Event identification in the NEXT experiment using CNNs

J.M. Benlloch-Rodríguez[1], P. Ferrario[2] ,
J.Renner[1], **M. Kekic**[1]
(NEXT Collaboration)
[1]*IFIC/Universitat de València*
[2]*DIPC/Ikerbasque*

The NEXT experiment

LSC Laboratorio Subterráneo Canfranc

NEXT

USA

University of Texas Arlington

Argonne National Laboratory

Harvard

Fermilab

Texas A&M

Iowa State University

Pacific Northwest National Laboratory

Spain

DIPC

IFIC Institut de Física Corpuscular

Universitat de Girona

USC Universidade de Santiago de Compostela

Universidad de Zaragoza

Universitat Politècnica de València

Portugal, Israel, Colombia

Universidade de Coimbra

universidade de aveiro

Ben-Gurion University of the Negev

UAN Universidad Antonio Nariño

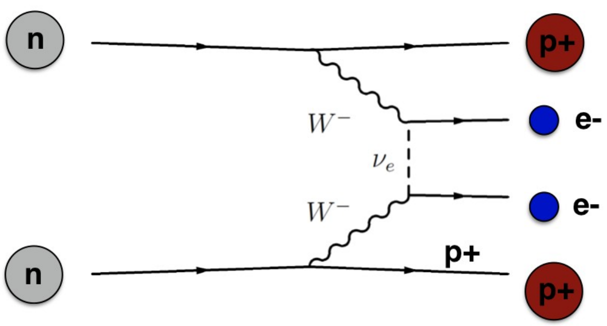
Co-spokespersons:
D. Nygren
J.J. Gomez-Cadenas

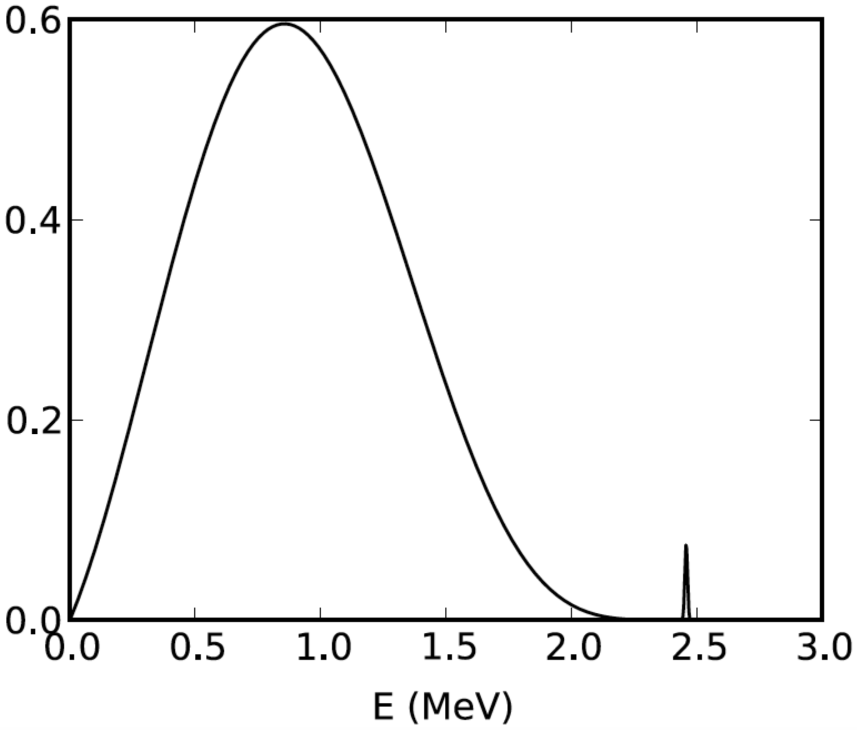
Funded by: erc, Marie Curie Actions, Gobierno de España – Ministerio de Economía y Competitividad, Generalitat Valenciana, FCT Fundação para a Ciência e a Tecnologia, U.S. Department of Energy, University of Texas Arlington

Neutrino Experiment with a Xenon TPC

- Next-WHITE (NEW) operating a 5 kg-scale demonstrator at the Canfranc Underground Laboratory (LSC)
- To be commissioned in 2020: 100 kg Xe, enriched to 136 Xe (90%)

Neutrinoless double beta decay

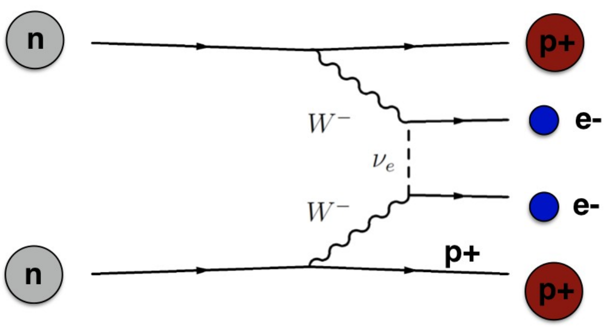


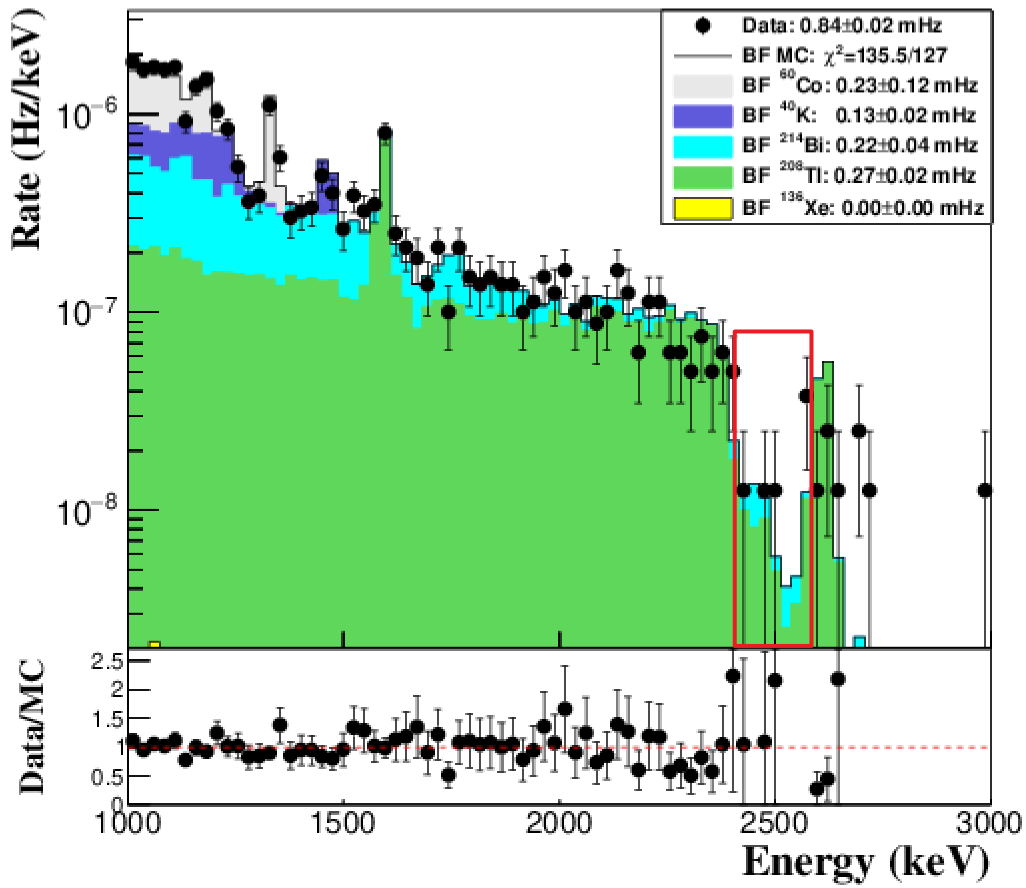


Essential:

