AI/ML in medical physics

B. Wynne

10/04/25

Disclaimer

Almost all of what follows is wishful thinking: descriptions of problems or example applications

Obviously medical physics is a huge area, and AI/ML development is moving very rapidly here

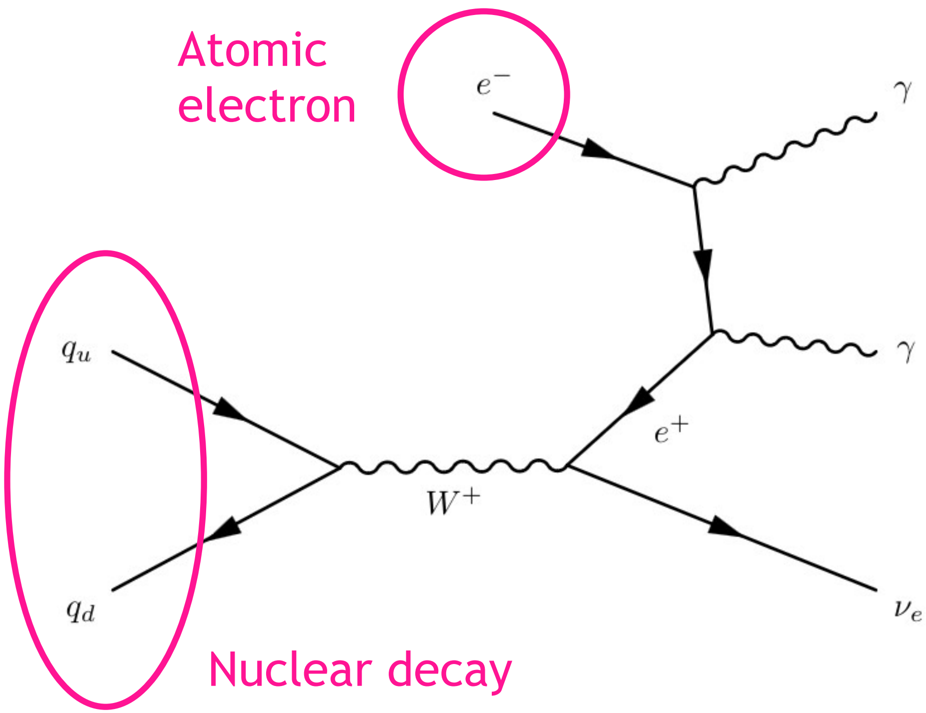
I can’t hope to be comprehensive, or guarantee to be up-to-date

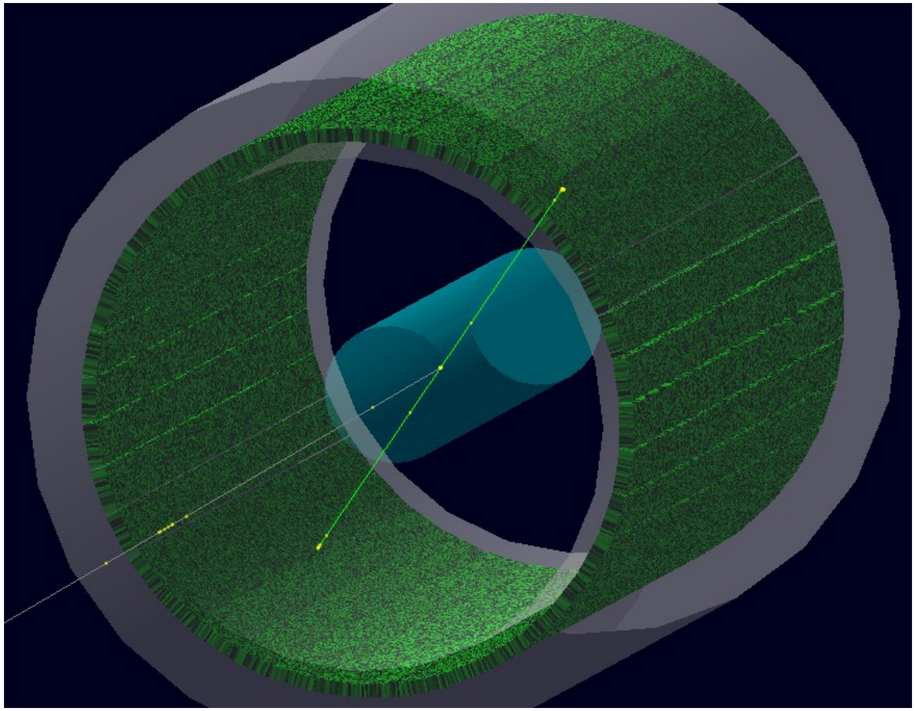
PET scan introduction

Our work here in IPNP focusses on PET scanners, and I'm going to spend some time introducing the topic for those who are not familiar

A radioactive isotope (e.g. Fluorine-18 or Gallium-68) is used to tag a biologically active molecule which will move through the body of a patient

