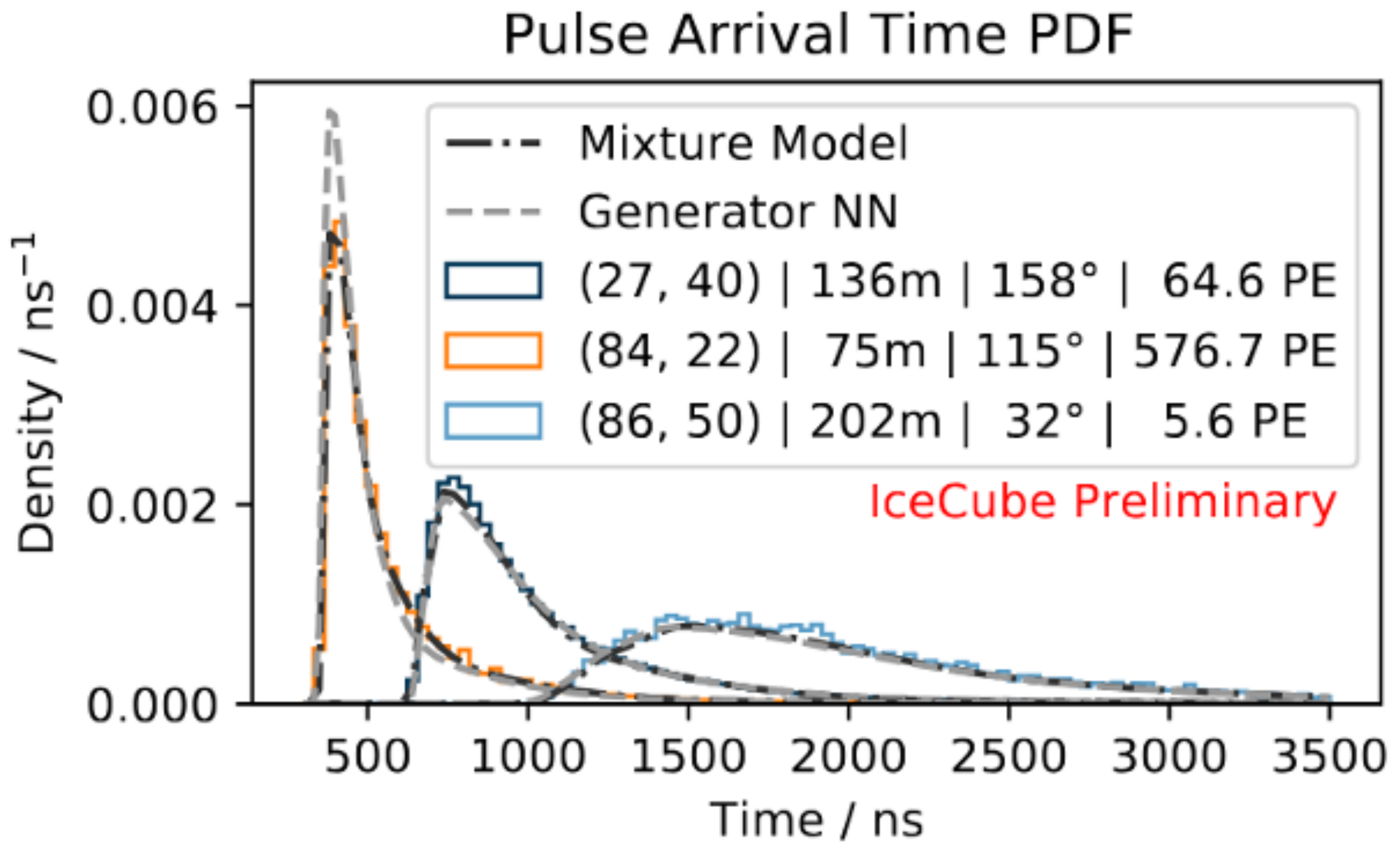
J. Pairin

Learning Representations of Events

- Some existing solutions:
  - **Summary statistics**: 9 statistical variables derived from the pulse series
  - **Asymmetric Gaussian mixture model (Event-Generator)**: fit the parameters of a mixture of asymmetric Gaussians using neural networks