1. Good energy resolution

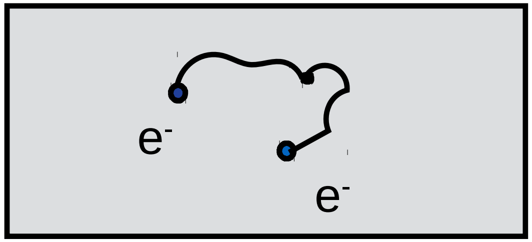
Neutrinoless double beta decay



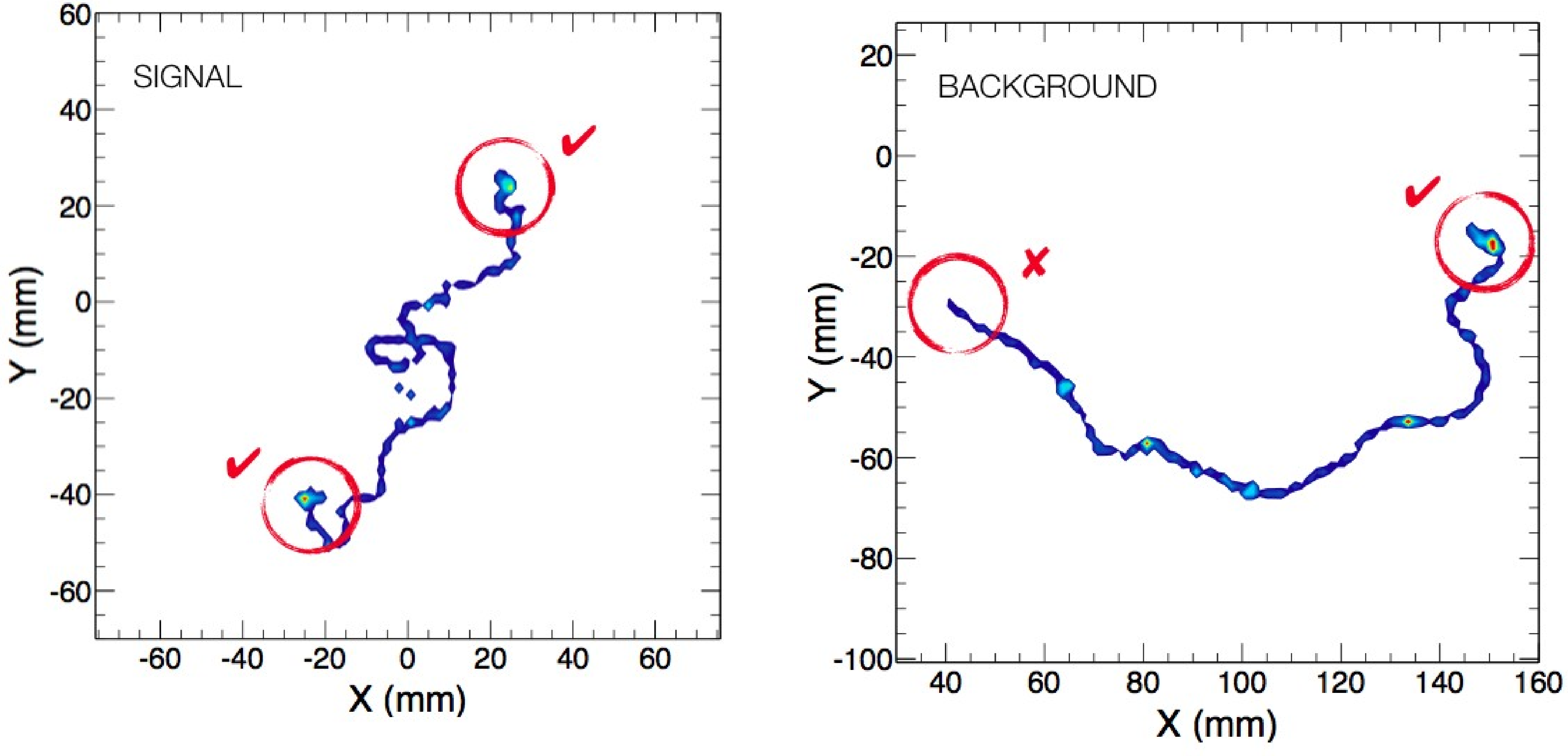


arXiv:1905.13625

Essential:
1. Good energy resolution
2. Good background identification



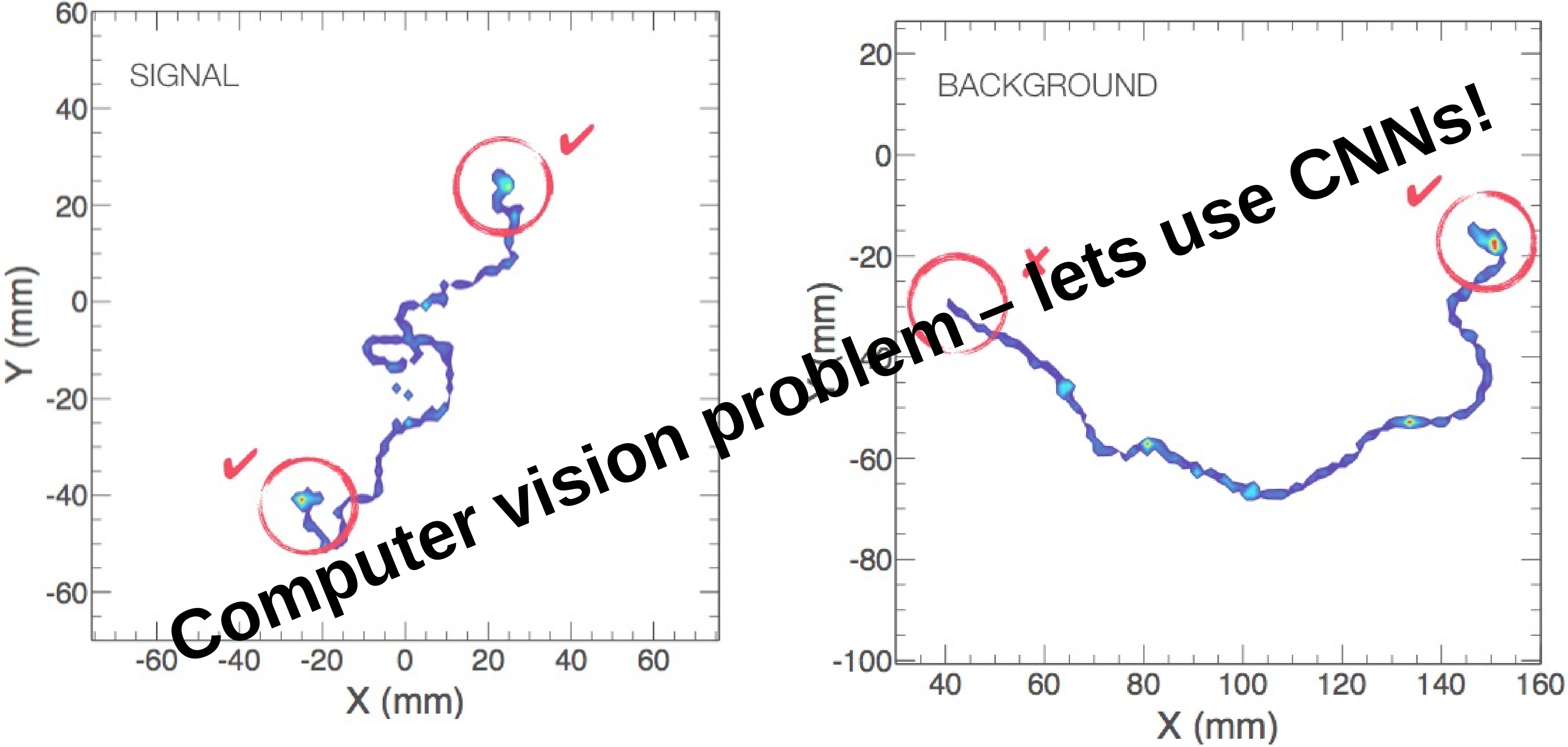
Background identification



At the end of the track the deposition of the energy increases - Bragg peak (blob):

- Signal : 2 blobs
- Background : 1 blob

Background identification



Computer vision problem – lets use CNNs!

At the end of the track the deposition of the energy increases - Bragg peak (blob):

- Signal : 2 blobs
- Background : 1 blob

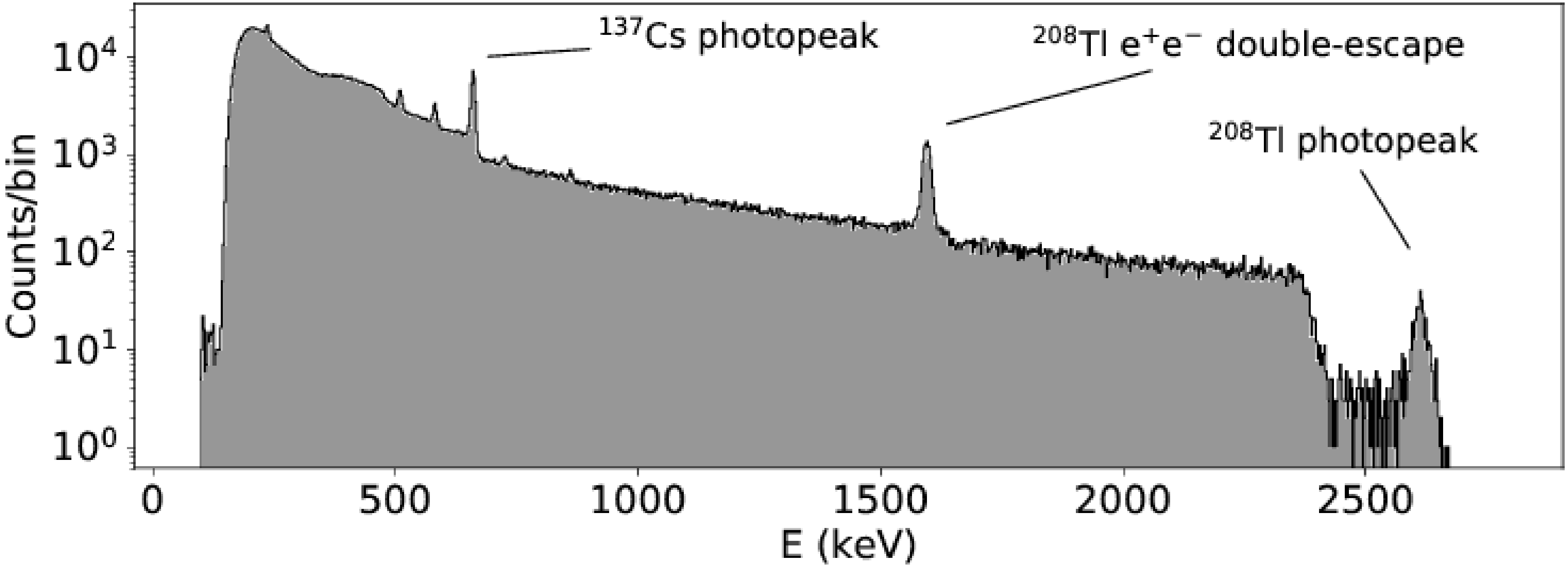
Previous study

- **Initial Monte Carlo-based study:** arXiv:1609.06202

Cut	Signal Events		BG Events (^{208}Tl)		BG Events (^{214}Bi)	
	$2\times2\times2$	$10\times10\times5$	$2\times2\times2$	$10\times10\times5$	$2\times2\times2$	$10\times10\times5$
(Initial events)	1.0	1.0	1.0	1.0	1.0	1.0
Energy	7.59×10^{-1}	7.59×10^{-1}	2.27×10^{-3}	2.27×10^{-3}	1.42×10^{-4}	1.42×10^{-4}
Fiducial	6.71×10^{-1}	6.68×10^{-1}	1.19×10^{-3}	1.17×10^{-3}	8.62×10^{-5}	8.54×10^{-5}
Single-Track	3.75×10^{-1}	4.79×10^{-1}	7.90×10^{-6}	1.81×10^{-5}	3.84×10^{-6}	8.75×10^{-6}
Classification*	3.23×10^{-1}	3.67×10^{-1}	7.70×10^{-7}	2.41×10^{-6}	2.90×10^{-7}	9.59×10^{-7}
Classification (DNN)	**3.23×10^{-1}**	**3.67×10^{-1}**			**1.80×10^{-7}**	**8.22×10^{-7}**

But now we have the data **measured** in our detector

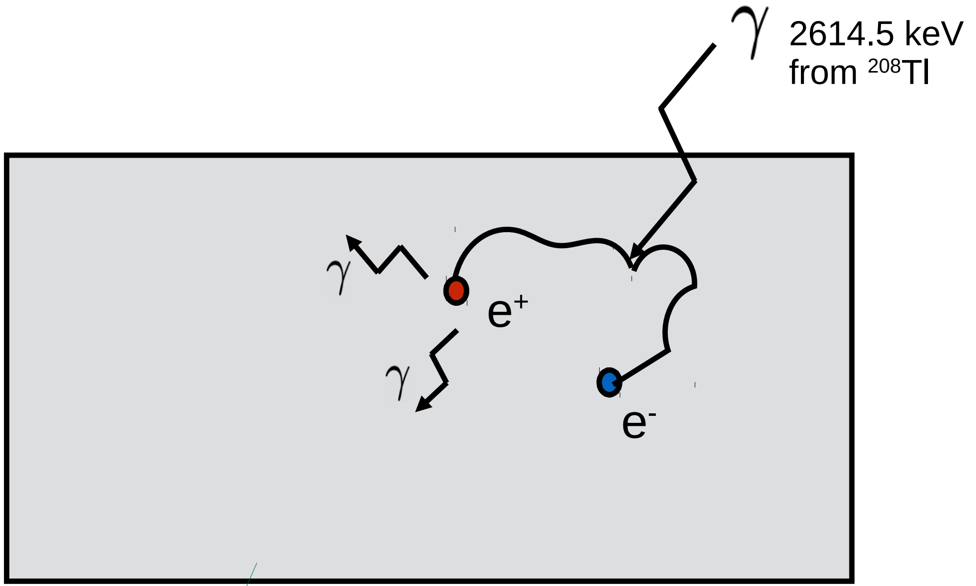
Data energy spectrum



arXiv:1905.13110 (2019)

Calibration with ^{137}Cs and ^{228}Th sources
Achieved <1% Resolution at FWHM near $Q_{\beta\beta}$