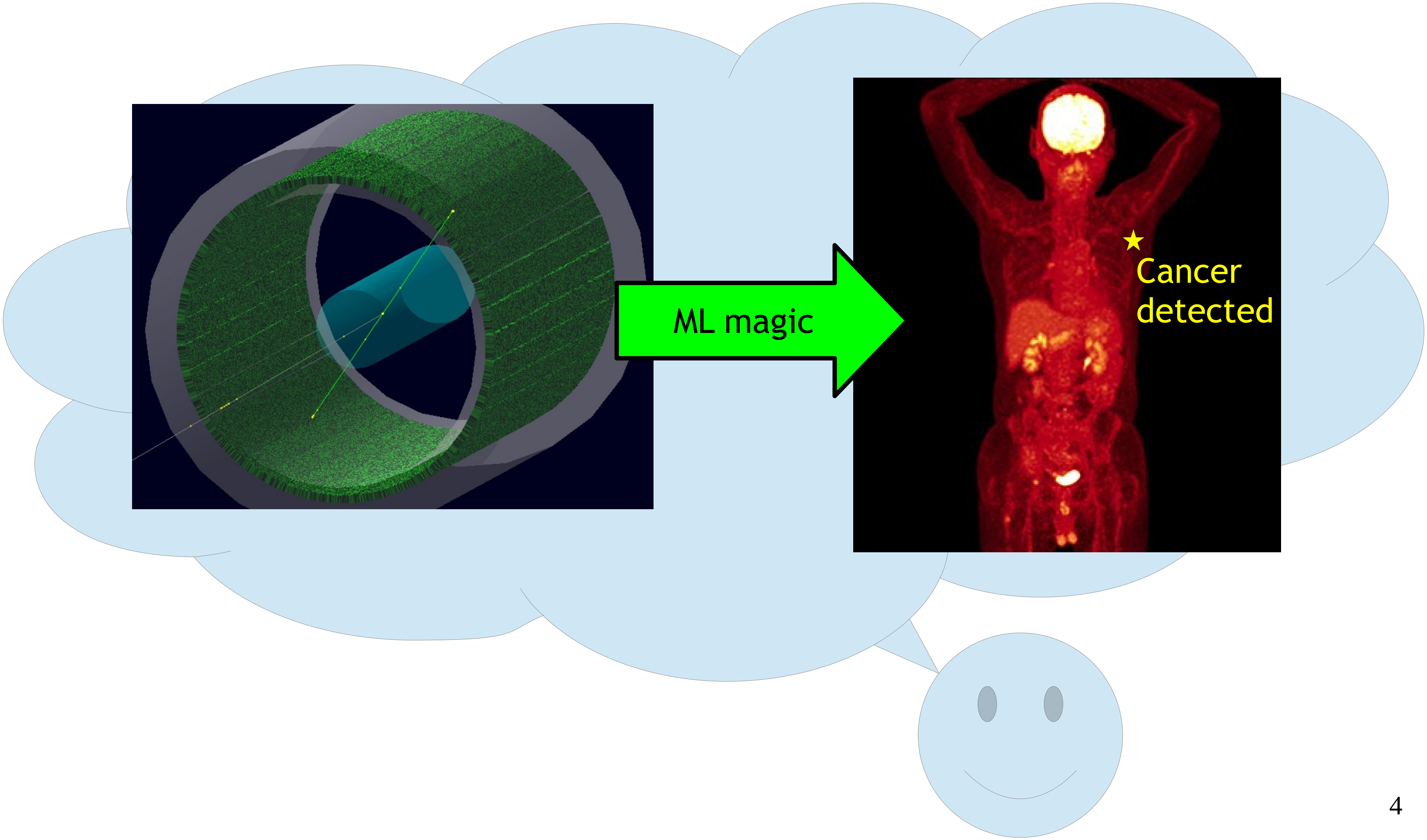
Positrons are emitted, and located by the pairs of 511keV photons produced when they annihilate





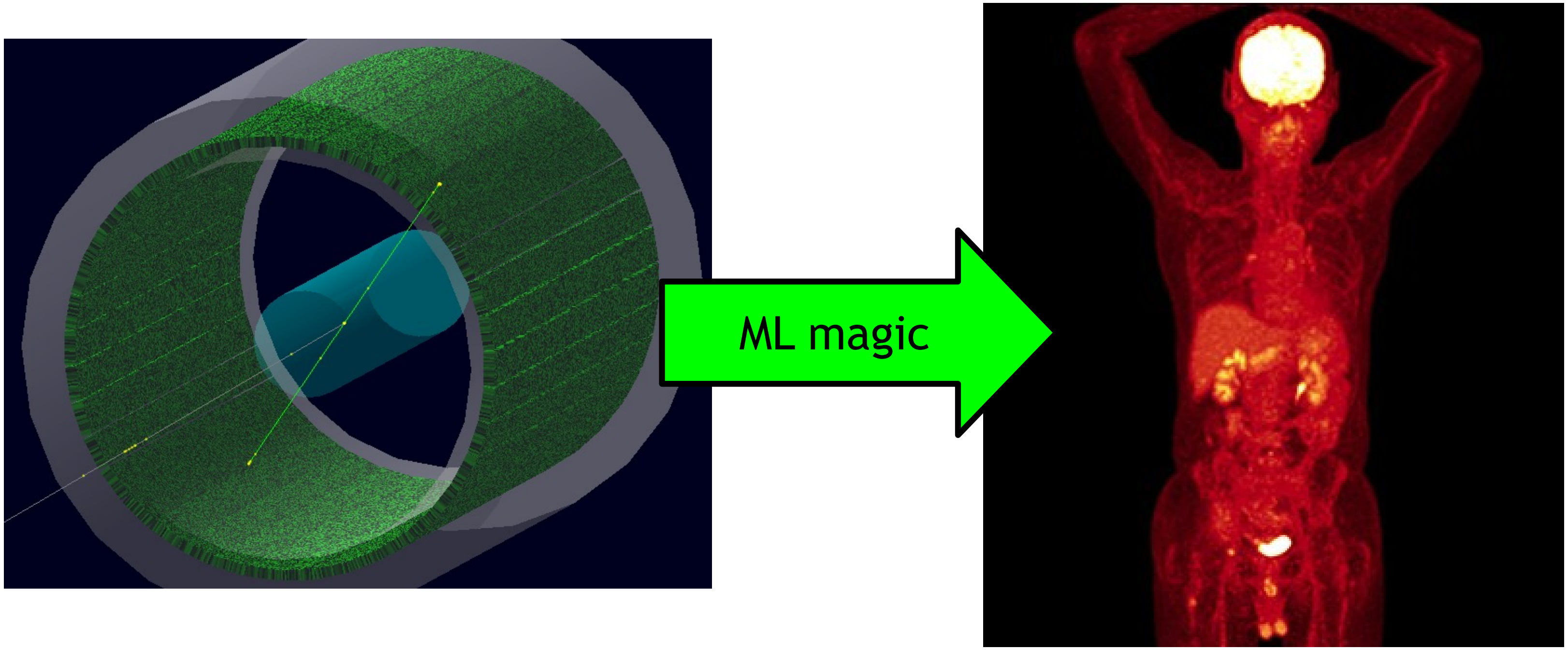
Caveat: this is applied science

Ultimately you have to try and produce something that will be useful in a clinical setting, rather than just tinkering with things that interest you personally



PET image reconstruction

"PET image reconstruction using ML is a solved problem"
-- paraphrasing someone selling PET scanners and their software



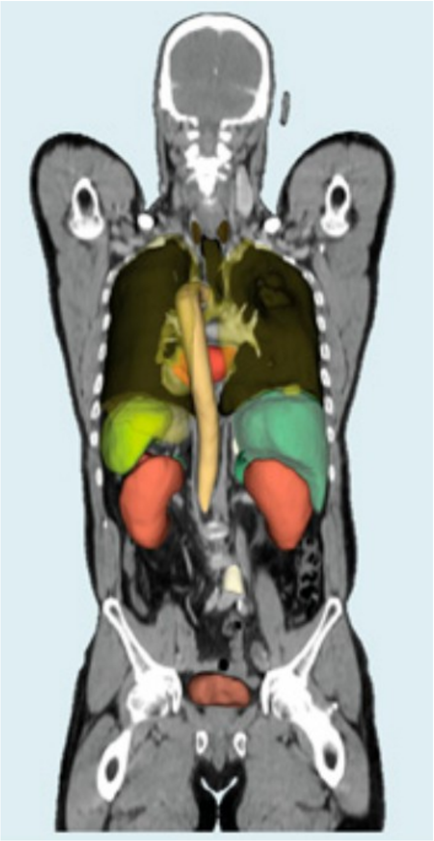
"Clinicians will never trust ML reconstruction of PET images"
-- paraphrasing someone who works with clinicians

Either way, it's not great news for researchers

PET scan directions

So, where should we apply ML in this field?