JINST 16 (2021) P07041

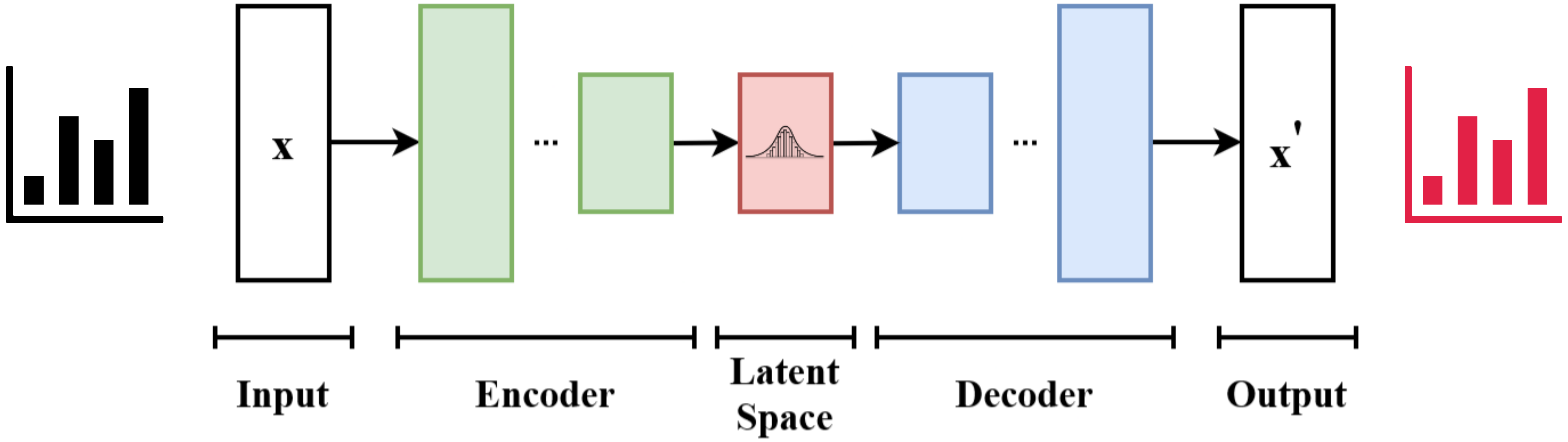


PoS-ICRC2021-1065

First pass: vanilla variational autoencoder

- First pass at a new idea: variational autoencoders

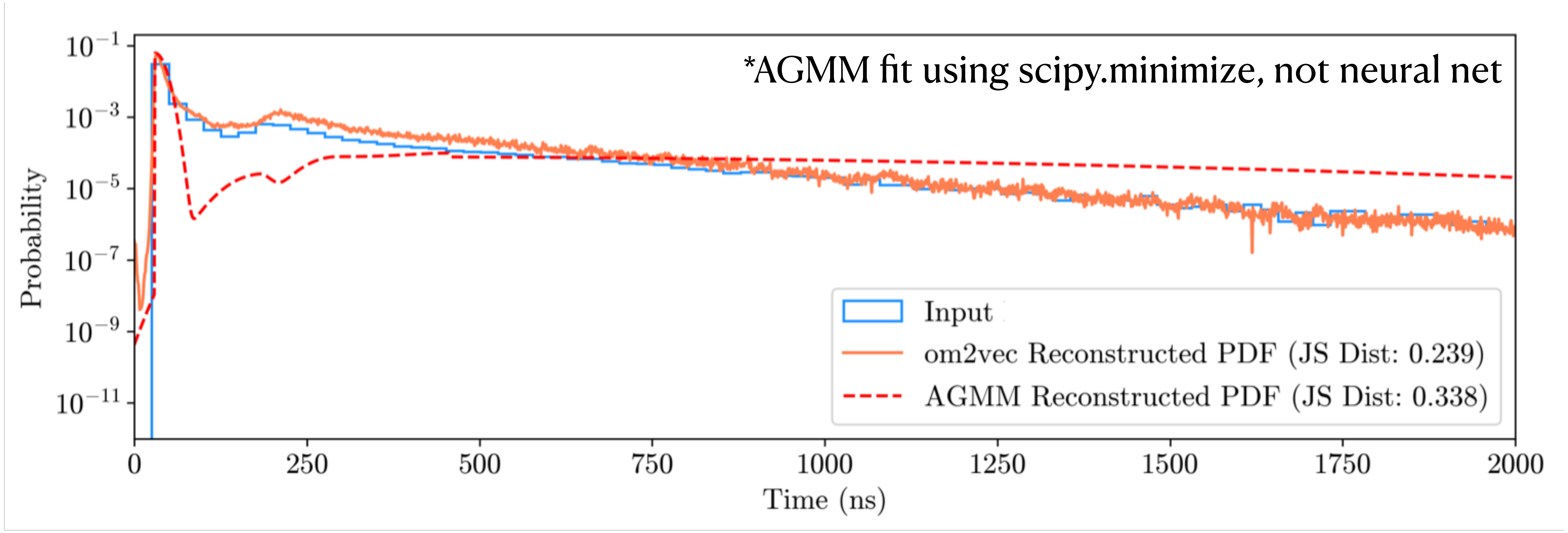
binned pulse series



reconstructed binned pulse series

- VAE learns to encode and decode binned pulse series to a smaller latent space
- Idea is that the latents are a information-rich **representation** of the pulse series, which we can use as an data-driven **summarization** of the timing information