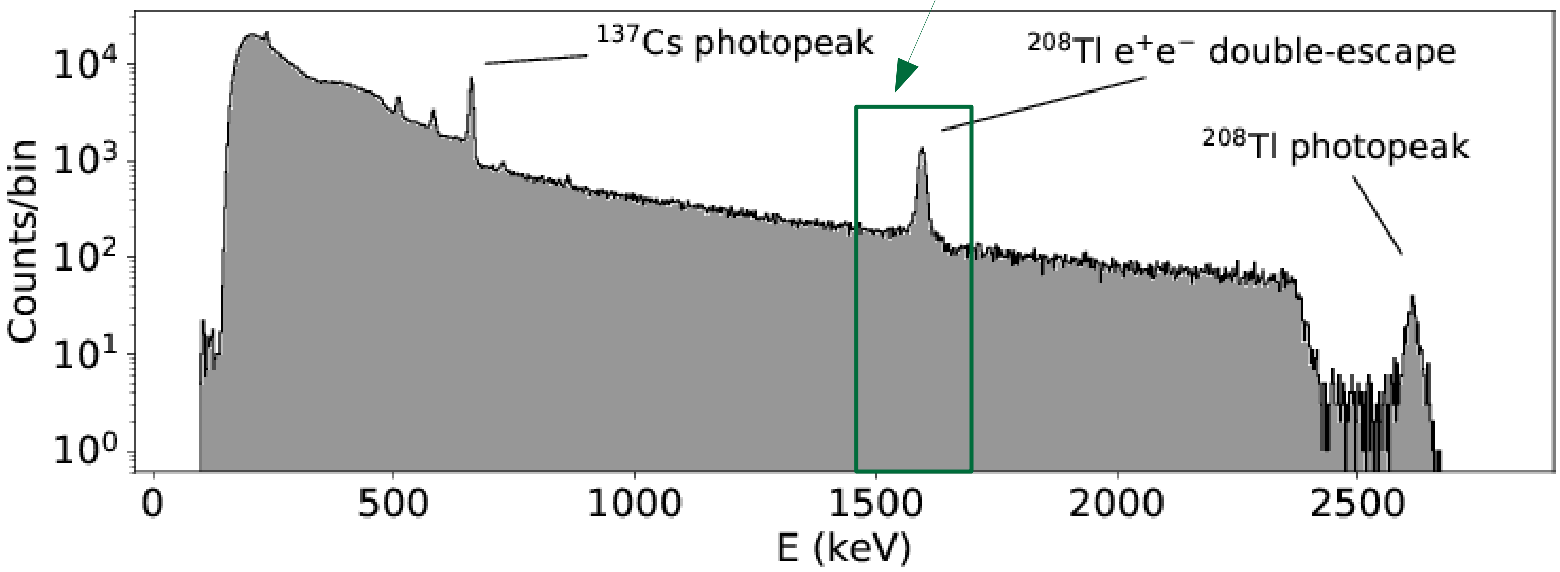
Proof-of-concept with e^+e^- track



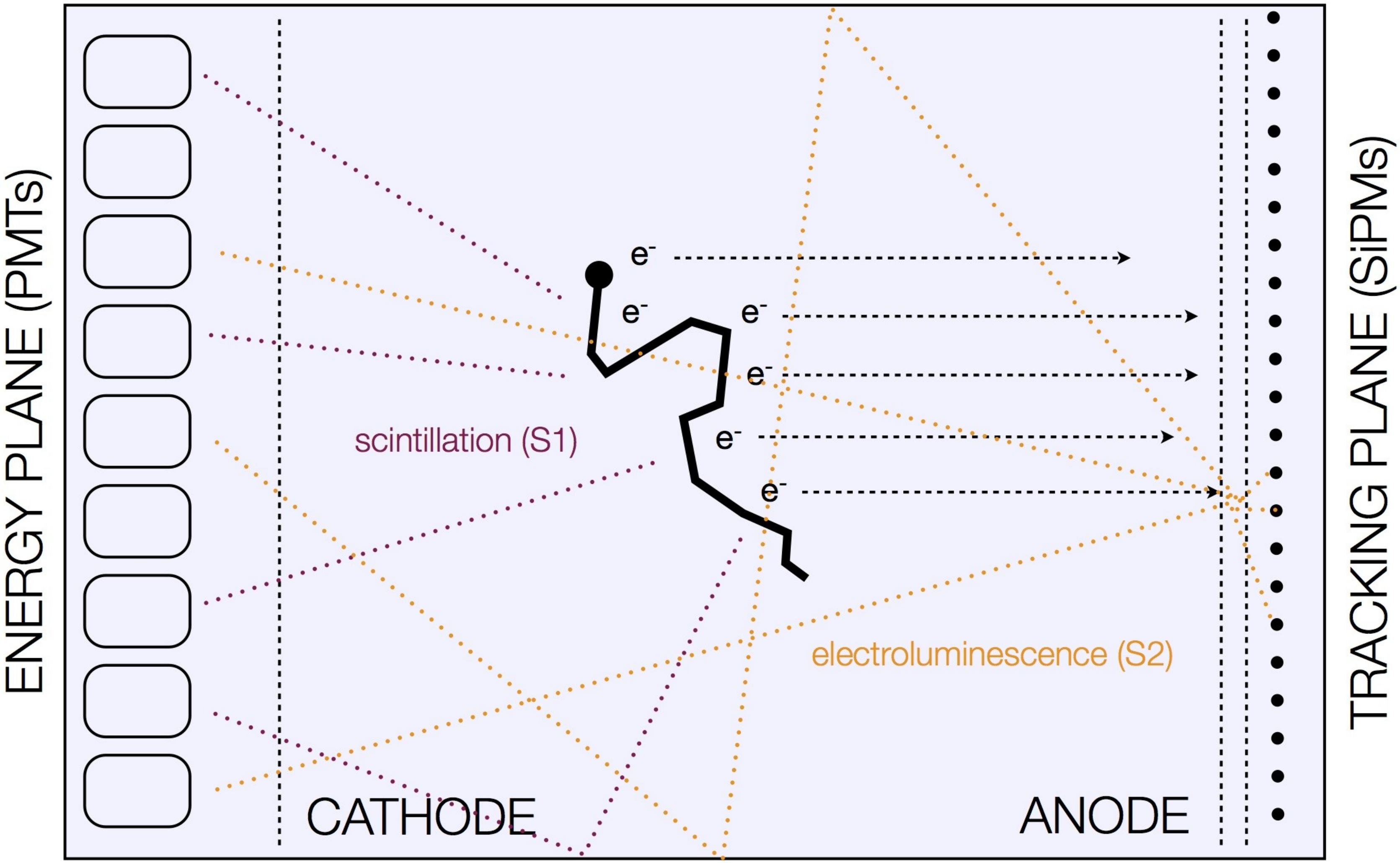
Of course no 0νßß events but e^+e^- pair leaves similar topological signature (we can test our model):

- Signal : e^+e^- track
- Background : single e^- track

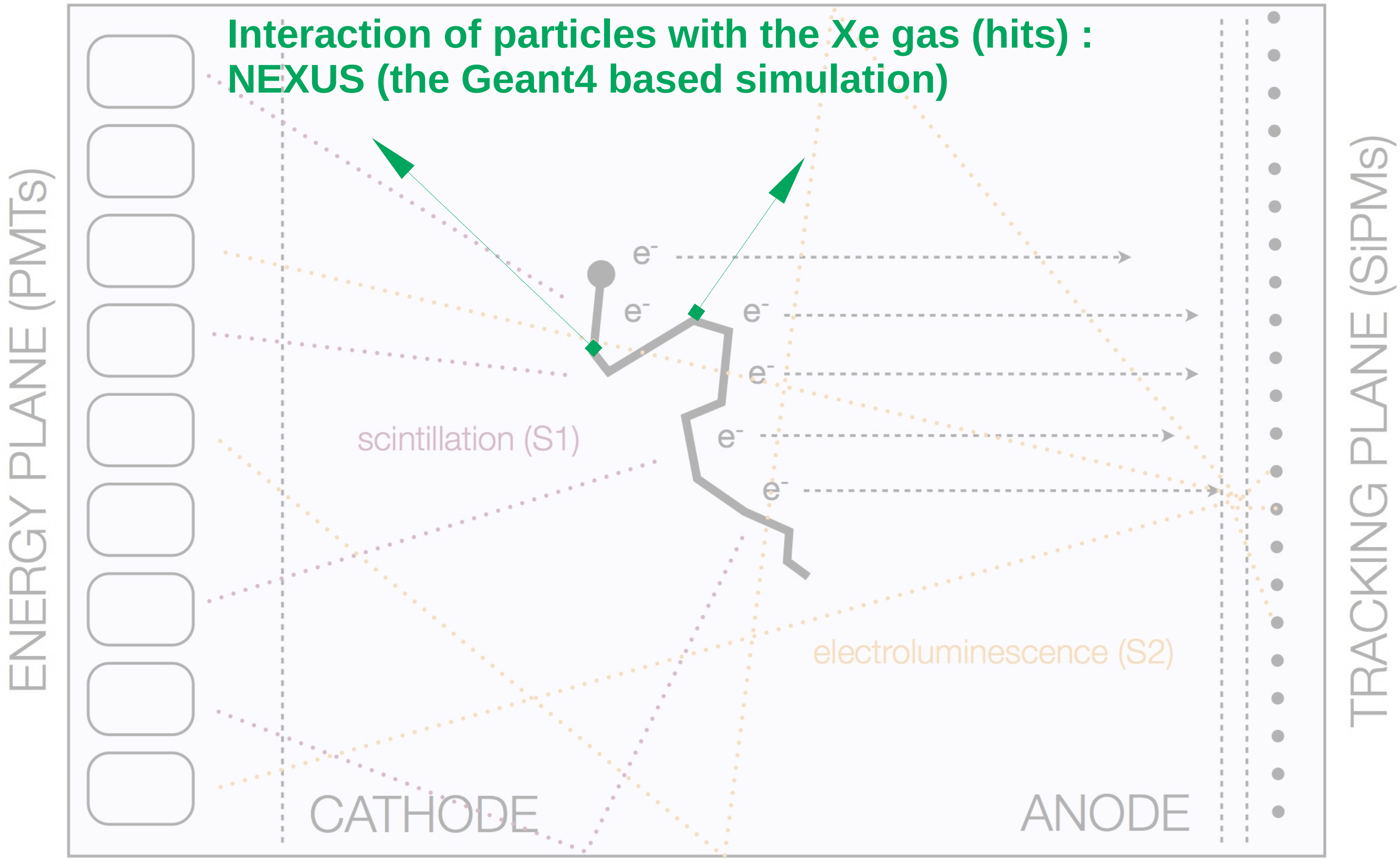
Can calculate ratio of the signal (gaussian) and the background (exponential)