Segmentation: it works for 2D images, and it can work for 3D medical data



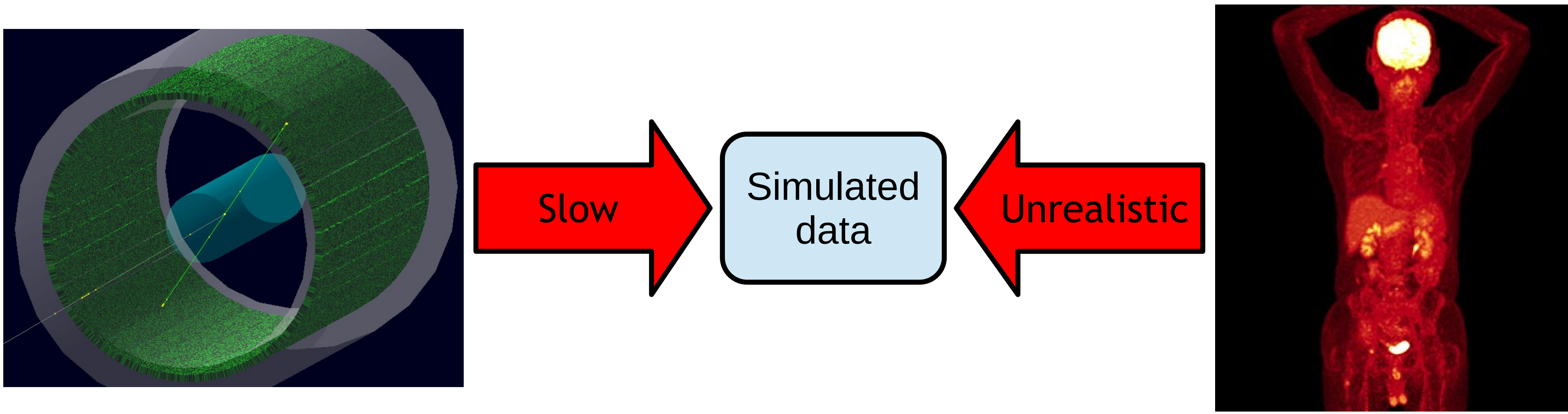
Another "solved problem" though, and it's pretty-much a function of training data in → better results out (i.e. it relies on 1000s of hand-labelled datasets)

PET simulation

There are two existing simulation techniques

- Using Geant4 to provide full-physics behaviour: slow
- Simple inversion of target image: poor approximation of detector effects

Additional problem that PET datasets are very large, so it would be nice to generate them at point of use

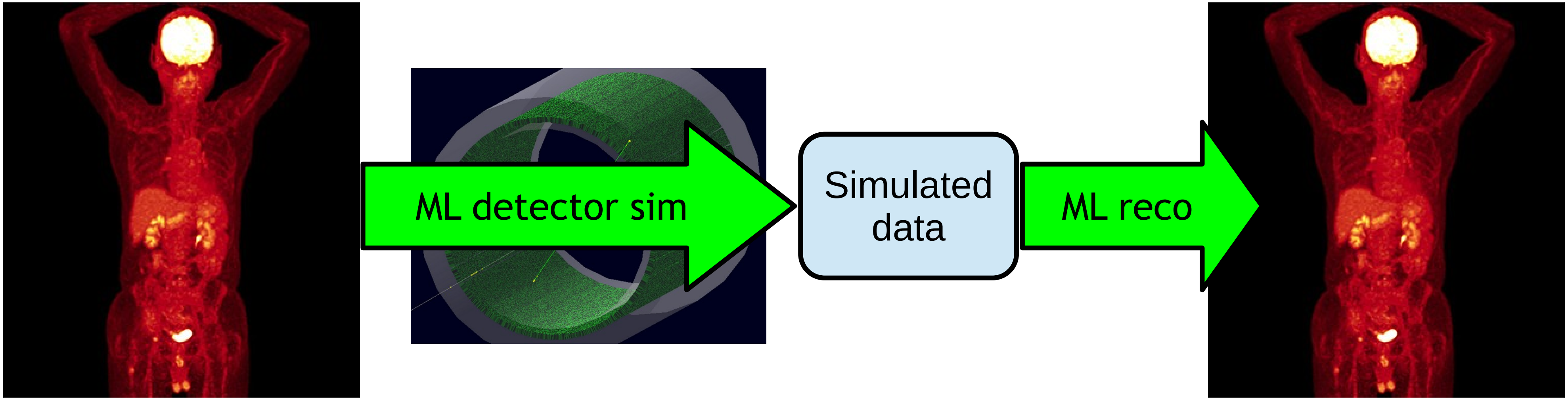


Generative ML models could potentially be faster than Geant4, and be adapted to arbitrary tasks without a lot of detailed 3D geometry building

- Most GenAI is some kind of interpolation between existing datapoints, e.g. abstract voxels

PET simulation

In an ideal world we could provide our target image as input to an ML model for the detector effects



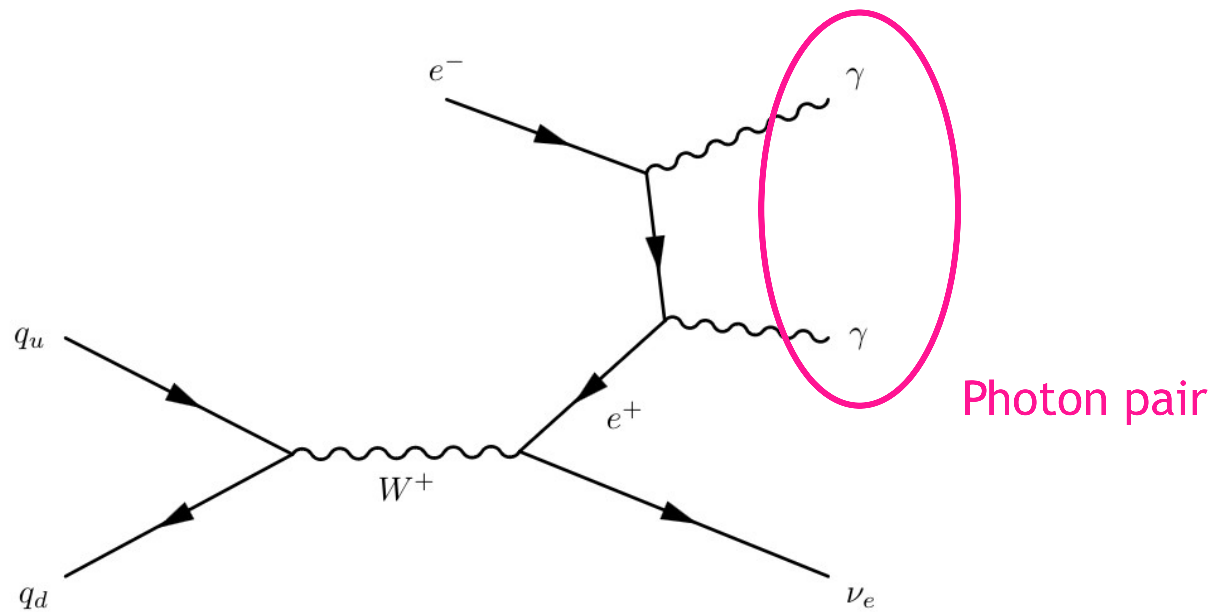
We can use a fast ML-based reconstruction to provide scoring for the training loop: doesn’t need to be trustworthy enough for clinical use

