# Gamma spectrometry for forward and reverse problems in radio-isotope identification

by

Anthony Nicholas Turner


A thesis submitted to the University of Birmingham for the degree of Doctor of Philosophy

School of Physics and Astronomy
College of Engineering and Physical Sciences
University of Birmingham
United Kingdom

August 2020

**Abstract**

This work seeks to develop methods for the automated and rapid identification of sets of unknown radioactive sources via gamma-ray spectrometry. The development, testing, and integration of various tools into a simulator enabled high-fidelity modelling of radioactive sources, shielding, and detection systems. Taking full advantage of this simulator, it served to supply a rapid development environment for testing the reverse, where Radio-Isotope IDentification (RIID) algorithms analyse multi-isotope energy spectra to make predictions. Historically, algorithms fail due to unexpected or large perturbations in the profile of these spectra. For example, fluctuations in the detector gain from drifts in temperature or voltage will change the calibration. Transient effects that alter spectra over time can be extremely difficult to explicitly code, so a machine learning approach was taken to address such challenges. A Convolutional Neural Network (CNN) was designed and investigated to meet this need in the RIID problem space. With a representative set of training data, these models were seen to perform extremely well for making predictions on uncalibrated, multi-isotope gamma spectra from NaI detectors perturbed by different source activities, environmental background, low energy thresholds, and extremes in gain shifts. These models were also applied to the more challenging RIID conditions of heavy shielding and close geometries with equally promising results.

**Acknowledgements**

I would like to thank our collaborators at AWE plc. and Culham Centre for Fusion Energy (CCFE) for their continued advice and support. This work is made possible with funding from the University of Birmingham and AWE plc. Computations were performed using the University of Birmingham's BlueBEAR HPC service.

I also thank those who have made great impact on my time with the University of Birmingham. The guidence and patience of Dr Carl Wheldon has been fundamental to my every success over the past four years, and I could never have hoped for a better advisor.

To my fellow students, I am incredibly grateful for making my time here such an enjoyable experience. It is an somber realisation that no office will ever compare, and I count myself lucky to have met each and every one of you.

Finally, I am unable to express my grattitue toward my family for their unconditional support. No matter the path taken or challenges faced, you have been right there with me. I would be lost without your love and unwavering belief in my success.

## List of thesis publications

Copies of these may be found in Appendix A.2.

Turner, A., Wheldon, C., Gilbert, M., Packer, L., Burns, J., Kokalova, T. & Freer, M. Generalised gamma spectrometry simulator for problems in nuclide identification. *J. Phys.: Conf. Ser.* **1643.** 27th International Nuclear Physics Conference (INPC2019). doi:10.1088/1742-6596/1643/1/012211 (2019)

Turner, A., Wheldon, C., Kokalova, T., Gilbert, M., Packer, L., Burns, J. & Freer, M. Convolutional neural networks for challenges in nuclide identification (2020, submitted)

# List of Figures


2.1 Representation of the main interaction mechanisms for photon detection. . . 5

2.2 Diagram of the electron structure of insulators and semiconductors. Blue shading indicates initial electron occupancy. Representations of the relevant mechanisms for radiation detection are highlighted appropriately for each. These are the de-excitation of electrons in scintillators, which are insulators, and the collection of charge carriers in semiconductors under an applied electric field. 8

2.3 Diagram of a simplified $p$-$n$ junction. Incident radiation eventually results in electron-hole pairs, which are swept toward their respective contacts by an applied electric field. The movement of charged particles constitutes a basic electrical signal. . . . . 9

2.4 Diagram of a Photomultiplier Tube (PMT), used to convert scintillation light into a measurable signal. The small number of photoelectrons is multiplied through a dynode array, with an accelerating potential between each dynode. 12

2.5 Schematic of key features that may be present in an energy spectrum. . . . . 14

2.6 Diagram of external interactions that result in other key features of the energy spectrum. The external materials are not simply the detector itself, but anything in surrounding environment such as walls, benches, and shielding. . 15

2.7 Example Artificial Neural Network (ANN) structure for gamma spectra. Every channel's intensity in a spectrum is connected to all $X$ nodes of the first fully-connected or 'dense' layer. The $X$ outputs are, in turn, connected to the $N$ nodes of the next layer, before finally making a prediction. Since every connection has trainable parameters that must be optimised, ANNs rapidly increase in complexity with the number of layers and the number of nodes per layer. . . . . 22