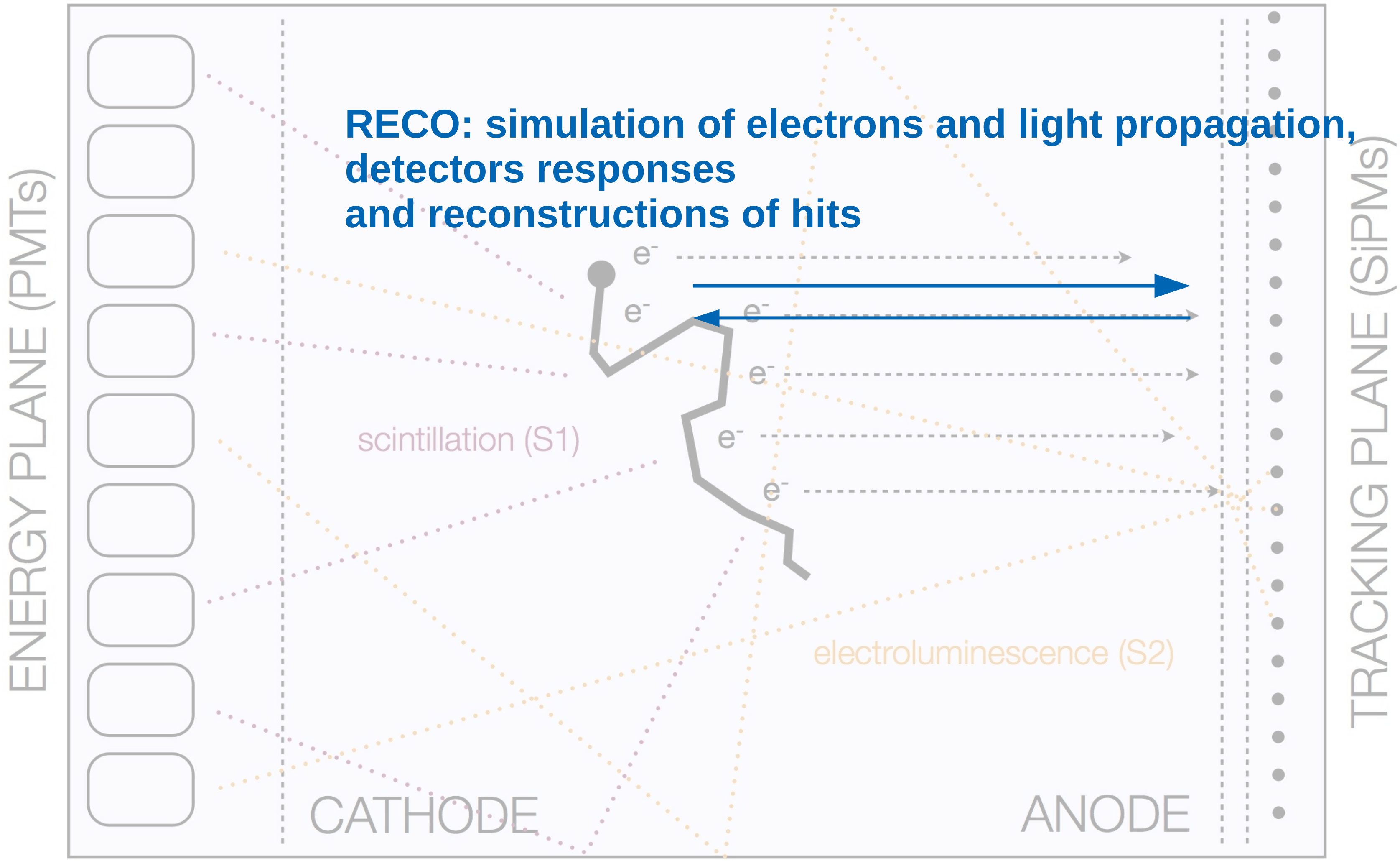
NEW : simulation and reconstruction



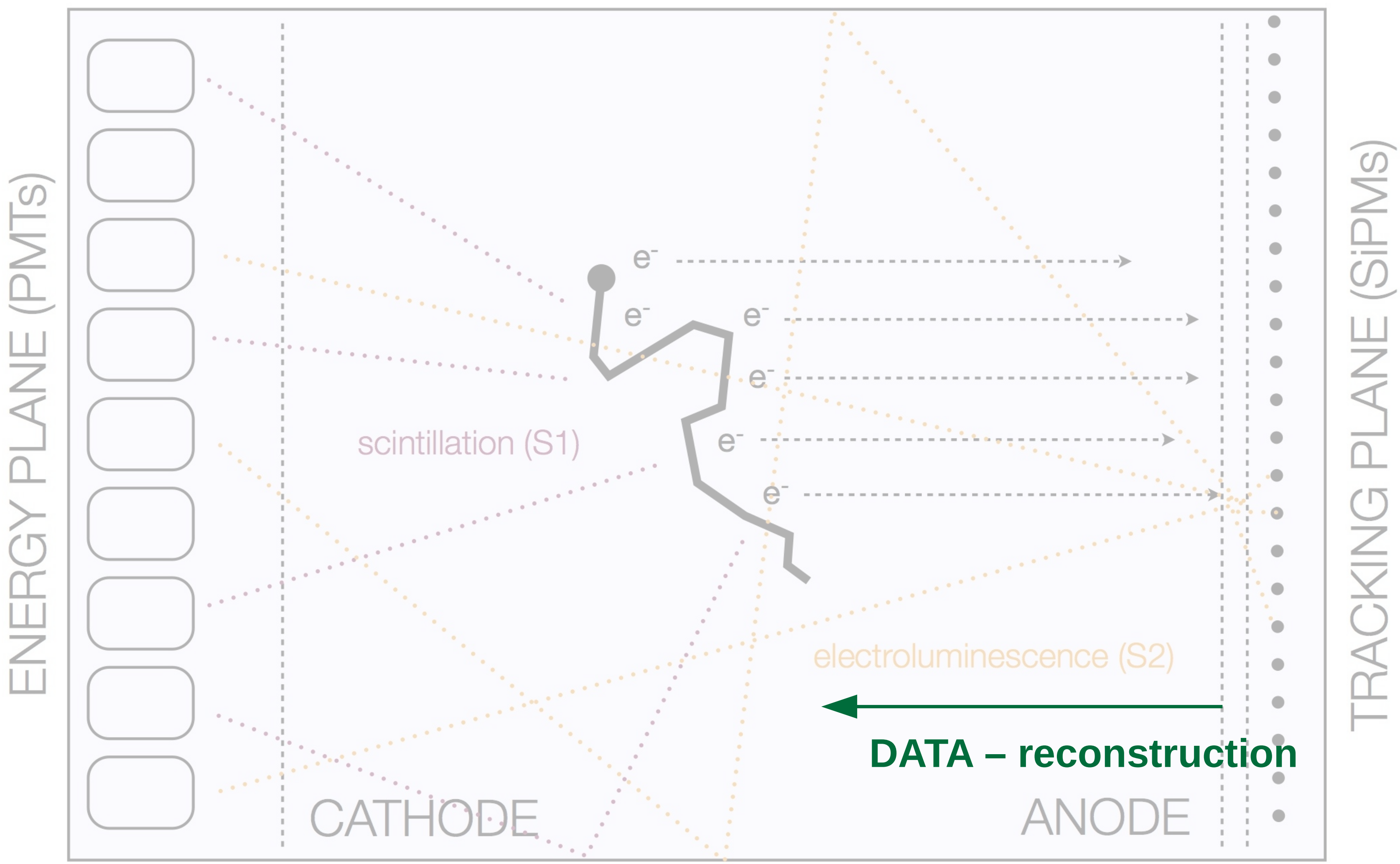
NEW : simulation and reconstruction



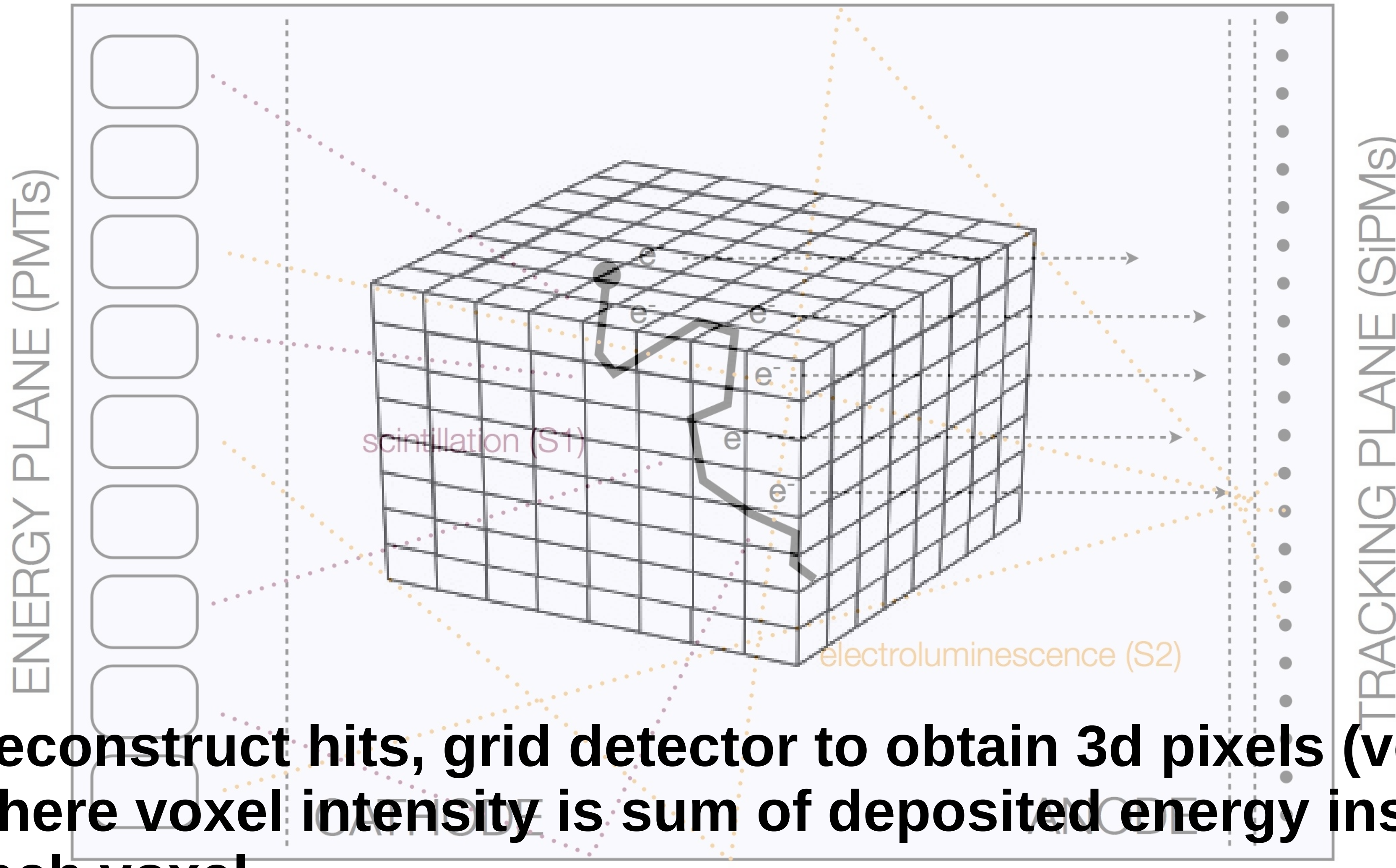
NEW : simulation and reconstruction



NEW : simulation and reconstruction

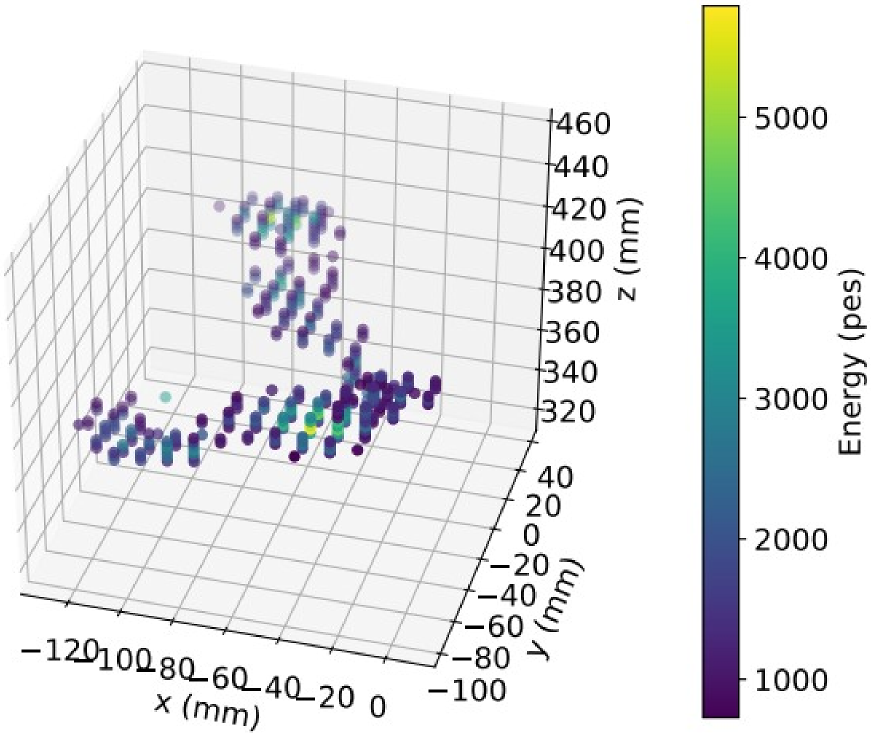


NEW : simulation and reconstruction

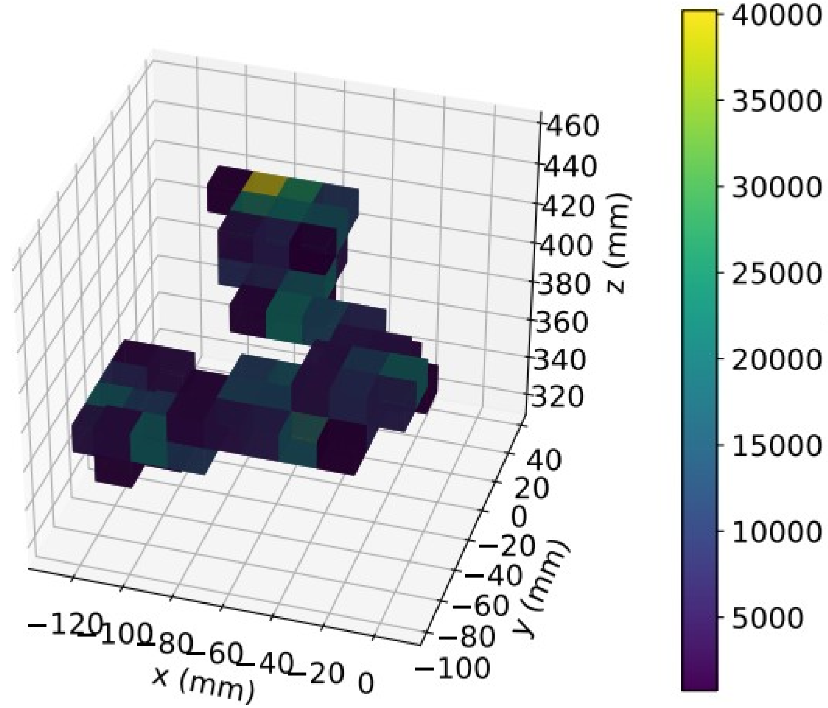


Reconstruct hits, grid detector to obtain 3d pixels (voxels) where voxel intensity is sum of deposited energy inside each voxel