Similar in concept to an autoencoder, although the “latent space” would instead be our simulated data: large and structured

PET coincidence selection

Identifing photons from the same decay

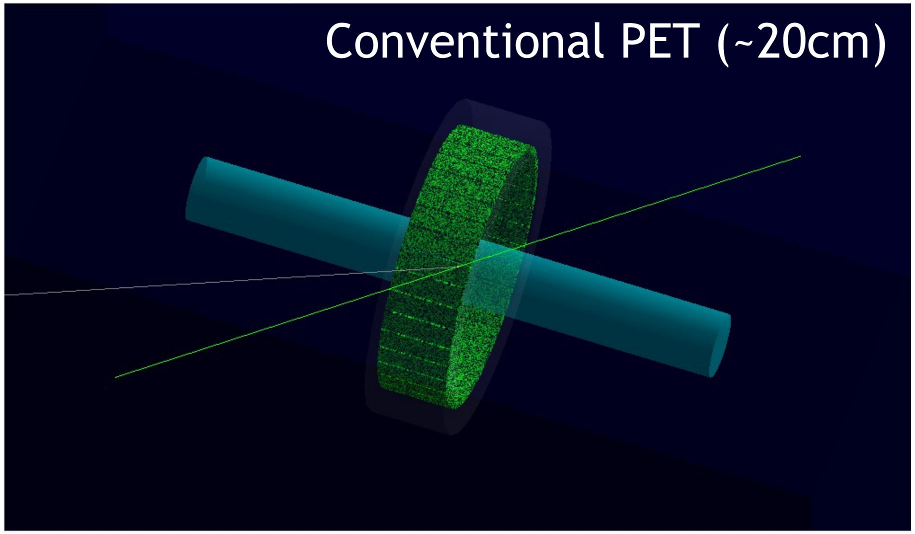
Ideally each positron emission can be identified by a simultaneous, back-to-back pair of 511keV photons when it annihilates

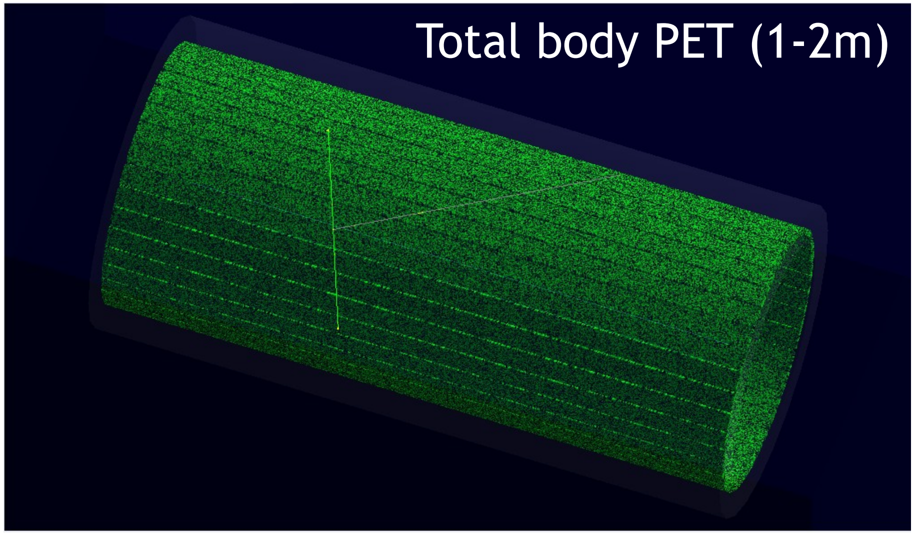


In reality we will be affected by scattering and attenuation within the patient (or the bed), by the flight time of the photons to the detector, and then by detector effects

PET coincidence selection

With the new generation of “total body” PET scanners, far greater acceptance

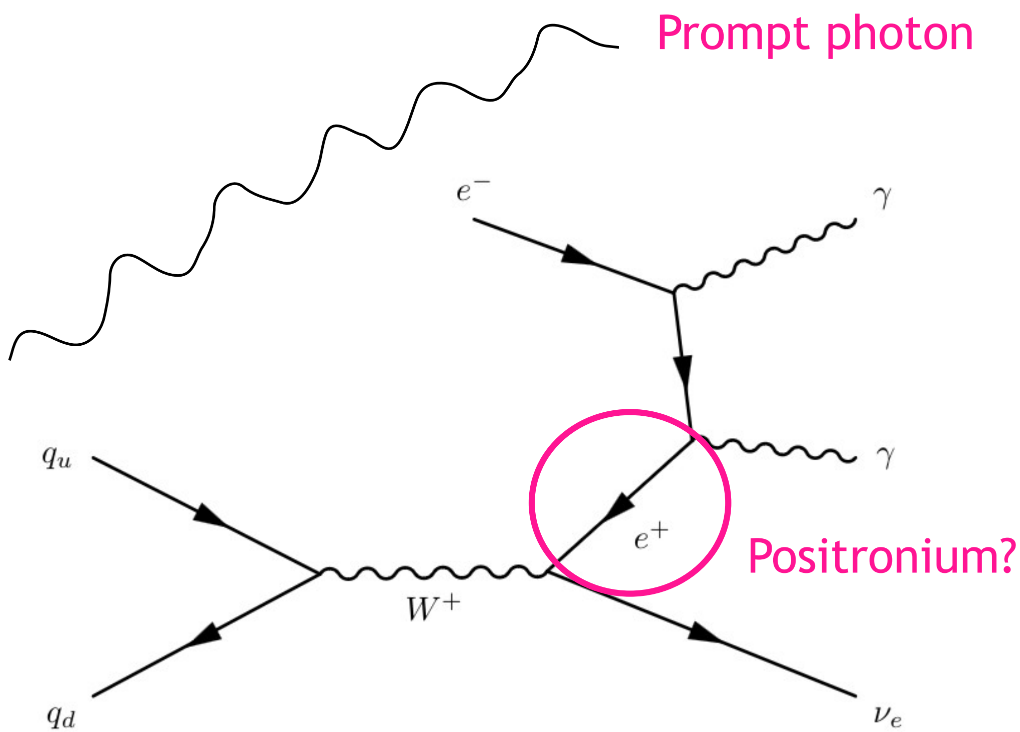




- Increased chance of pile-up from multiple decays (>2 photons within coincidence window)
- Oblique photons more susceptible to scattering/attenuation
- Greater time-of-flight difference possible for oblique photons (longer coincidence window needed)