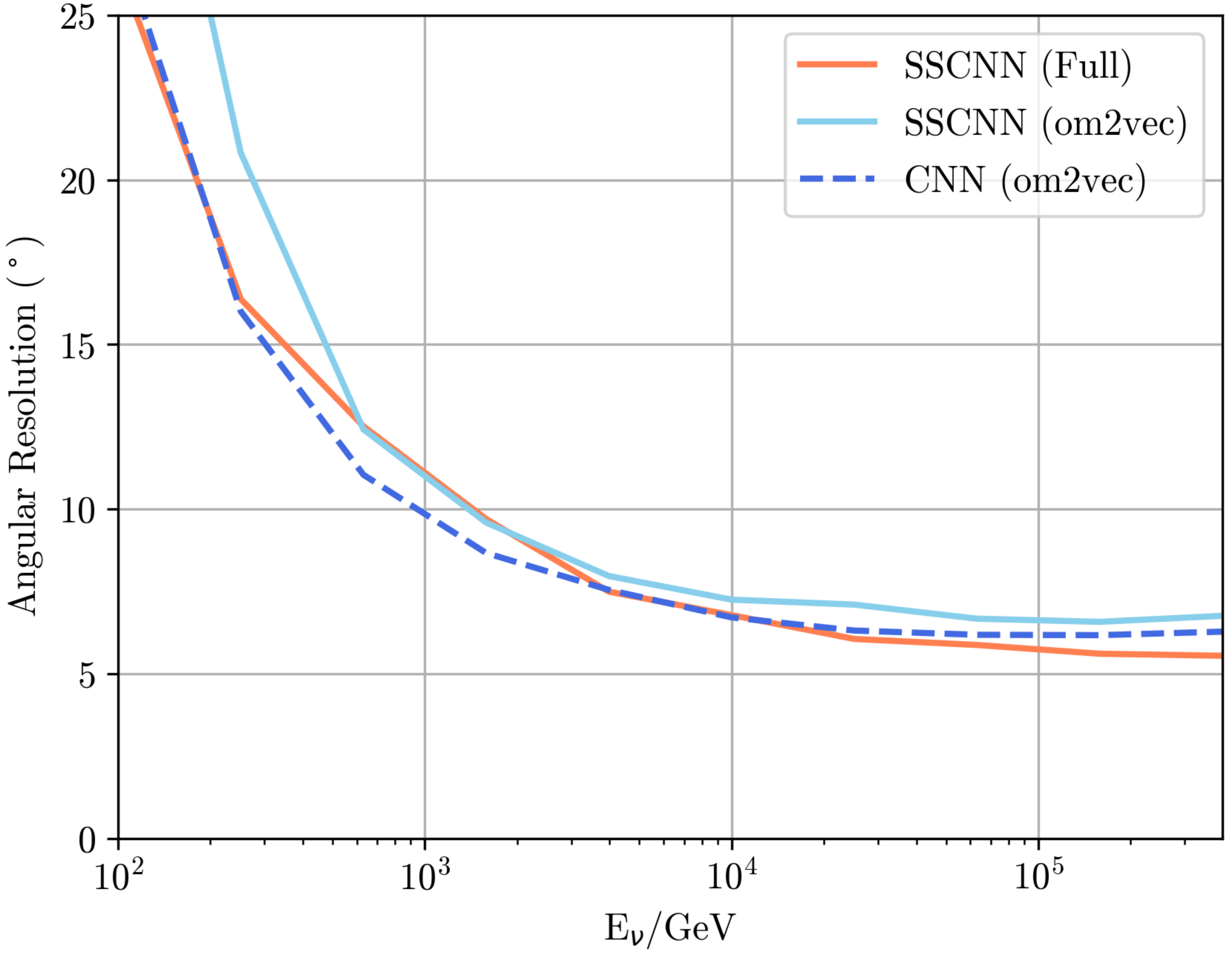
Proof-of-concept study with Prometheus



- Conducted proof-of-concept study with Prometheus events, with "**om2vec**" VAE
- **Important note**: Prometheus events use individual photons hits and not pulses (which would be data from a real experiment), so this is an idealized case study

Combining SSCNN and VAEs

- Combining SSCNN with the om2vec VAE for angular reconstruction, we can reduce our problem from 4D to 3D by reducing the time dimension:
  - **SSCNN (Full)**: is the 4D SSCNN shown previously
  - **SSCNN (om2vec)**: uses latents from VAE, summarizing the time dimension (3D)
  - **CNN (om2vec)**: 2D standard ResNet, also using latents from VAE and arranged into 2D images



SSCNN (om2vec) is **~4x faster** than SSCNN (Full) on GPU

Outline

- Improve reconstruction speed
  - Efficient ML reconstruction of direction/energy using sparse submanifold CNNs (SSCNN)
- Representation learning
  - Learning effective and compact representations of neutrino telescope events
- Future plans for application in IceCube (WIP)
  - Fast ML at early levels in data pipeline

Two major challenges:

Spatial: Neutrino telescope data is extremely sparse and of large-scale

Temporal: Events can span over thousands of nano-seconds, but fine timing resolution is important for certain analyses

IceCube Data Pipeline

IceCube has been collecting data for >10 years (>315 million seconds)

Level	Data Rate (events/sec)
Trigger	~2700
Muon Filter (Level 1)	~45
Muon Filter (Level 2)	~2
Analysis-specific data reduction cuts	Varies
Analysis final levels	Varies

2016 *JINST* **11** P11009

Data rate dominated by air shower backgrounds (atmospheric μ)