Event example



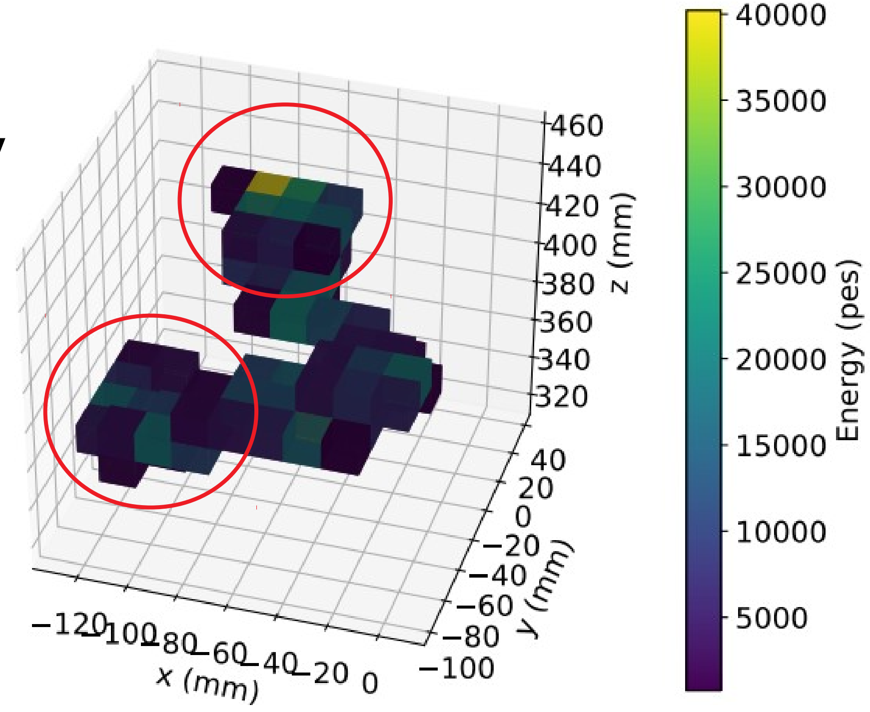
Reconstructed hits



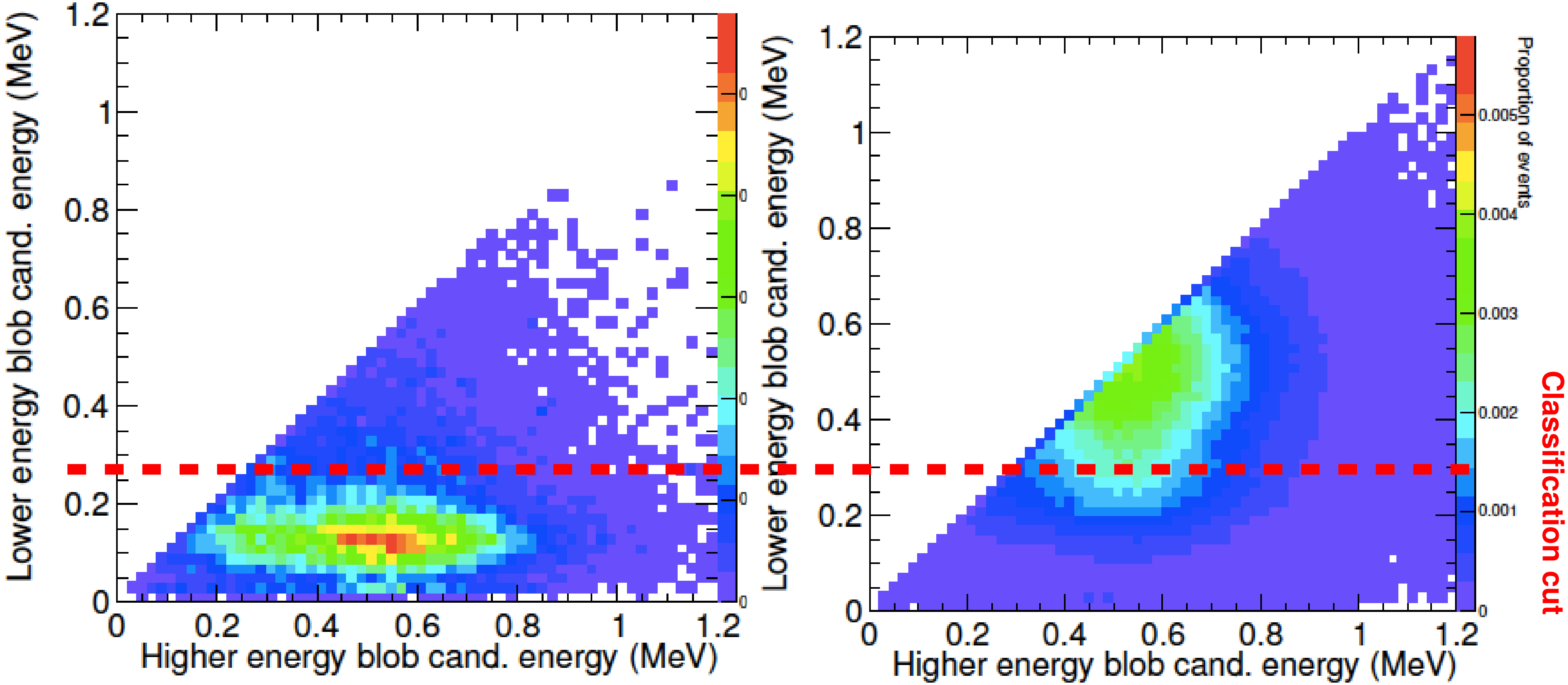
Voxelized hits

Classical approach

1. Find the track based on graph theory
2. Identify track extremes
3. Calculate energy inside the blobs



Classical approach



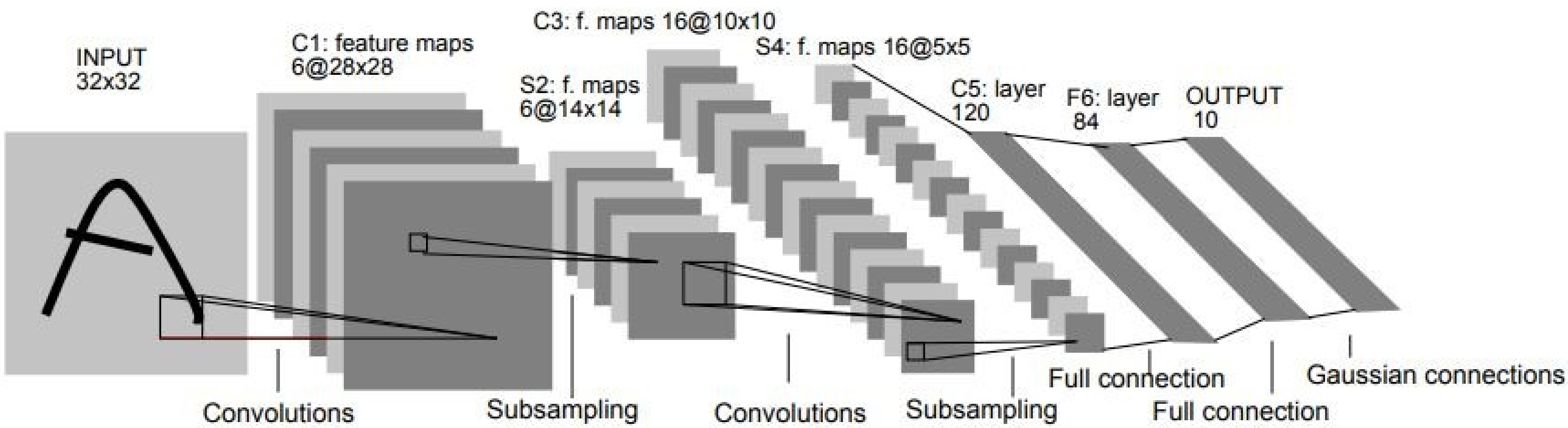
JHEP 01, 104 (2016) [arXiv:1507.05902]

Geant4 Monte Carlo:

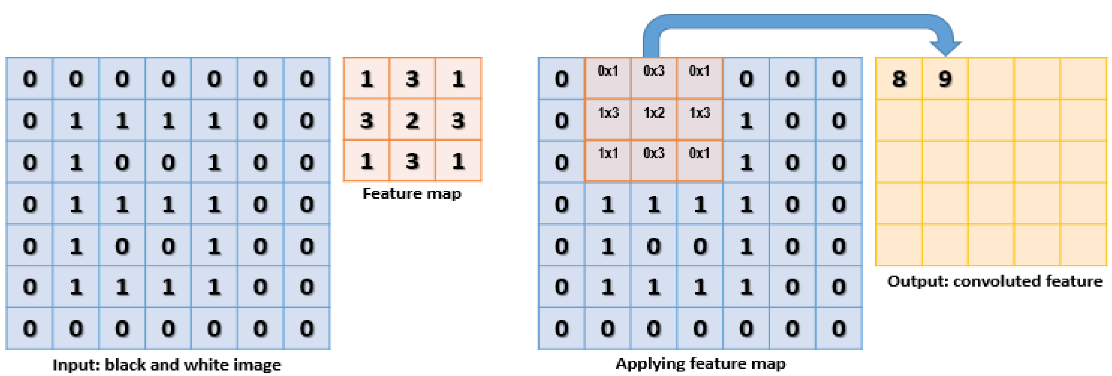
(left) 2.44 MeV gammas from ^{214}Bi decay, (right) 0vßß events

DNN approach

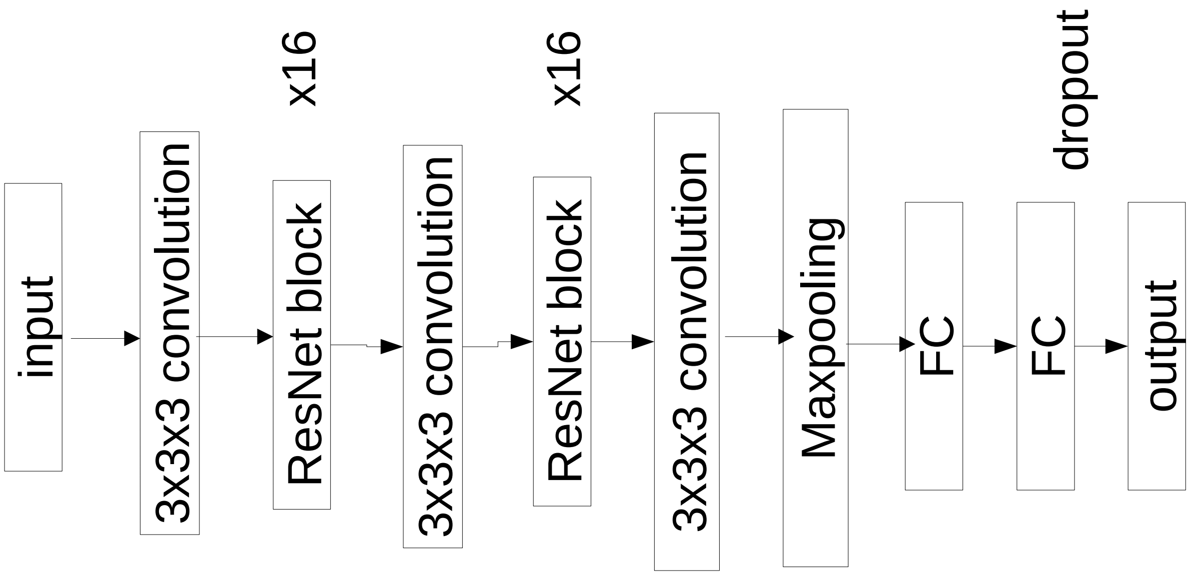
Feed Voxels directly to a Neural Network!

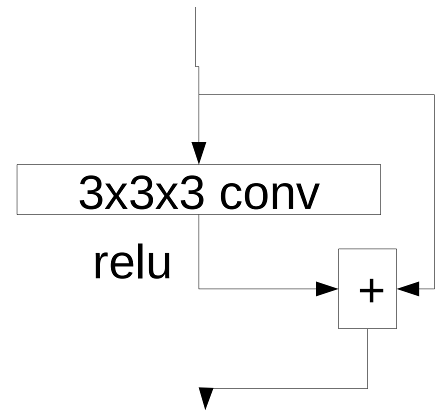


Original Image published in [LeCun et al., 1998]



DNN details





ResNet block

arXiv:1512.03385

We use random rotations and shifting on-the-fly to avoid overfitting, as well as well L1 weight regularization and dropout.
The network runs around 20 epochs.
Software : Keras with Tensorflow backend.