PET coincidence selection

We can (and do!) complicate the picture still further by using tracer isotopes that also emit a prompt photon



If we can associate this third gamma with the pair from the positron annihilation, useful extra information is available

- Simultaneous imaging of multiple tracers (discriminate 2 or 3-photon isotopes)
- Positronium formation may lead to delayed emission of the photon pair relative to prompt – and the decay time depends on the chemical environment

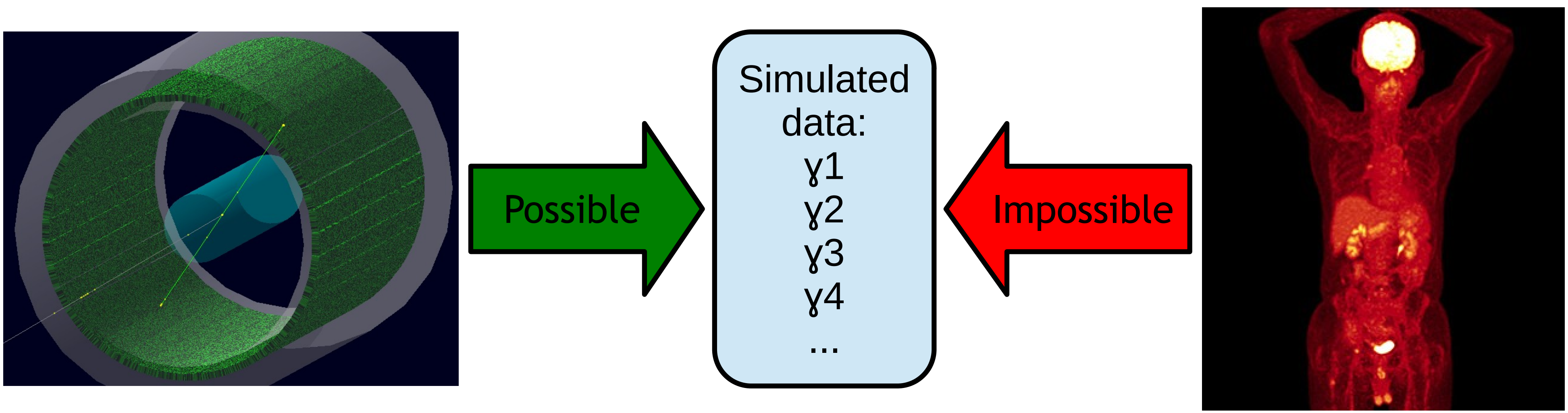
PET coincidence selection

All this is to say that there is far more complexity to raw PET data than just combining photons as they arrive 2-by-2

- Multiple photons, either from overlapping decays or additional prompt decays (or radioactive background)
- Energy spectrum that includes attenuation and detector resolution
- Time range arising from time-of-flight and delayed positronium decays

ML methods have the potential to analyse these correlated effects

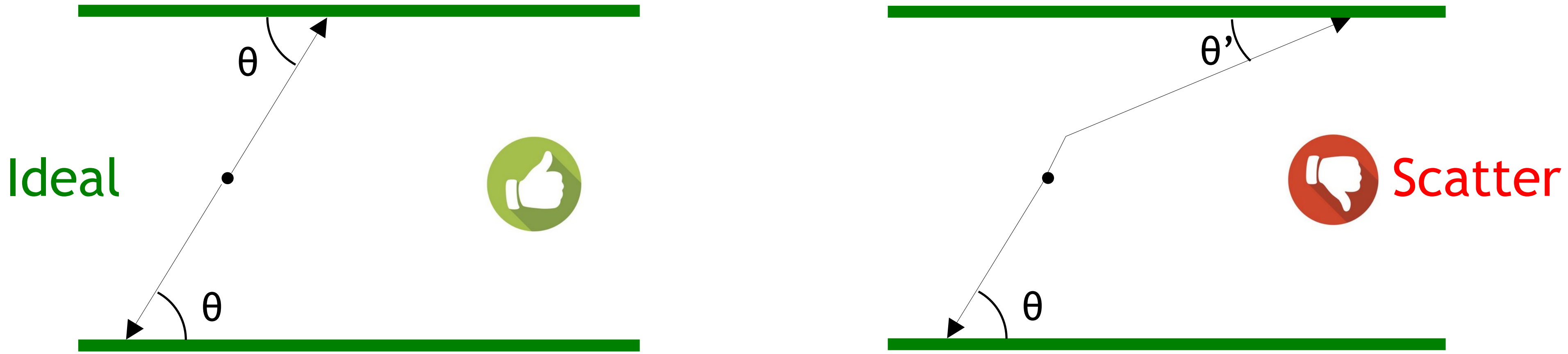
Bonus: extra motivation for (fast) detector simulation, since per-photon raw data cannot be derived straight from a target image (can only produce histograms)



Enhanced readout

PET detectors couple an array of scintillator crystals (e.g. 5x5) with an array of SiPMs (e.g. 4x4), giving a readout of energy deposited, and which crystal it was deposited in, at which time. Can we do better?

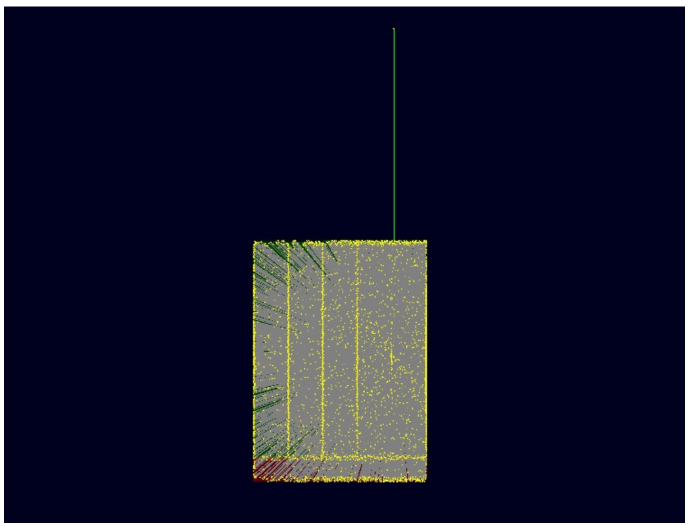
- Reconstructing the incident angle of the photons would allow background suppression