Even higher data rates for water-based/larger neutrino telescopes

T Chiarusi *et al* 2017 *J. Phys.: Conf. Ser.* **898** 032042

Atm. μ : Atm. ν_μ : Astrophysical ν_μ ratio is ~ $\mathbf{10^9 : 10^3 : 1}$

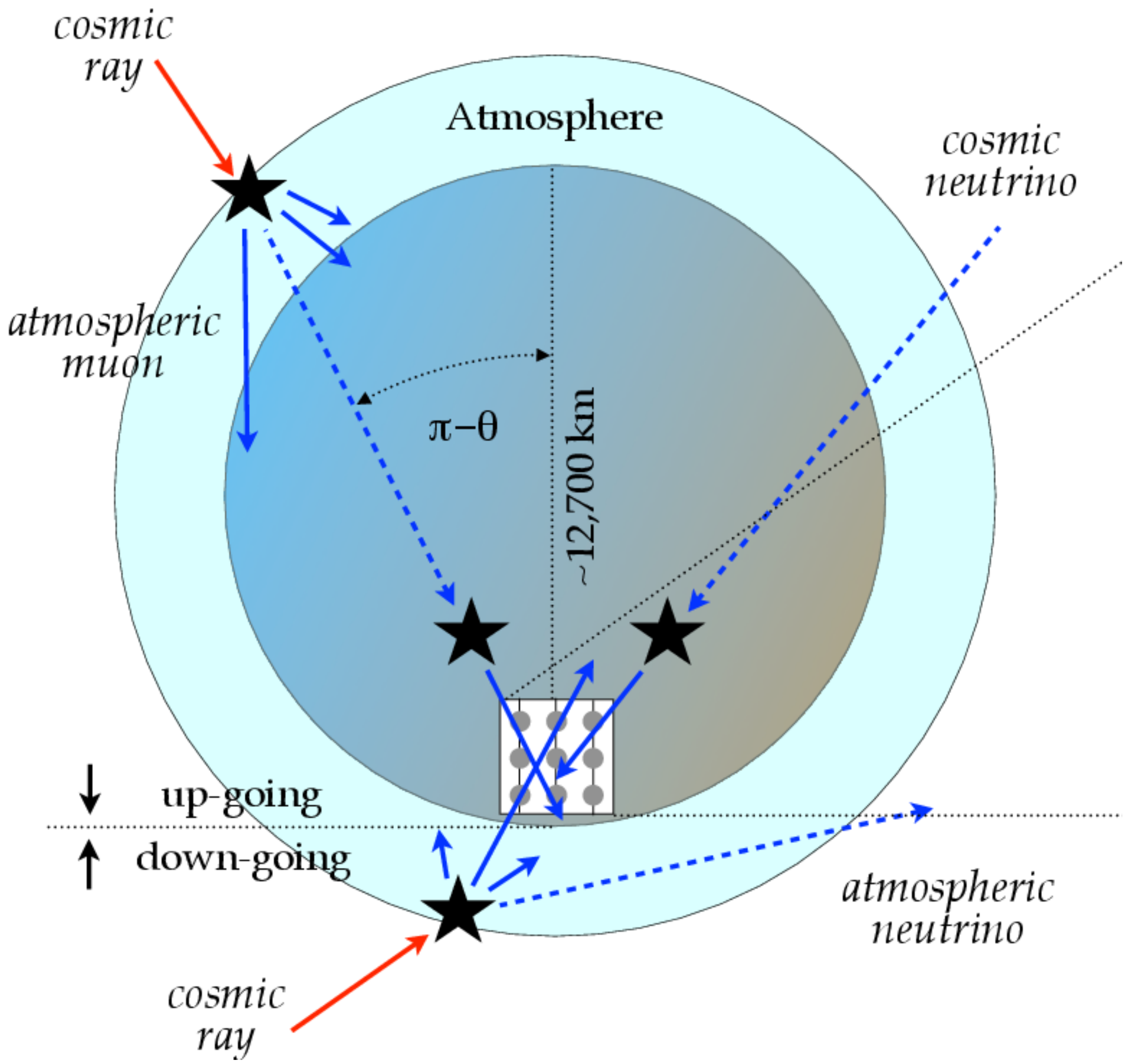
IceCube Data Pipeline

IceCube has been collecting data for >10 years (>315 million seconds)

Level	Data Rate (events/sec)
Trigger	~2700
Muon Filter (Level 1)	~45
Muon Filter (Level 2)	~2
Analysis-specific data reduction cuts	Varies
Analysis final levels	Varies

2016 *JINST* **11** P11009

Atm. μ can be cut out using directional reconstruction



IceCube Data Pipeline

Level	Data Rate (events/sec)	Typical Directional Reco Method
Trigger	~2700	Simple line-fitting algorithms
Muon Filter (Level 1)	~45	Simple maximum likelihood methods
Muon Filter (Level 2)	~2	Complex maximum likelihood methods
Analysis-specific data reduction cuts	Varies	Complex maximum likelihood methods
Analysis final levels	Varies	Complex max likelihood/ Machine learning

2016 *JINST* **11** P11009

Usually, ML is only used after significant data reduction steps after Level 2 filters, due to GPU and runtime constraints