Dataset for training

Construction of MC training sample:

1) NEXUS - ‘true’ Geant4 hits

2) RECO – reconstructed hits

3) FRAMING and VOXELIZATION : voxels of dimension 5x5x5 mm^3

the center of the image is the barycenter of the event

4) Event energy normalized to 1 (the network has no information of the event energy)

5) Around 60 000 balanced signal/background events

Evaluation metrics

1. **AUC-ROC*** : Degree of distinguishability between classes – higher is better

True Negative rate $(\frac{\text{rejected background}}{\text{total background}})$

vs

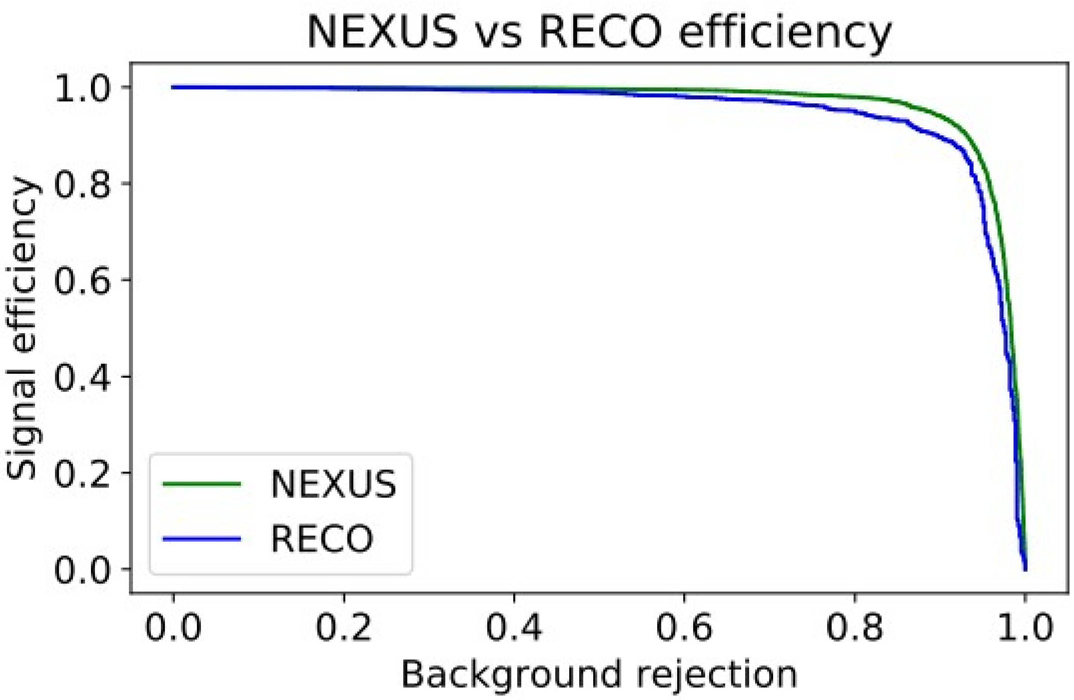
True Positive rate $(\frac{\text{accepted signal}}{\text{total signal}})$

2. **Figure of merit** : $\frac{\epsilon_{sig}}{\sqrt{\epsilon_{bck}}}$ – higher is better

The sensitivity to the half-life of the ββ0ν decay is proportional to f.o.m in background-limited experiments (arXiv:1010.5112v4)

RESULTS MC :
True labels

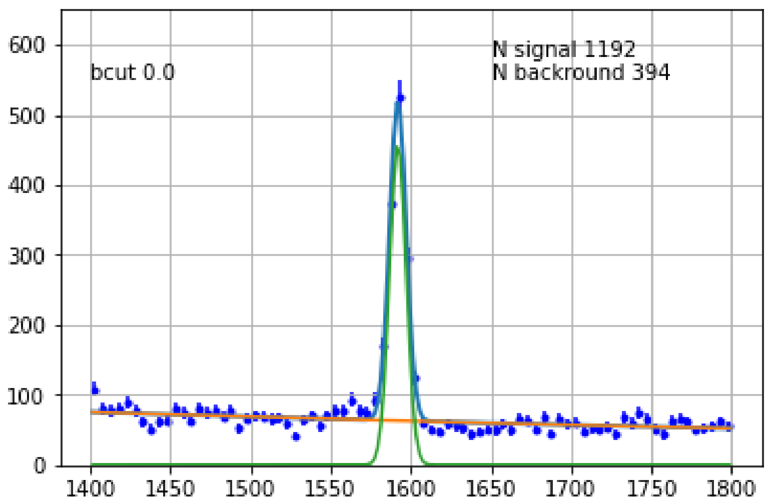
We use **true** nexus information to evaluate the performance of the network.

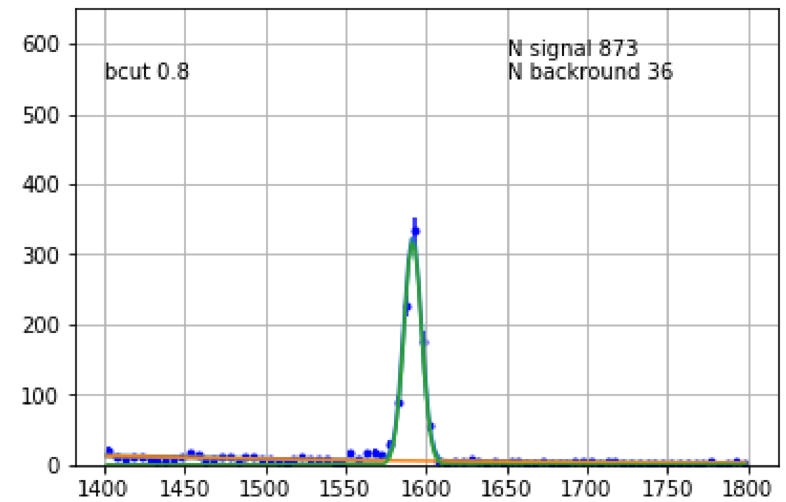


92% signal efficiency; 8% background acceptance

90% signal efficiency; 10% background acceptance

RESULTS MC:
Gaussian signal + exponential background model

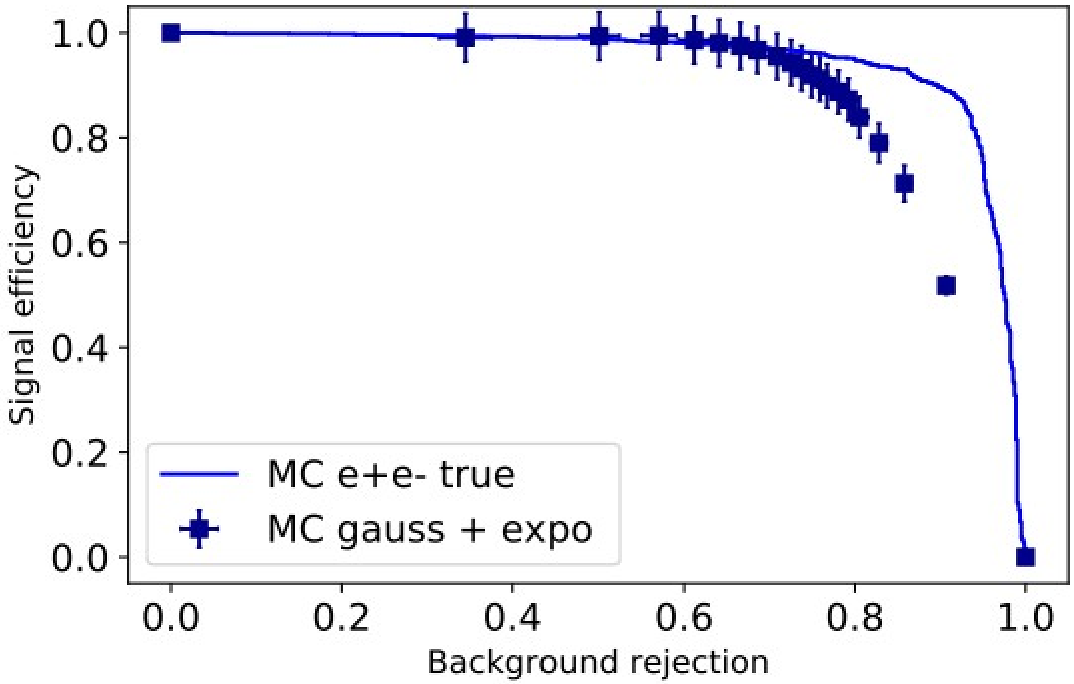




1. Fit the histogram to gaussian (signal) and exponential (background)
2. Integrate to calculate total number of signal and background $N^0_{bck}, \quad N^0_{sig}$
3. Apply i^{th} cut on DNN prediction and calculate $N^i_{bck}, \quad N^i_{sig}$

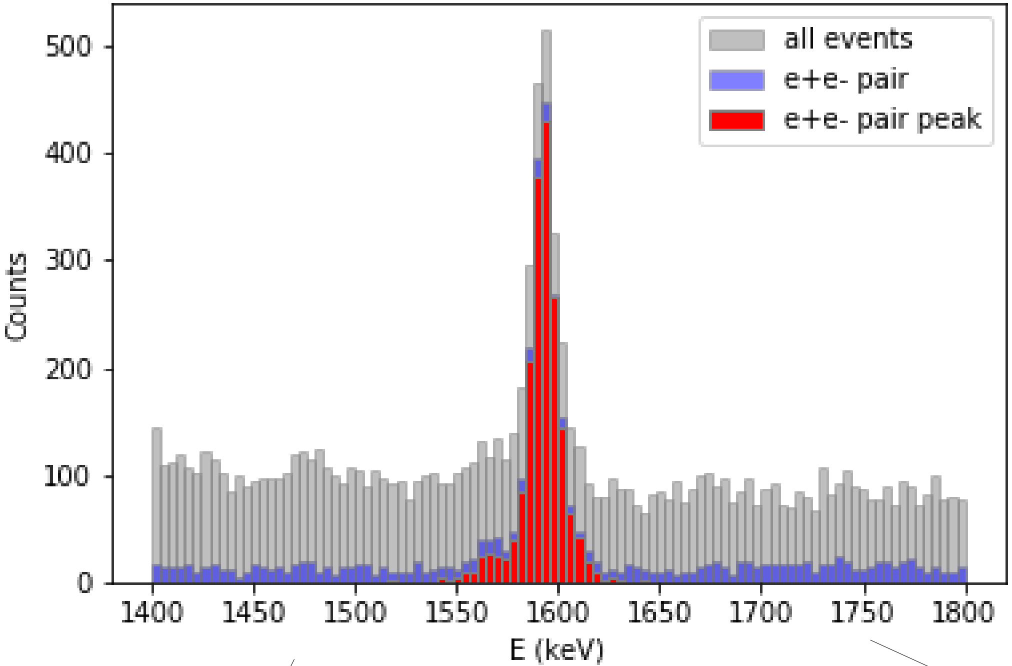
$$\epsilon^i_{sig} = \frac{N^i_{sig}}{N^0_{sig}} \qquad \epsilon^i_{bck} = \frac{N^i_{bck}}{N^0_{bck}} \qquad f.o.m = \frac{\epsilon^i_{sig}}{\sqrt{\epsilon^i_{bck}}}$$