2.8 Diagram demonstrating class and label concepts. In this case, 'classes' are the possible sources that might be present in a spectrum, while 'labels' denote which of these are actually seen. . . . . 30
2.9 Example of relatively good learning, where the training and validation curves of accuracy (a) and loss (b) metrics begin to converge over several epochs. Through monitoring these curves, many problems with the optimisation or datasets are revealed. . . . . 31
2.10 A single node of a fully-connected layer used to make a prediction. Every output ($x$) from the previous layer is connected with a trainable weight ($w$) and combined linearly. An activation function, in this case, encodes the probability of a source being present to a value between 0 and 1, with some offset defined by the bias ($b$). Validation samples follow optimisations using the training samples. . . . . 31
2.11 Applying a trainable filter to build up a feature map of new abstractions of the original. . . . . 34
2.12 Representation of the convolution module. An array of intensities represents the spectrum. This trains first $H$ filters, then $K$ filters in two convolution layers to emphasise key features in the spectra. Pooling and dropout then reduce the dimensionality and combat over-fitting. For demonstration purposes, the shapes exclude a dimension for batch size. . . . . 34
2.13 Representation of the classification module. The convolution module output is flattened into a vector. A small 'fully-connected' ANN makes final predictions of present sources, with a 0-1 probability through a Sigmoid activation. . . . 36
2.14 Simplified CNN structure. The basic sequential model consists of two modules, each performing 1D convolution operations. The resulting tensor is flattened into a vector to interface with a small, fully-connected ANN. The final output layer provides the probability of each source being present in a gamma spectrum. 37
2.15 Example plot of Receiver Operator Characteristic (ROC). The ideal case tends toward (0,1) where all predictions are correct. A more realistic curve is also shown, staying above the $y = x$ boundary that demonstrates performance achieved by randomly assigning labels. The Area Under Curve (AUC) is taken as the shaded region. . . . . 39
2.16 Example Precision-Recall (PR) curve demonstrating several cases. Ideal curves tend toward (1,0), while a baseline of randomly assigned predictions is defined by the fraction of positive samples in the dataset. . . . . 40

3.1 The `Step` is made up of pre- and post-step points. . . . 43
3.2 A `Track` is made up of multiple linked steps. . . . 43
3.3 The `Event` contains a track for every transported particle, with a new track defined for any secondary particles created. . . . 44
3.4 Model of the HPGe detector used. . . . 46
3.5 Model of one of the NaI detectors used. . . . 47
3.6 HPGe detector dimensions used for the simulated model. . . . 47
3.7 General purpose calibration curve to be used for the reference set of HPGe gamma spectra. . . . 49
3.8 FWHM calibration for use as Gaussian energy broadening parameters. Note the divergence at the low energies (<100 keV) from multiple contributions that appear to be single peaks given the detector resolution, and the influence of annihilation photons. . . . 50
3.9 Correcting for the broadening of the annihilation peak in $^{22}$Na due to a residual net momentum of $\beta^+$ and electron. . . . 52
3.10 An $^{241}$Am simulation both with and without the decay chain limit. These are compared to real data (black line, light blue fill) for reference. . . . 53
3.11 Process for limiting long decay chains by looking at the particle lifetimes of a newly created track object. . . . 54
3.12 Example of X-ray fluorescence in the assigned region with reduced production cuts. The expected escape peak is now seen ∼10 keV below the 60 keV full energy peak of this $^{241}$Am spectrum. The simple simulation environment also demonstrates a difference in backscatter and low-energy background that can be extremely difficult to generalise. . . . 55
3.13 Example of extraneous sources contributing to the background profile. The $^{137}$Cs seen in the experimental data present is absent from this point-source NORM simulation. . . . 58
3.14 Example of one attempt to generalise the background profile, this time with NORM sources homogeneously embedded in surrounding concrete. Several peaks remain either over- or under-represented in the low-energy region because of the dominance of backscatter and attenuation. . . . 59
3.15 Spurious summations between the 68/78 keV and the much later 511 keV or 1157 keV emissions of the $^{44}$Ti decay chain. In more complex decay chains there can be many more false summations. . . . 61

3.16 Basic case to demonstrate the track reconstruction, with primary (green) and secondary (red) particle steps shown. Examples A and B are two independent tracks that must be reconstructed from Hit objects. In Example B, the parent-child chains are followed through these Hits, and may be linked to combine the energy deposits of tracks $x$, $y$, and $z$. Summation between independent tracks is based on the entry times $t_A$ and $t_B$. . . . . 62
3.17 Basic reconstruction of Hit data into independent tracks that are entering the crystal. All energy deposits are added to that of the relevant independent track. 63
3.18 The times for tracks entering the detector are checked against an acceptance window, and treated as independent if outside this time window. This removes the previous false summations while retaining true summations. . . . . 64
3.19 Comparison of photopeak intensities as the position of a $^{60}$Co source is varied. Note the summation peak following the expected solid angle ($\Omega$) summation trend. . . . . 65
3.20 Demonstration of the loss of precision at large global times using the 154.8(8) ns state in $^{44}$Sc. The time logged measures the lifetime of this first excited state, with the exponential fit showing an incorrect 157.3(5) ns half-life due to rounding artefacts. . . . . 66
3.21 A simplified $^{44}$Ti decay scheme, showing reset points (dashed red) for the global time. All time differences are calculated to the nearest reset point rather than the primary particle, keeping the magnitude of stored values small. . . . . 67
3.22 Adding timing data for each new particle track, resetting the global time to zero at key points to manage magnitudes of stored times. The conditions for doing so may be redefined as needed since the handling of timing information is self-contained. . . . . 68
3.23 A scandium-44 example for global time reset and base offset calculation. The red box indicates a reset point, while blue boxes are reconstructed independent tracks. . . . . 69
3.24 Demonstration of the time reset methodology for the 154.8(8) ns [80] half-life in $^{44}$Sc. Since GEANT samples from literature values, Figure 3.20 should have agreed this well. . . . . 70
3.25 Time spectrum of a late stage excited state of $^{208}$Pb in the long $^{232}$Th decay series. The fit half-life of 279(2) ps compares well to the 294(15) ps [81] literature value, maintaining precision with no effort on the part of the user. 70