Filters and subsequent analysis-specific cuts rely on max-likelihood methods

IceCube Data Pipeline

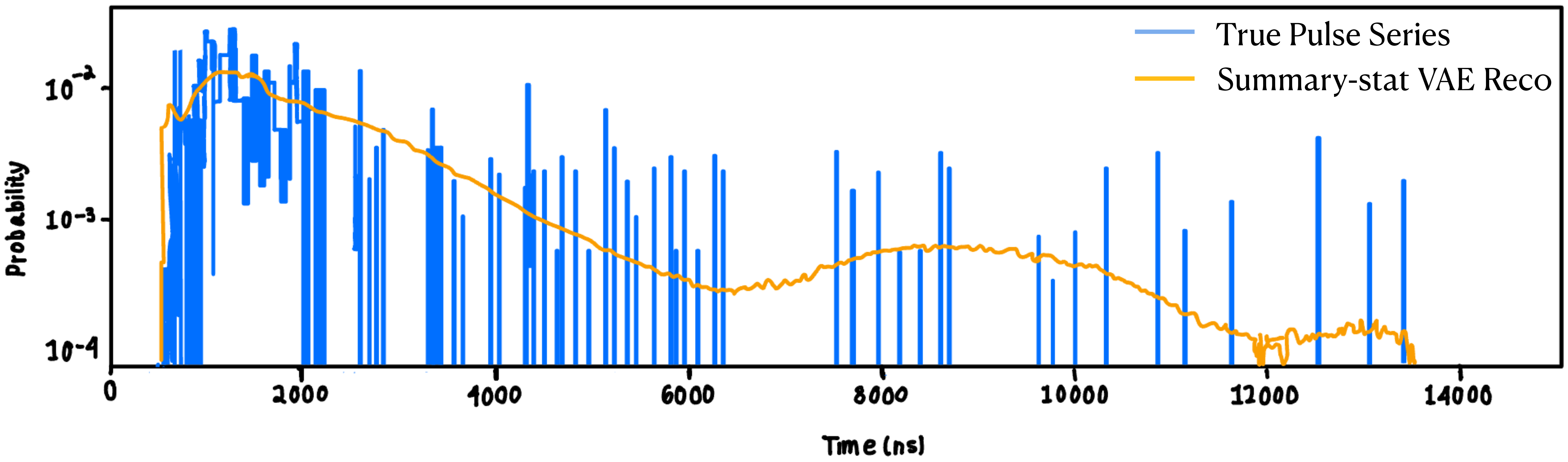
Level	Data Rate (events/sec)	Typical Directional Reco Method
Trigger	~2700	Simple line-fitting algorithms
Muon Filter (Level 1)	~45	Simple maximum likelihood methods
Muon Filter (Level 2)	~2	Complex maximum likelihood methods
Analysis-specific data reduction cuts	Varies	Complex maximum likelihood methods
Analysis final levels	Varies	Complex max likelihood/ Machine learning

2016 *JINST* **11** P11009

The goal is to use SSCNN, VAEs, and other fast ML techniques to push upwards in the pipeline

Fast ML

Current and Future Works



$$\text{latent} = [\underbrace{c_{total}, c_{500ns}, c_{100ns}, t_{20\%}, t_{50\%}, t_{first}, t_{mean}, t_{last}, t_{std}}_{\text{Summary stats}}, \underbrace{1.8078, \ldots, 2.078}_{\text{Encoded timing info}}]$$

- SSCNN implemented on IceCube data, working as expected for Level 2 Muon Filter events
- VAEs for learning representations of IceCube pulse series (WIP, difficult vs. Prometheus photon hits)
    - VAE + summary stats encoding
    - Normalizing flows?

Conclusions

- Neutrino telescope data is **spatially sparse**, and many downstream tasks benefit from **fine timing resolution**
- These challenges can be addressed with **SSCNNs** and **VAE** latent representations for flexible, efficient and performant reconstructions
- On-going work to incorporate these ML techniques into **earlier levels** of the IceCube data pipeline

Thank you!

felixyu@g.harvard.edu