RESULTS MC:
Gaussian signal + exponential background model



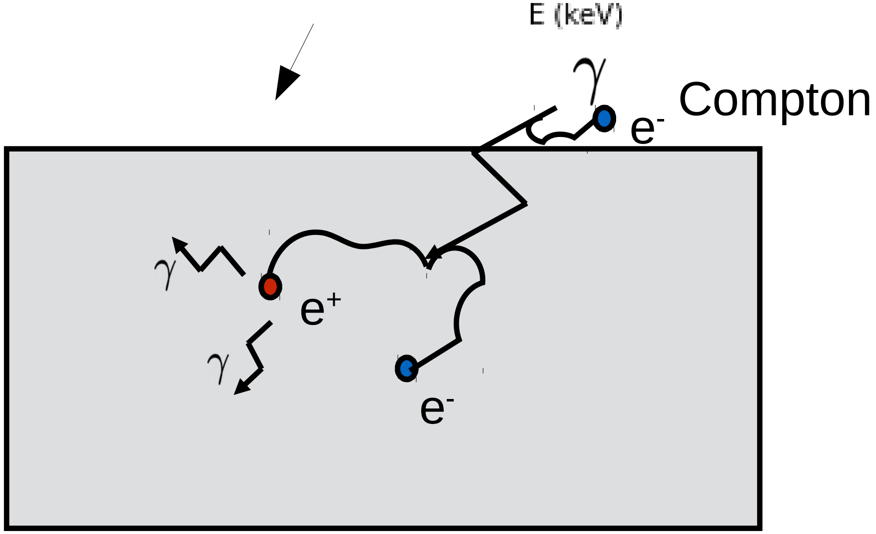
Not quite matching…

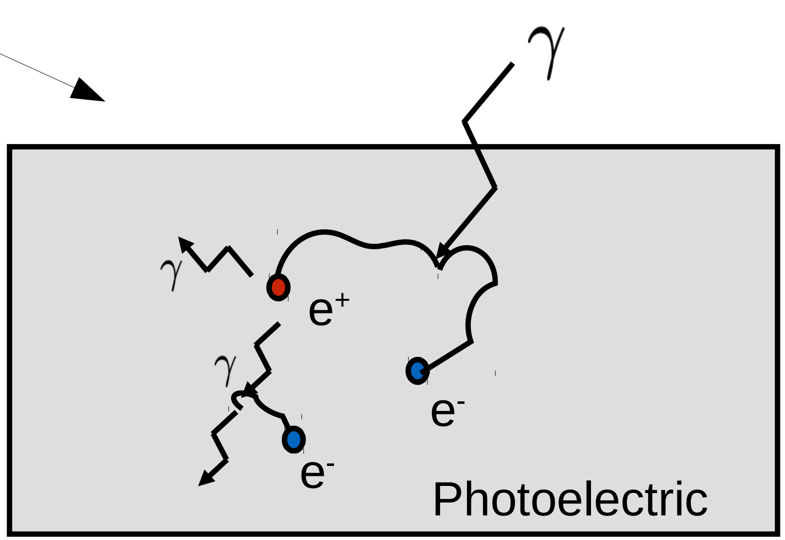
RESULTS MC:
Gaussian signal + exponential background model



Solution:
Training a network where signal is a positron-electron pair with the energy ~ 1.59 MeV

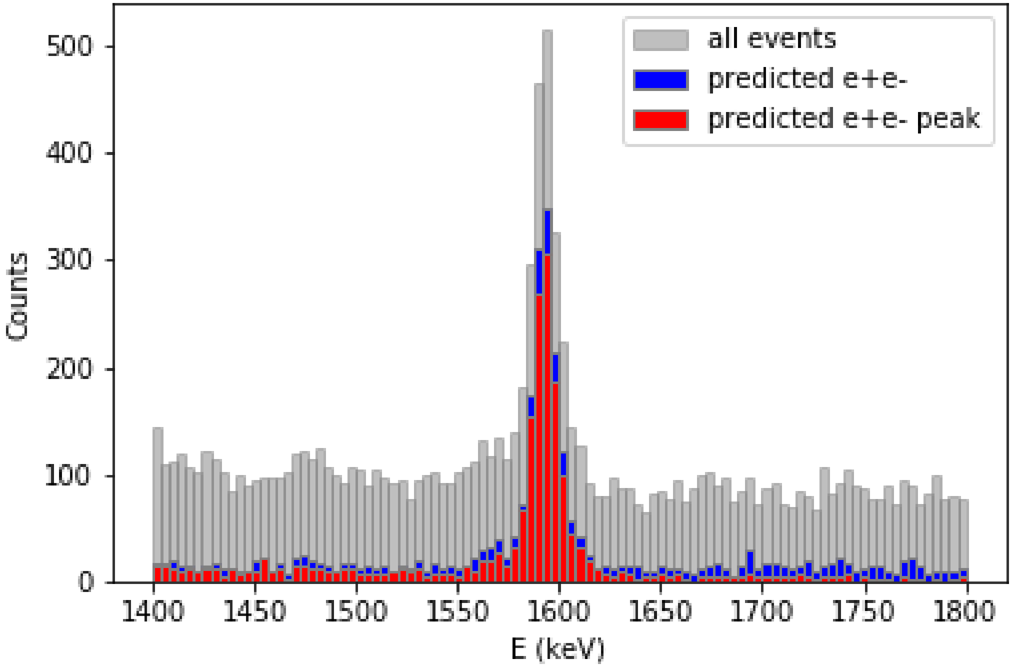
label RECO peak





RESULTS MC: Gaussian signal + exponential background model

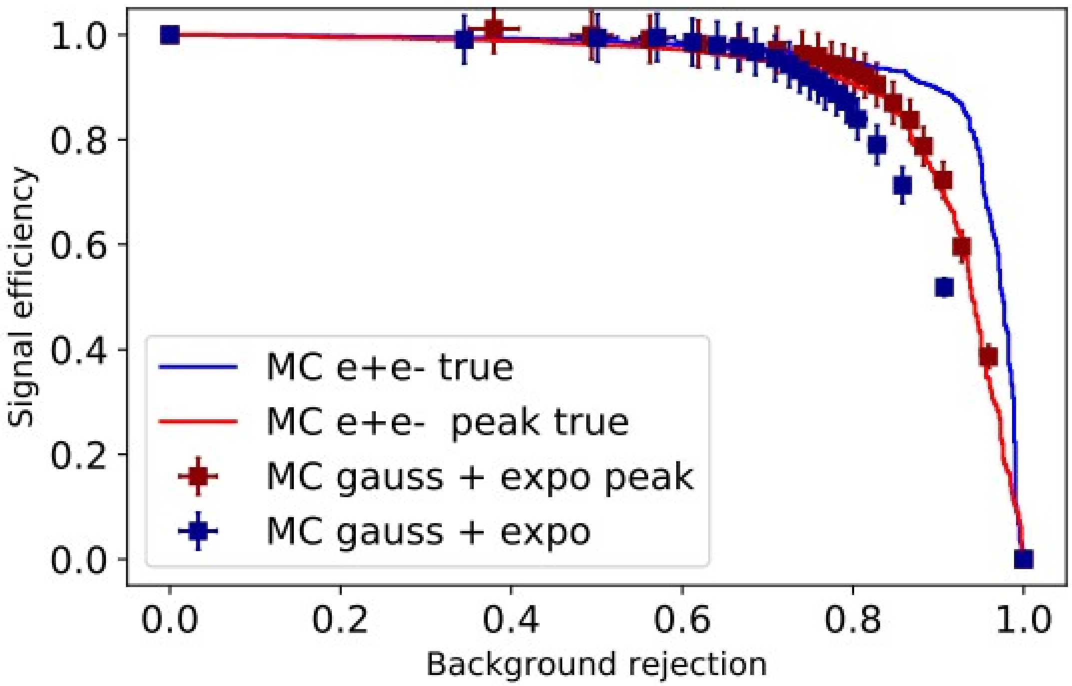
Distribution of predicted signal (threshold cut 0.8)



Solution:
Training a network where signal is a positron-electron pair with the energy ~ 1.59 MeV

The network cut out some of the higher energy events; probably similar to one track cut of classical analysis.

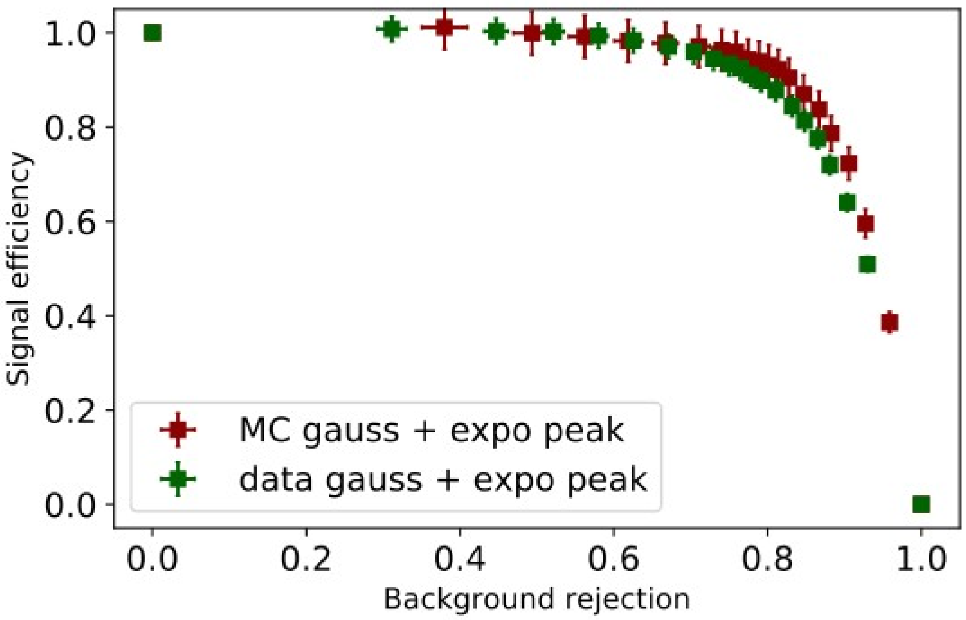
RESULTS MC:
Gaussian signal + exponential background model