3.26 Multiple detectors with a central $^{44}$Ti source for $\gamma$-$\gamma$ coincidence measurements. . . . . 71
3.27 Comparison to the $^{44}$Sc lifetime measurement from Ref. [78]. The real data are fit with a convolution of a Gaussian and exponential, with the simulation in excellent agreement. . . . . 71
3.28 Partially implemented angular correlations of GEANT. As the $^{108m}$Ag example shows, any nuclide with the same $4^+ \rightarrow 2^+ \rightarrow 0^+$ cascade will work, but there is no generalised solution implemented for any other set of transitions. The $6^+ \rightarrow 4^+ \rightarrow 2^+$ is expected to show the same trend (though slightly supressed). 72

4.1 Example sampling of template spectra, downsampled for clarity. The original template for this selection of experimental $^{152}$Eu data are well reproduced with $1\times10^6$ randomly sampled counts, while worse statistics are emulated by taking only $1\times10^4$ counts. In the higher energy regions, where photopeaks are less prominent, the noise becomes even more exaggerated. . . . . 79
4.2 Example of the loss of information with 50 keV low-energy discrimination. Simulated spectra are used for demonstration purposes. Note that these low energy peaks are typically the most dominant features, making their loss significant for RIID. The $^{241}$Am spectrum is seen here in its entirely, hence the 50 keV maximum. . . . . 80
4.3 Gaussian noise added to a $^{60}$Co spectrum with width $\sigma = 0.2$ as an alternative for emulating poor statistics. . . . . 81
4.4 Calibration change, altering only the energies assigned to each bin. Bin 17 still contains $N$ counts but is now twice as wide on the same energy scale. This is useless for models that take only the intensities, as the peak has not moved position in the channel array. . . . . 82
4.5 Gain shift, maintaining bin structure and moving counts around. The $N$ counts of Bin 17 are now split across Bins 33 and 34 of the same array. Useful for models that take only the intensities, as the peak has now moved position in the array. This also has the advantage of inherently accommodating arbitrary and inconsistent bin structures. . . . . 83
4.6 An example 1.9× gain shift applied to a multi-isotope spectrum containing $^{108m}$Ag and $^{60}$Co. The 2.0× upper limit for gain shifts was chosen to accommodate the dominant $^{60}$Co peaks, though for spectra such as $^{207}$Bi some more minor high-energy peaks will be lost. . . . . 84

4.7 Fairly extreme example of re-binning $^{60}$Co spectrum taken with a NaI detector. With only 8000 counts in total, the original structure is noisy, even in such a simple spectrum. . . . . 85
4.8 Example selection of arbitrary datasets to build up customised training and test sets. Samples are stored individually in the core datasets, with subsets chosen by specifying paths in the *files paths* list. This is then used to generate the associated *labels* list. . . . . 88

5.1 Representation of the convolution module. This takes the $L$ entries of the previous feature map, and trains first $H$ filters, then $K$ filters in these two convolution layers. Pooling and dropout reduce the dimensionality and combat over-fitting. For demonstration purposes, the shapes exclude a dimension for batch size. . . . . 92
5.2 Representation of the classification module. The convolution module feature map is flattened into a vector. A small 'fully-connected' ANN makes final predictions of present sources, converting outputs to a 0-1 probability via the Sigmoid activation. . . . . 93
5.3 Comparison of the standard Rectified Linear Unit (ReLU) activation function with its 'leaky' variant. This adds a small gardient in the negative region to counter the 'dying ReLU' problem. The gradient ($\alpha$) is 0.2 in this case. . . . 97
5.4 Perfect classification rate performance as the dropout rates are varied in the convolution modules. . . . . 98
5.5 Representation of the classification module. . . . . 99
5.6 Learning metrics from training the reference model over 150 epochs, showing good conversion of the training and validation sets. . . . . 101
5.7 ROC curves extended to the multi-label case for the reference model. Provides comparisons of performance for each source as the decision threshold is varied. The average (red) is included in 5.7a. Figure 5.7b shows $^{22}$Na, $^{60}$Co, and $^{44}$Ti as marginally responsible for more false classifications. Dashed blue in (a) represents random label assignment. . . . . 102
5.8 Experimental $^{44}$Ti, $^{22}$Na, and $^{60}$Co spectra, showing the difficulty in resolving features. The $^{241}$Am feature is large enough to stand out from close peaks of other sources. . . . . 103