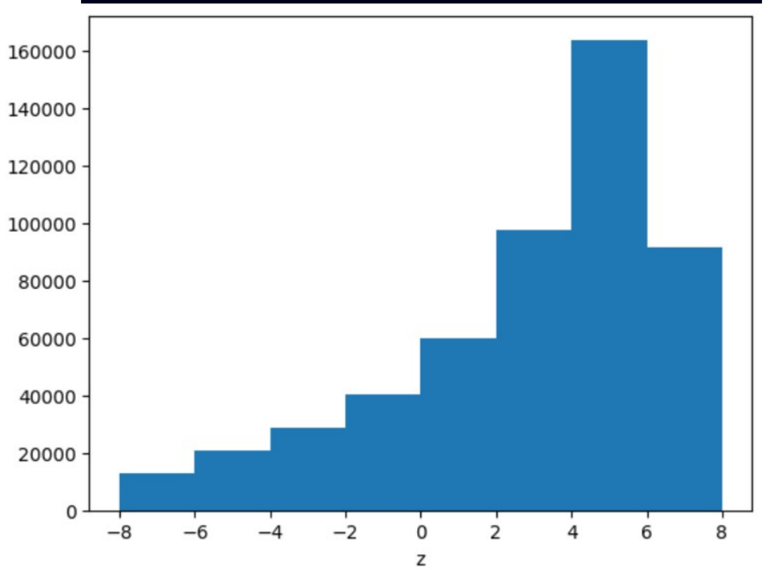


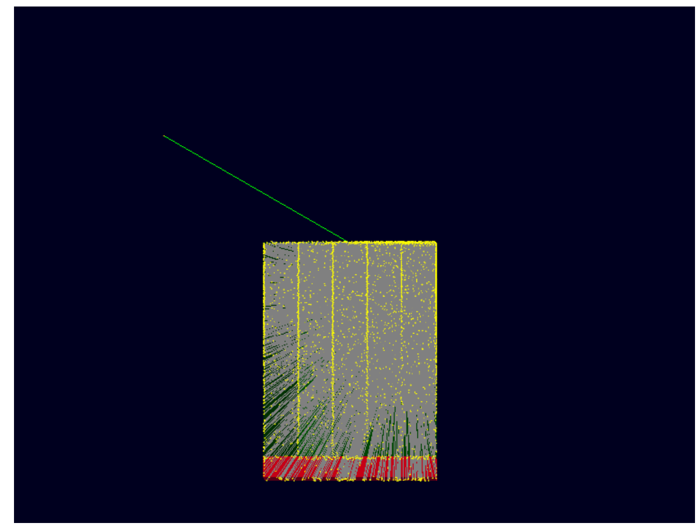
- It would also be a potential tool for associating prompt photons with the corresponding positron-annihilation photon pair

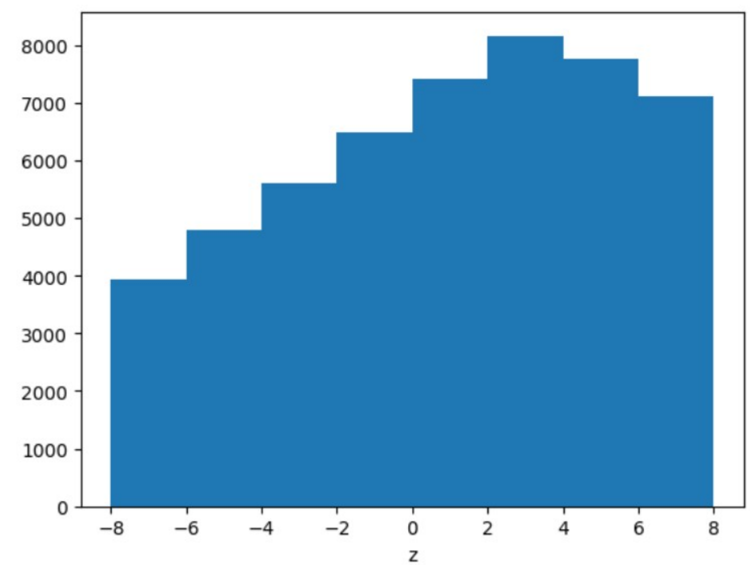
Enhanced readout

Simulated results suggest this is possible, although note the correlation with the total number of detected photons (and thus the energy measurement)









Enhanced readout

We expect the polarisation of the photons from the positron annihilation to be correlated

- Photon polarisation reconstruction would provide another tool for coincidence selection

The POLAR experiment (GRB satellite) does this with similar detectors to PET, in a similar energy range
sciencedirect.com S0168900217310239
sciencedirect.com S0168900205012891

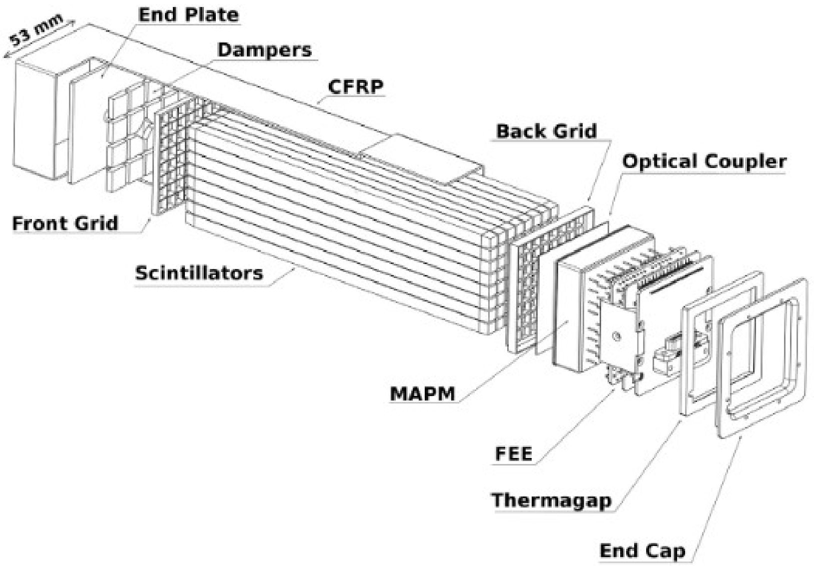


Fig. 2. Exploding view of one module.