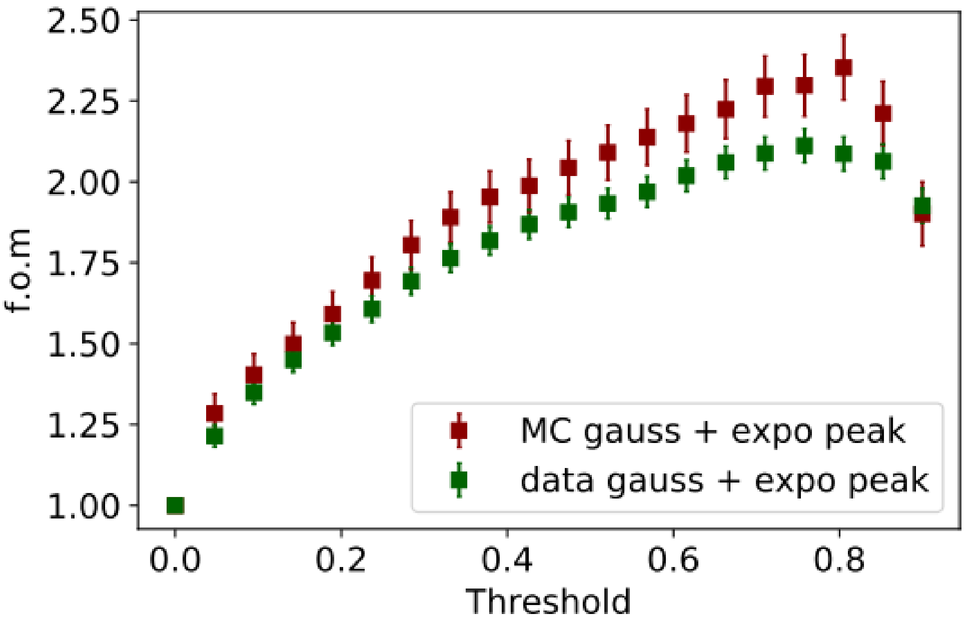


85% signal efficiency;
15% background acceptance

READY FOR THE DATA

RESULTS DATA vs MC



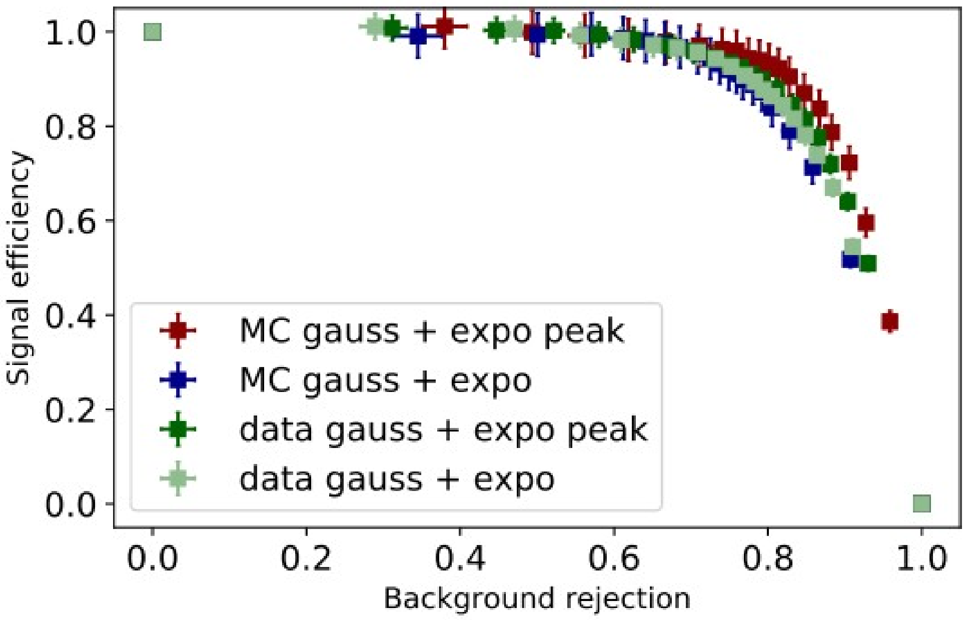


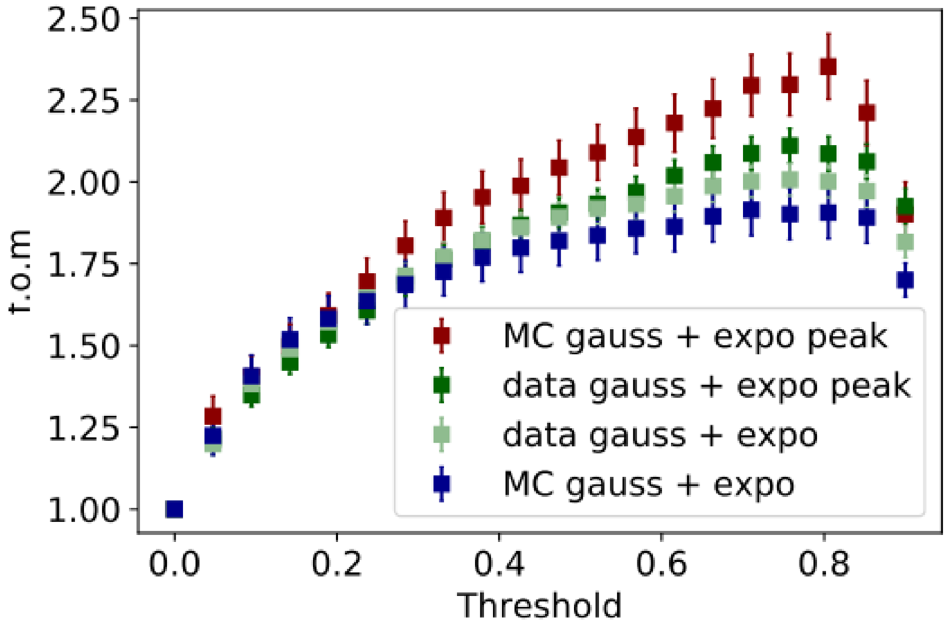
85% signal efficiency; 15% background acceptance

84% signal efficiency; 16% background acceptance

RESULTS DATA vs MC

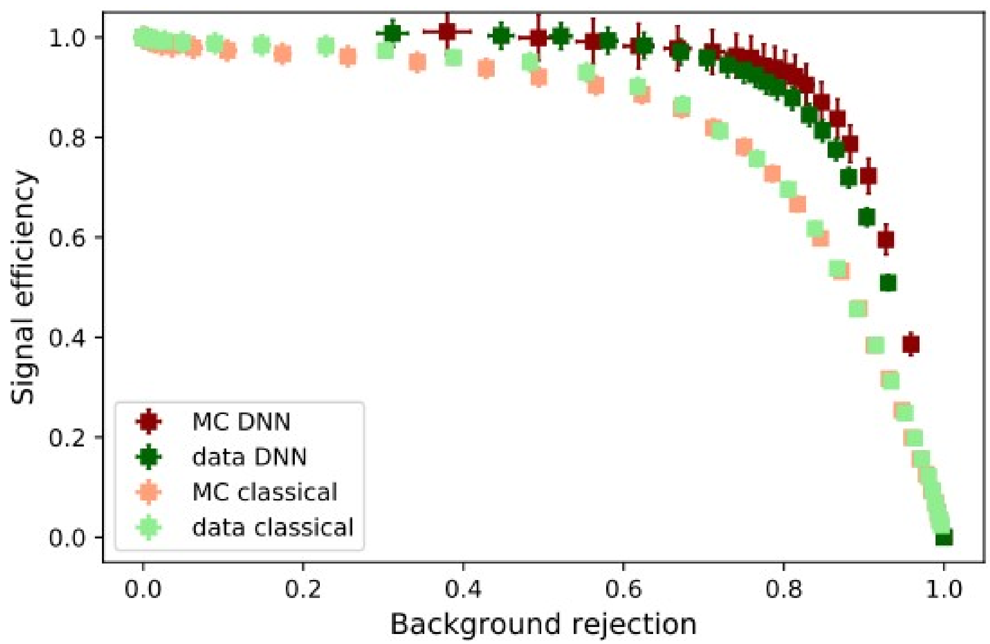
Different networks comparison

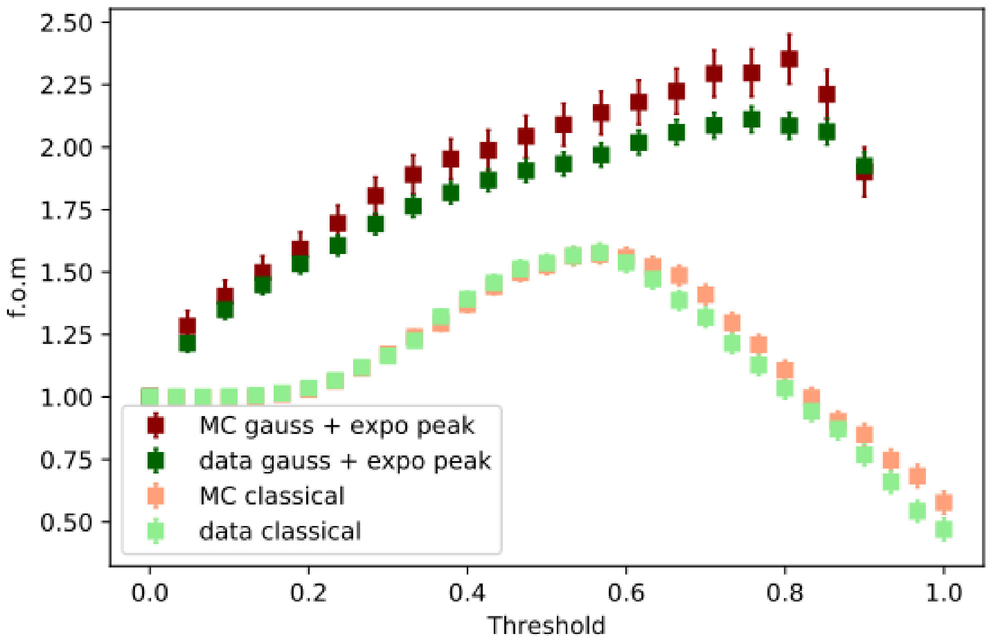




Data only slightly improved with different labeling...

RESULTS DNN vs Classical





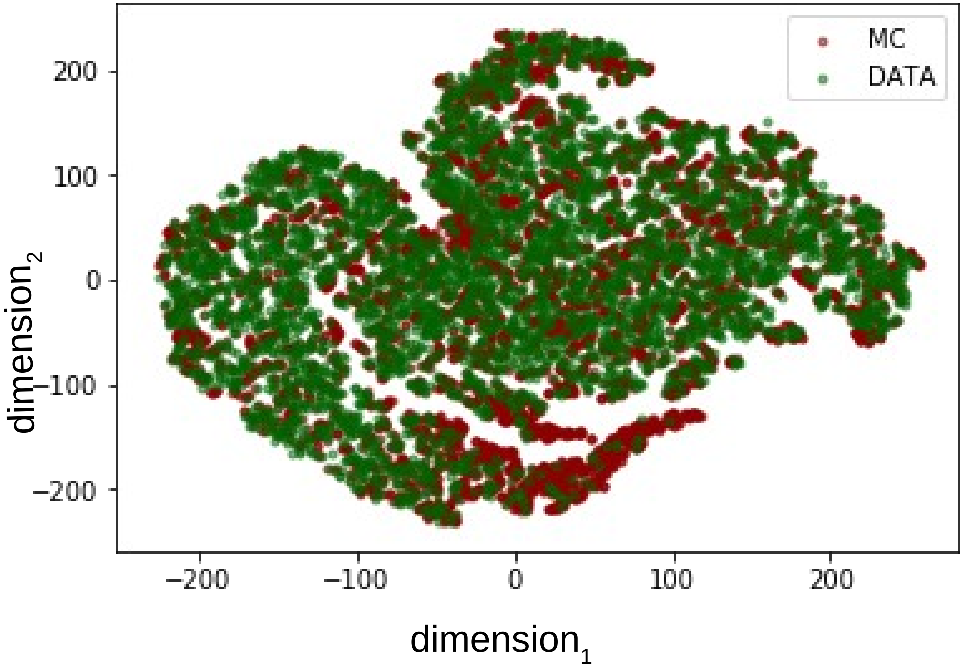
arXiv:1905.13141 (2019)

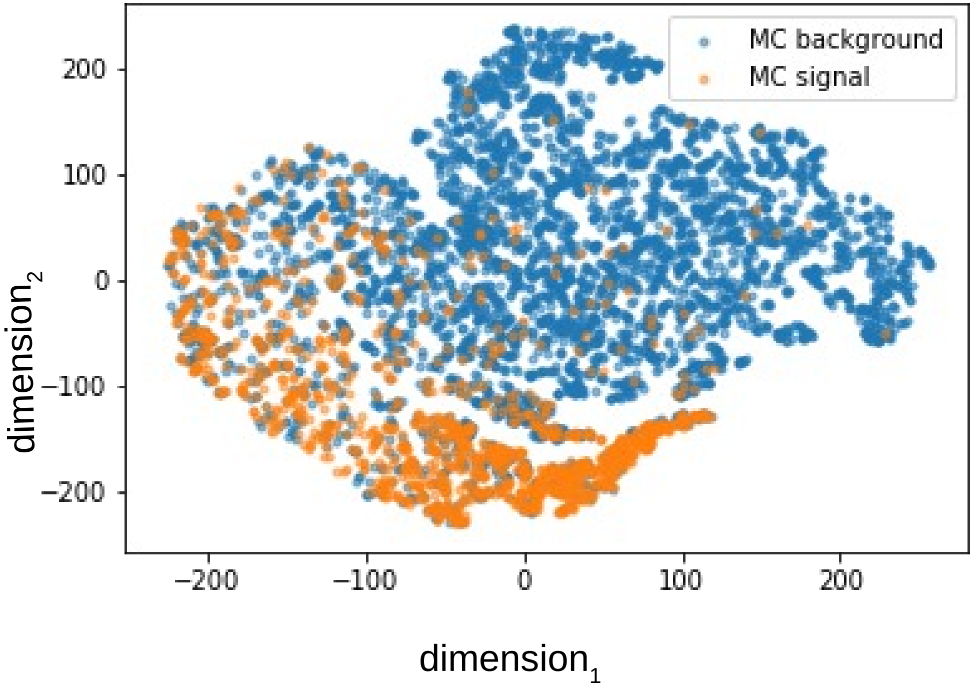
Significant improvement!

Just for fun: t-SNE

t-Distributed Stochastic Neighbor Embedding (t-SNE) is a non-linear technique for dimensionality reduction, it preserves local distances.

Journal of Machine Learning Research 9 (2008) 2579-2605





Next steps

- Use similar analysis on NEXT-100 simmulations.
- Try SparseConvNets :

  PyTorch SparseConvNet library by Facebook: arXiv:1706.01307v1
- Try DNN for reconstruction?

- Look for double escape peak close to $Q_{\beta\beta}$ peak
- Understand differences between MC and data!

Next steps

- Use similar analysis on NEXT-100 simmulations.
- Try SparseConvNets :

  PyTorch SparseConvNet library by Facebook: arXiv:1706.01307v1
- Try DNN for reconstruction?

- Look for double escape peak close to $Q_{\beta\beta}$ peak
- Understand differences between MC and data!

Thank you!