5.9 Example test spectrum containing $1\times10^6$ total counts from equal activity $^{241}$Am, $^{152}$Eu, and $^{44}$Ti. There is 6 cm of aluminium shielding between the sources and the detector. Note the loss of low energy information due to attenuation and a large backscatter feature. . . . . 104
5.10 Receiver Operating Characteristic (ROC) for a heavily shielded test data set, showing $^{241}$Am, $^{152}$Eu, and $^{44}$Ti to be responsible for more incorrect predictions. Americuim-241 in particular contributes to the majority of false negative classifications. . . . . 106
5.11 Example of the variation in classification rates as the number of training samples per unique source combination is varied. . . . . 110
5.12 Example of the variation in classification rates as the decision threshold is varied. 111

# List of Tables


2.1 Simple binary classification outcomes. . . . . 38

3.1 GEB parameters used for simulations. . . . . 51

4.1 Example template mapping. . . . . 77

5.1 Scope of the training and test sets. . . . . 91

5.2 Example of overall classification outcomes for a model with a perfect prediction rate of 31.7(5)%. The vast majority of labels are still correctly predicted, making it difficult to use these more traditional performance measures when comparing models. . . . . 93

5.3 Classification rates for a range of batch sizes. . . . . 94

5.4 Classification rates for a selection of initialisers. Those marked with an asterisk (*) failed to converge for at least one of the sets. . . . . 95

5.5 Classification rates for a range of convolution kernels and numbers of filters. For both, it is only at the lower extremes that performance suffers, quickly stabilising for increased computational expense with no additional benefit. . 96

5.6 Relevant reference model hyperparameters. . . . . 100

5.7 Classification outcomes for the reference model. These values are averaged across test sets of experimental data taken using a 3”×3” NaI detector at a stand-off distance of 10 cm. . . . . 100

5.8 All performance metrics for a range of shielding conditions. Data were collected using a 3” NaI detector at 10 cm, with shielding placed between the source and detector. Each aluminium plate is 2 cm thick. . . . . 105

5.9 All performance metrics for a range of stand-off conditions. . . . . 107

5.10 Comparison of performance on test data sets. The models trained individually are compared against the generalised model. . . . . 112

# Contents

**List of Figures** vii

**List of Tables** viii

**1 Introduction** 1

1.1 Overview . . . 1

1.2 Thesis work . . . 3

**2 Theory** 4

2.1 Radiation detection and measurement . . . 4

2.1.1 Gamma interactions . . . 5

2.1.2 Detection and measurement . . . 7

2.1.3 Energy spectra . . . 13

2.2 Radio-isotope identification . . . 17

2.2.1 Aims of RIID . . . 17

2.2.2 Technical considerations . . . 18

2.2.3 Algorithmic methods overview . . . 20

2.2.4 Machine learning in RIID . . . 21

2.3 Convolutional Neural Networks . . . 28

2.3.1 General concepts . . . 28

2.3.2 CNNs for RIID . . . 33

2.3.3 Performance evaluation . . . 38

**3 Simulation of gamma spectra** 41

3.1 Overview . . . 41

3.1.1 Radiation transport codes . . . 41

3.1.2 Simulation aims . . . 42

3.1.3 Geant4 basic concepts . . . 43
3.2 Simulation techniques . . . 46
3.2.1 Detector models . . . 46
3.2.2 Reference data . . . 48
3.2.3 Core challenges . . . 50
3.2.4 Timing . . . 60
3.3 Summary of simulator . . . 73

**4 Data preparation for machine learning 74**
4.1 Datasets . . . 74
4.1.1 Template spectra . . . 74
4.2 Preprocessing techniques . . . 75
4.2.1 Methods . . . 76
4.2.2 Handling large datasets . . . 86

**5 Radio-isotope identification models 89**
5.1 Reference data sets . . . 90
5.2 Hyperparameter trends . . . 92
5.2.1 Set-up and structure . . . 92
5.2.2 Training hyperparameters . . . 94
5.2.3 Core hyperparameters . . . 96
5.3 Model performance . . . 100
5.4 Challenging conditions . . . 104
5.4.1 Heavily shielded conditions . . . 104
5.4.2 Close geometries . . . 107
5.5 Data sets . . . 109
5.5.1 Data set size . . . 109
5.5.2 Including real data . . . 110
5.5.3 Generalising the model . . . 112

**6 Conclusions 113**

**7 Future work 117**

**Glossary 119**

**Appendices 1**

**A Publications** **2**

A.1 List of other publications . . . . . 2

A.2 Copy of thesis publications . . . . . 3

# Chapter 1

# Introduction

## 1.1 Overview

The Segrè chart shows a summary of ∼3300 nuclides known to physics, of which only 252 are stable. Stability is gained by releasing surplus energy as radiation. This may be achieved through various decay mechanisms such as alpha or beta decay. Following radioactive decay, daughter nuclei will often be in an excited state. The emission of gamma rays is an extremely common way for these nuclei to lose internal energy and de-excite to their ground state. This research focuses on the quantification and interpretation of gamma radiation.