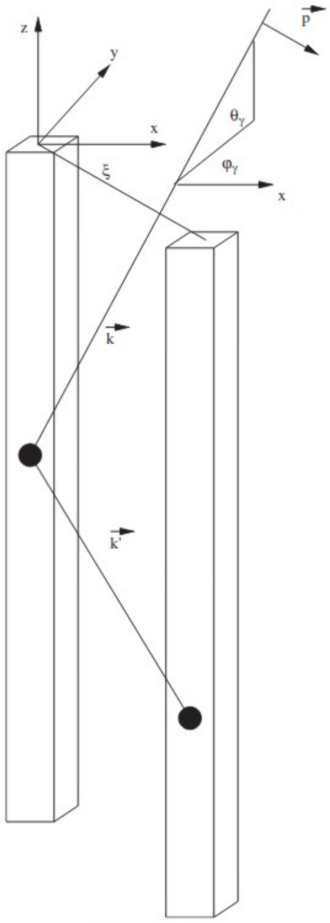


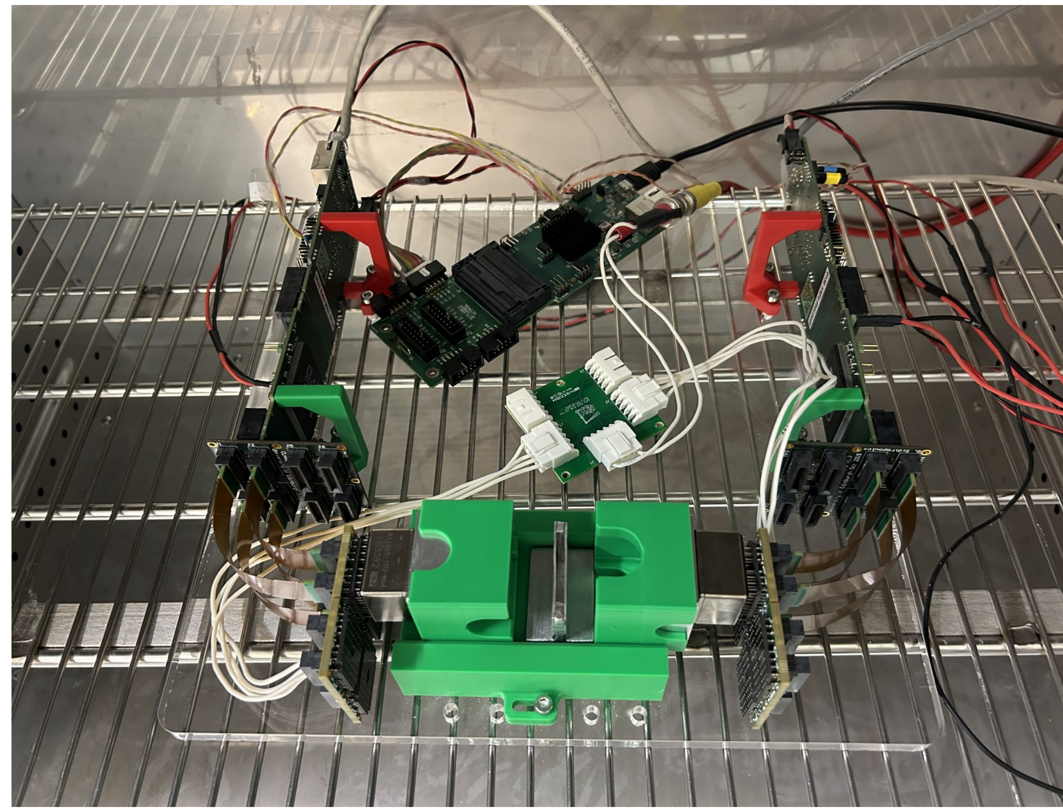
Fig. 1. Geometry of the large angle Compton scattering. The two bars where interaction occur are shown. θ_γ and ϕ_γ are the entrance angle of the photon relative to a detector fixed coordinate system. ξ is the measured azimuthal direction that correlate with polarization direction $\vec{p}$.

Enhanced readout

From glancing at the angular reconstruction example, you might imagine that these examples would only require fairly trivial ML (if it is needed at all)
- good!

Standard practise to use FPGAs to read out photodetectors with low (fixed) latency, and intention is to implement this in the FPGA firmware

hls4ml



Existing project motivated by HEP experiments to convert ML models to firmware

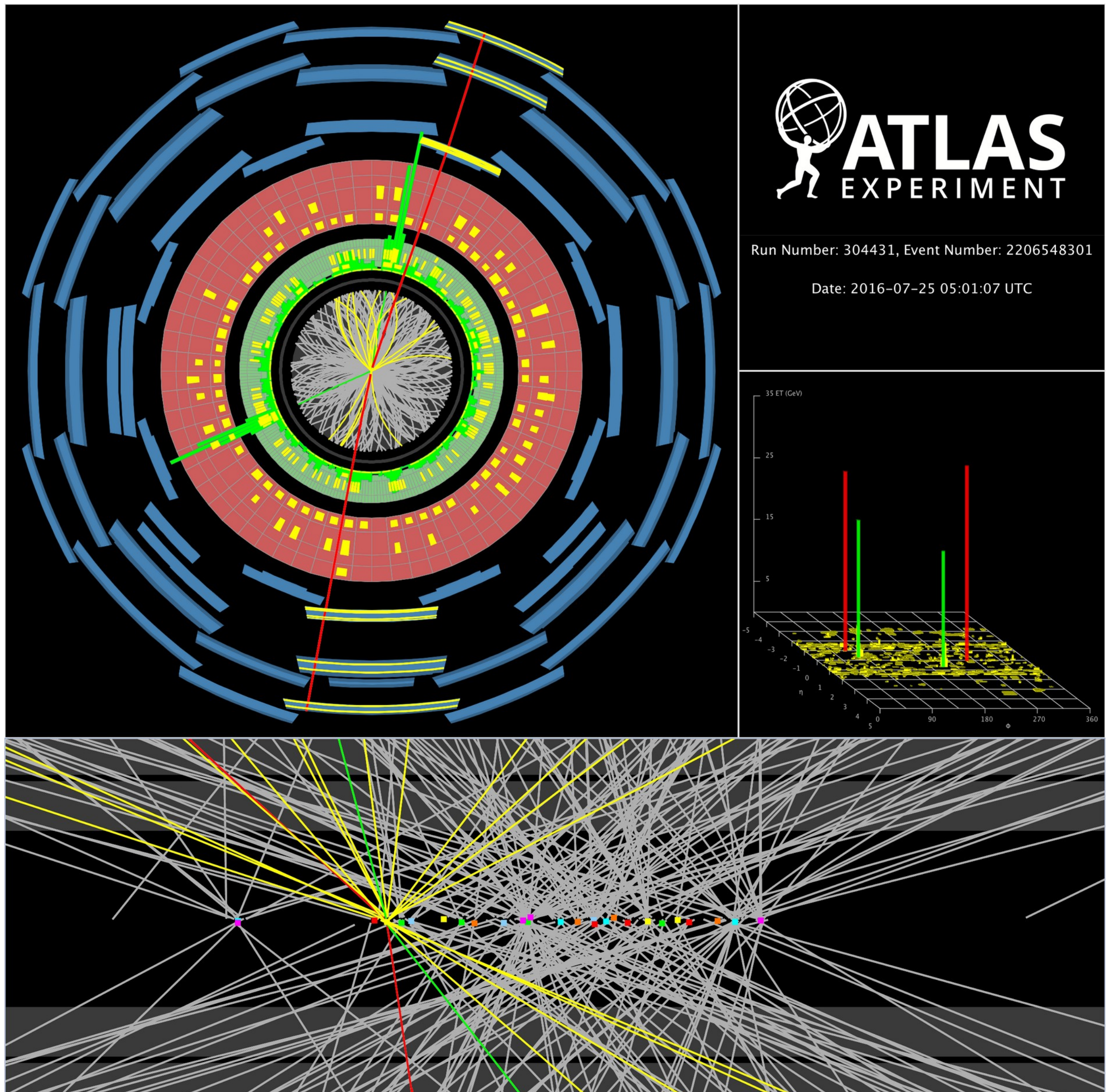
Quantisation in ML models a great tool to keep readout and firmware simple