Much like electrons, the quantised states of a nucleus exist at very specific energies. Since these states are influenced by the nuclear structure (i.e. the precise arrangement of protons and neutrons), every isotope is different. Gamma radiation is the result of transitions between these well-defined energy states and is, therefore, unique to the isotope that produced it. The goal of this research is to improve or develop new methods that use this 'nuclear fingerprint' to identify unknown radioactive material.

Gamma spectrometry is the quantitative measurement of gamma-ray sources. Energy spectra are used to identify which radioactive sources are present. These are often produced using High Purity Germanium (HPGe) solid-state detectors or Sodium Iodide (NaI) scintillators. A photon entering one of these detectors will interact and generate a signal proportional to its energy (see Section 2.1). Collecting these data in a histogram, a spectrum is formed with peaks that correspond to different photon energies. Identifications can be made by comparing peak locations with known energy data. Even this simple analysis of gamma ray energies is effective enough to be the basis of many Radio-Isotope IDentification (RIID) techniques.

RIID via gamma spectrometry effectively investigates the unique characteristics of a nu-

clide encoded within the long-range radiation. The relatively energetic gamma rays are highly penetrating due to their uncharged nature. This makes it possible to robustly perform on-site non-destructive assay of even sealed containers, removing any need for direct contact with hazardous material. As such, the applications of RIID systems span a diverse range of sectors. Perhaps the most obvious is security, and indeed much research is focused toward screening for illicit material in the context of nuclear non-proliferation [3–7].

More broadly, applications touch any area that requires alarms and detection, identification, and mapping [8]. Systems with RIID capability are incredibly useful in the decommissioning and management of nuclear waste, with legacy stockpiles filled with unknown material that must be characterised. Rapid and automated identification is essential for handing such large volumes efficiently. Handheld systems also play a crucial role in public health and safety [9, 10]. First responders to sites of nuclear incidents can manage crises far more effectively with accurate information on the nuclides involved. Large scale surveys can also be conducted, though these remain a challenging problem [11].

Unfortunately, the manual interpretation of spectra is both time consuming and labour intensive, so there has always been a drive to automate identification. Algorithms to automate RIID have historically developed alongside advances in computing. For example, peak finding algorithms were very quickly applied to NaI spectra [12] (Section 2.2.3). It is testament to the robust nature of these detectors from the 1940s [13] that they remain one of the most widely used today. Commercial availability of detectors like these has encouraged some degree of consistency across the nuclear industry. This establishes a standard of well understood hardware, which is excellent news for algorithm development.

The prevailing issue with the majority of historically used RIID algorithms is that they often fail when small perturbations alter the calibration or profile of a spectrum. For example, variations in temperature can misalign peaks, shifting them away from the calibrated energy so that direct comparisons fail.

Recent decades have seen a move toward applying machine learning methods for RIID, the history of which is discussed in Section 2.2.4. This mirrors a broader trend in physics as machine learning techniques become more accessible. Interested readers should see a recent review by Mehta *et al.* [14]. The ability to accommodate transient effects without being explicitly programmed to do so is invaluable to RIID, and eventually became a focus of this research.

## 1.2 Thesis work

In order to develop RIID algorithms that address the challenges of these applications, a simulator was required to produce varied and realistic gamma-ray spectra that mimic a range of conditions. The development, testing, and integration of various tools into this simulator enabled high-fidelity modelling of radioactive sources, shielding, and detection systems.

Due to the broad scope of this research, and the potential for use in future projects, this simulator must be as generalised as possible. The simulator was developed using GEometry ANd Tracking (GEANT) [15], an open-source Monte-Carlo radiation transport framework. Creation of synthetic spectra for a diverse range of conditions provides a rapid development environment for investigating algorithms. It also enables the development and testing of conditions for which experimental data are not readily accessible.

The prerequisite of a gamma spectrometry simulator for the testing of RIID methods defines a clear progression for this thesis. After an overview of relevant background knowledge in Chapter 2, Chapter 3 details the development of a generalised simulator that supplies gamma spectra for a variety of conditions. Chapter 4 then discusses how these spectra were pre-processed for investigating the RIID methods of Chapter 5.

For the simulator, solutions were implemented to reconstruct energy depositions so that true coincidence summations could be handled accurately, and with minimal user intervention. Additional work was carried out to handle timing information in the GEANT framework without compromising precision. This not only increased the accuracy of summation peaks, but expanded the simulator's capability to include coincidence measurements with arrays of multiple detectors.

A review of the literature and history of RIID automation certainly influenced the direction of this open project. The advantages of using machine learning techniques in this problem space were clear, with models beginning to demonstrate a level of flexibility that surpasses traditional algorithms, without being explicitly programmed to do so [16–18]. The slow progress of applying these techniques has seen a resurgence of interest in recent years as open-source tools become more widely accessible. A type known as the Artificial Neural Network (ANN) has remained the primary choice over almost three decades of research that began with the work of Olmos *et al.* [19]. The work presented in this thesis takes a new approach, basing models on a variant known as the Convolutional Neural Network (CNN). The theory and benefits of using a CNN are discussed in detail in Section 2.3. This was developed independently of other work at the time [20–22], and will elucidate the potential for CNN models to surpass the computationally expensive ANN models.

# Chapter 2

# Theory

## 2.1 Radiation detection and measurement

Gamma radiation is a fundamental mechanism for losing surplus nuclear excitation energy. It is the result of transitions between excited states of a nucleus. Much like electrons, the protons and neutrons of the nucleus may be described as only occupying certain shells and energy levels. The nucleus is excited when one or more nucleons jump to higher energy shells, or when the nucleus as a whole has excess energy through vibrations or rotations.

The energy of an excited state is well-defined, but still has a finite width. The intrinsic width is inversely related to the lifetime through the Heisenberg uncertainty principle,

$$\delta E \times \delta t \geq \frac{h}{2\pi}, \tag{2.1}$$

where $\delta E$ may be taken as equivalent to an energy resolution, and $\delta t$ the mean lifetime. Lifetime ($\tau$) can be expressed as a function of decay constant ($\lambda$) or half-life ($T_{1/2}$) as

$$\tau = 1/\lambda = T_{1/2}/ln(2). \tag{2.2}$$

The state width is broader for shorter lived states, but even the 0.9 ps [23] half-life of the 1333 keV state in $^{60}$Co has a width of $\sim 3\times10^{-4}$ eV. This is negligible in comparison to the practical limitations of the detectors used, which often measure the result of statistical fluctuations in units of kiloelecton volt (keV).

Of course, gamma radiation is not the only way that a nucleus can release surplus energy. Internal conversion sometimes competes with gamma emission. Nuclear energy is transferred directly to a bound electron, which is then ejected from the atom as a conversion electron.

One good example is $^{137}$Cs, which $\beta^-$ decays to populate the 662 keV state of $^{137}$Ba around 95% of the time [24]. The only gamma transition from this state has a relative intensity of 85%. The "missing" 10% undergoes internal conversion instead, emitting near monoenergetic electrons (assuming the energy depends only on the electron shells). The vacancy left will induce the release of secondary radiation, as described in Section 2.1.1.

With any uncharged incident radiation, the immediate goal of a detector is to generate charged particles. Being charged, these signal carriers are considerably more effective at depositing their full energy in a detector's active region, where a measurable signal is produced. The following sections discuss various principles important for physically measuring the energy of gamma rays emitted by a radioactive source.

### 2.1.1 Gamma interactions

There are several ways for a photon to interact once it enters a detector. A majority of features seen in the energy spectrum can be attributed to a combination of three primary mechanisms. These are, in order of probability with energy: the photoelectric effect, Compton scattering, and pair production. Interactions are depicted in Figure 2.1.

It is worth noting that while other interactions such as Coherent (Rayleigh) scattering[1]

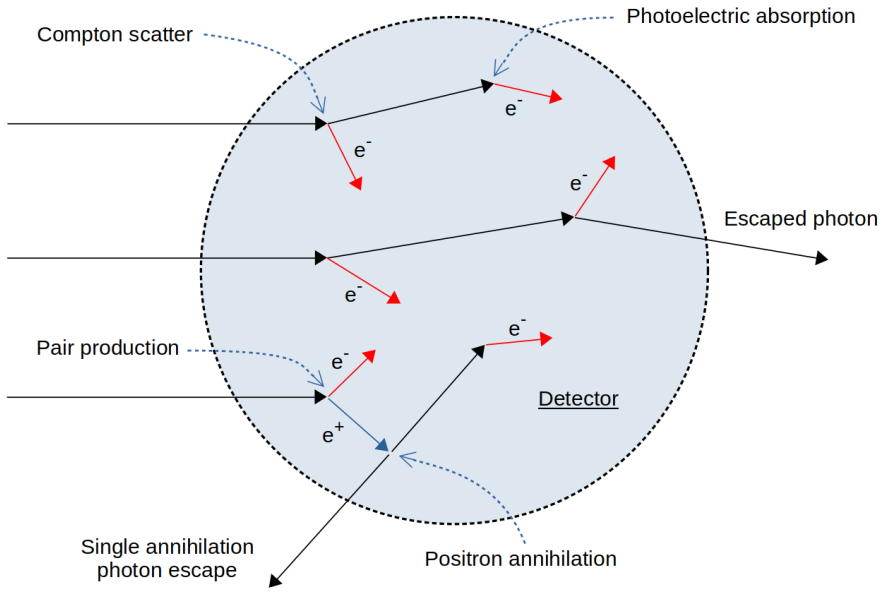



Figure 2.1: Representation of the main interaction mechanisms for photon detection.