- A 2-bit angular reconstruction can still provide ~50% background suppression

A brief aside to talk about something real

ATLAS upgrade tracking

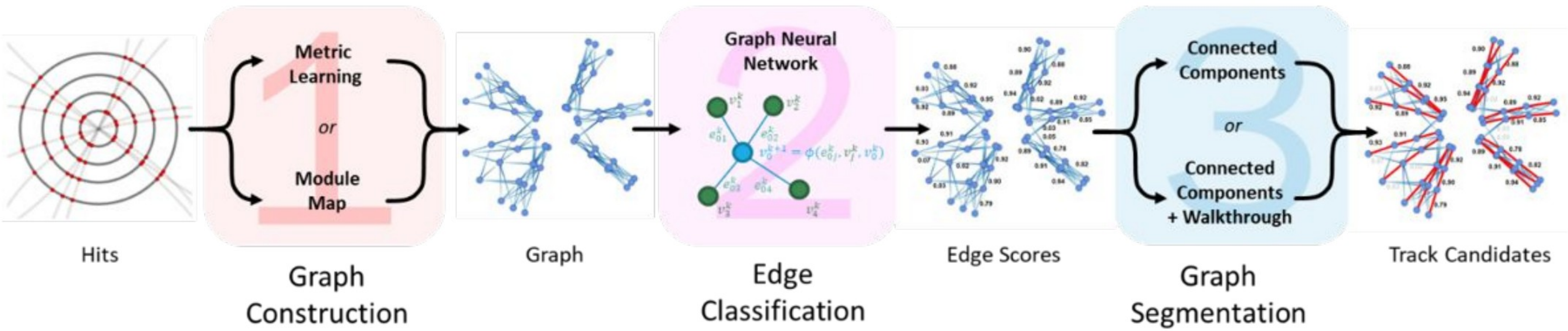
The ATLAS detector reconstructs the tracks of charged particles leaving the centre of the detector, by connecting the hits on layers of silicon sensors



ATLAS upgrade tracking

With the upgraded LHC comes a far larger number of particles in the detector at once, and a huge increase in the computational complexity of finding tracks

One attempt at addressing this has been to use a GNN, with nodes of the graph formed by silicon hits and edges being potential tracks



Graphs can contain 1-4 million edges, which makes some too large to fit in GPU memory

Simple but effective workaround: edge preparation, message-passing, and edge updates are performed for fixed-size (100k) subsets of edges

Memory use reduced, wall-clock time reduced, inference performance unaffected

Summary

There's a huge amount of work in the medical physics space turning very large hand-annotated datasets into very large ML models that may or may not be used to replace existing techniques

- As a physicist I don't want to try and tell clinicians how to do their job (or that we will try and replace it with ML)

Arguably more interesting to try and enable new analyses or new detector outputs

PET scanners are detectors that produce large amounts of raw photon data

- Can we make smarter selections of the photons to use for image reconstruction?
- Can we improve the information about each photon?
- Can we simulate this data efficiently, to test new methods?

In almost entirely unrelated news, the ATLAS track reconstruction upgrade may make use of GNNs in future

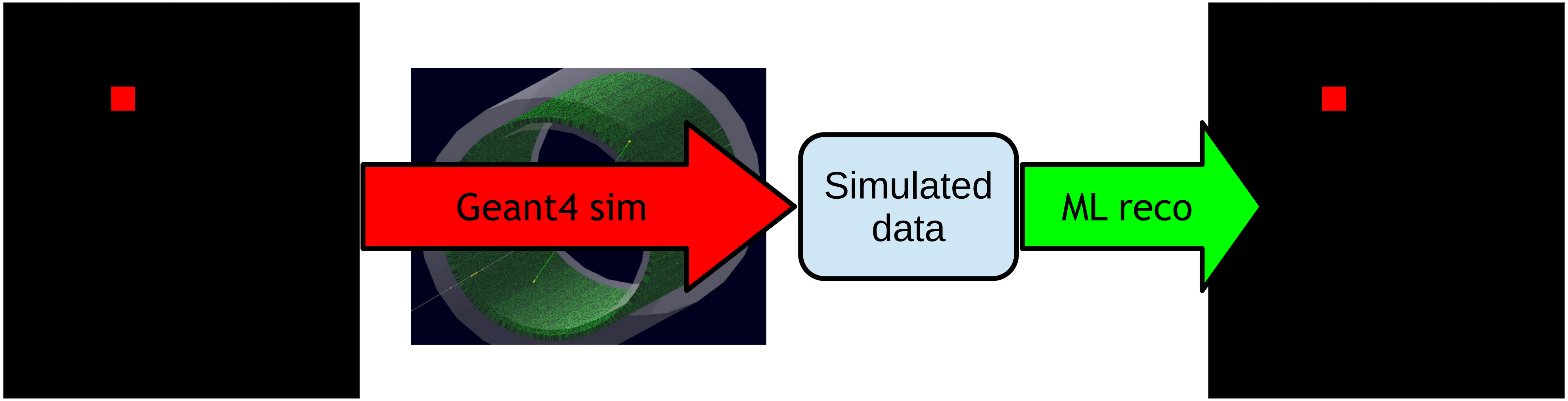
BACKUP

PET scan directions

So, where should we apply ML in this field?

Simulation: generative models for fast simulation

Start by using full Geant4 simulation of a sample of voxels throughout the target image space to train the reconstruction



Once the ML reconstruction algorithm can reliably reproduce fully-simulated voxels, can train ML fast simulation to create those same voxels, and then to interpolate between them