[1]Coherent scattering neither ionises or excites the absorber atom. Only the direction of the incident photon changes [25], with virtually no loss of energy after interacting coherently with all electrons in an atom. This typically occurs at very low energies, but can become more prevalent in high Z detectors.

are possible, the effects may be considered negligible for the applications of this research.

At lower photon energies, the photoelectric effect is dominant, especially in high-Z materials. In this interaction, the photon is completely absorbed by a bound inner electron. This provides a two body system following the absorption, in which energy and momentum become distributed between the ejected electron and a recoiling atom. Energy and momentum are not simultaneously conserved for absorption with a free electron.

The ejected 'photoelectron' is emitted with energy $E_{e^-} = E_\gamma - E_{binding}$. An inner electron absorption becomes increasingly likely with increasing photon energy. This is most often the case in gamma spectroscopy, where the photon energies generally far exceed the binding energy of most K-shell electrons. The recoil atom is left ionised after the interaction, so secondary atomic effects such as characteristic X-rays and Auger electrons may be present if the vacancy is not filled by capturing a free electron from surrounding material. The proton number (Z) of the absorber atom also effects the probability of interaction, with high Z materials increasing the likelihood of absorption. This trend is typically quoted in terms of the photoelectric absorption probability ($\tau$) as $\tau \propto Z^4/E_\gamma^{3.5}$ [25, 26], though the exponents vary slightly with photon energy. This is why high-Z detectors, like scintillators, are often chosen for gamma spectroscopy due to the much improved detection efficiency, despite shortcomings in other areas such as resolution and timing response.

The probability of Compton scattering increases with photon energy. After scattering with an electron, an incident photon will be deflected from its original path. The recoil electron may possess a range of energies, spanning from virtually zero up to some maximum defined by backscatter (photon scatter at $\pi$ radians relative to the direction of incidence). Compton scattering is described by

$$E' = \frac{E}{1 + \frac{h\nu}{m_0 c^2}(1 - cos(\theta))}, \tag{2.3}$$

where $E$ and $E'$ are the photon energy before and after the scatter, and $m_0$ is the rest mass of an electron. The varying deflection angle ($\theta$) results in a continuum of energies known as the Compton continuum. If enough energy is imparted to a bound recoil electron, atoms can be left ionised.

As photon energies increase further still, pair production begins to occur. Pair production is only possible above a 1.022 MeV threshold. In this interaction, the photon is destroyed to create an electron-positron pair. Any excess energy above the $2m_0c^2$ minimum is carried off as kinetic energy. To conserve both energy and momentum, the photon will interact in the vicinity of a nucleus, imparting momentum to a recoil atom for full conservation. As

energy increases above ∼2 MeV this process can start to be of more significance, producing additional features in the energy spectrum such as escape peaks (see Figure 2.1).

All of these interactions end up transferring the photon energy to fast electrons at some point, often leaving atoms in an excited state. To gain stability, an atom with a vacancy can rearrange its electrons from other energy levels. For example, a characteristic X-ray is often emitted when an outer electron de-excites to fill the vacancy. Alternatively, the excess excitation energy may be transferred directly to one of the outer electrons, which is ejected and known as an Auger electron. These secondary radiations are quickly reabsorbed.

The photoelectric effect, Compton scattering, and pair production provide multiple interaction mechanisms for converting a photon's energy into charged particles (very often fast electrons). The next section looks at how these secondary electrons interact with the detection medium to produce a measurable signal.

### 2.1.2 Detection and measurement

Again, the fundamental role of any gamma detector is to generate secondary charged particles to act as signal carriers. These charged particles are extremely ionising and rapidly deposit their full energy within the detection medium. Two of the most ubiquitous detectors in gamma spectrometry are the HPGe and NaI detectors. Radiation is detected via interactions with the crystals of both, though the method of charge collection varies.

Free, neutral atoms have a precisely defined electronic arrangement. When combined into a crystal structure, the collection of atoms establishes allowed energy bands. The outer-shell electrons bound to atoms in the lattice occupy the valence band. The next available energy state above this is the conduction band, where electrons are free to migrate through the crystal and contribute to the conductivity. The two are separated by a forbidden region known as the band gap with energy ($E_{gap}$). This is represented in Figure 2.2 for both insulators and semiconductors.

For electrons to migrate through the material they must overcome the band gap energy to reach the conduction band. When an electron is promoted to the conduction band, a vacancy known as a 'hole' is left in the valence band. Holes are also mobile, and are effectively positively charged. When an electric field is applied to a semiconductor, electrons and holes drift to their respective contacts. They each contribute to the conductivity since both are charged. The semiconductor (or 'solid-state') detectors make use of this property to collect the electron-hole pairs and produce a signal. As the most direct method of charge collection, these are discussed first.

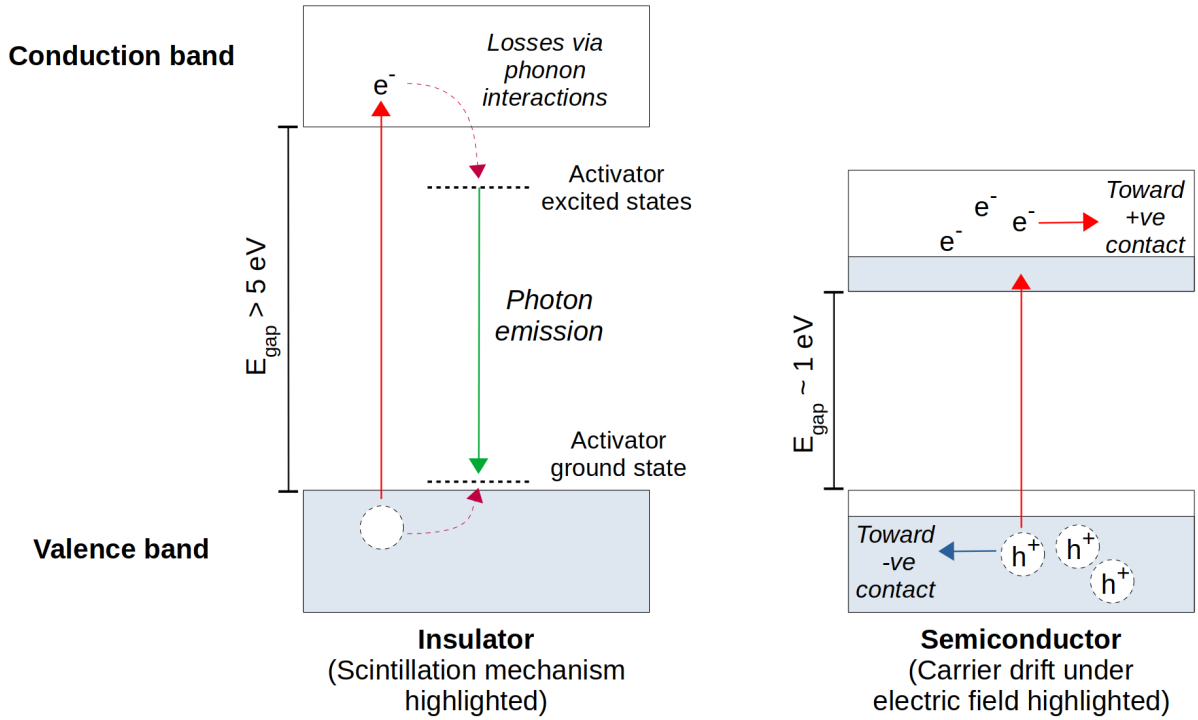



Figure 2.2: Diagram of the electron structure of insulators and semiconductors. Blue shading indicates initial electron occupancy. Representations of the relevant mechanisms for radiation detection are highlighted appropriately for each. These are the de-excitation of electrons in scintillators, which are insulators, and the collection of charge carriers in semiconductors under an applied electric field.

### Solid-state detectors

A photon absorbed in the detector results in a fast electron that far exceeds band gap energies. This induces many excitations along its trajectory by imparting kinetic energy to electrons in the valence band. This can, and often does, promote an inner-shell electron well above the base of the condition band. The ensuing redistribution of the electron-hole pair induces further excitations as they move to the bottom and top of conduction and valence bands respectively. Note that the electron-hole pairs are formed within a few picoseconds along the particle track [25], an important simplification in simulations later.

The important value is the ionisation energy ($I$), taking an average of 3 eV to create an electron-hole pair in silicon and germanium. This is largely independent of the incident radiation energy, so the number of signal carriers produced can be considered directly proportional. If the ionisation energy is the average energy taken to create an electron-hole pair, then the number of pairs produced ($n$) is a function of gamma energy ($E$) absorbed.

$$n \approxeq E/I \tag{2.4}$$

This direct relation is excellent for measurements, and $n$ becomes extremely important to energy resolution. With an incident photon converted into many electron-hole pairs, the goal is now to integrate this charge and extract the electrical signal from the crystal.

By adding impurities into the crystal lattice, donor and acceptor sites can create states close to the conduction and valence bands. This changes the band structure slightly, with adjustments made through impurities known as doping. A '$p$-type' has an excess of holes, while '$n$-type' has an excess of electrons.

Semiconductor gamma detectors rely on what happens when $p$ and $n$ types are placed in contact. These are usually grown in place by evaporation and diffusion or ion implantation to make a single crystal. Where they meet becomes a $p$-$n$ junction, as shown in Figure 2.3. Electrons and holes diffuse even under thermal influence, with electrons drifting to the $p$ side and vice versa, annihilating when they meet in the middle. The result is known as the 'depletion region', where charge carriers have cancelled each other out.

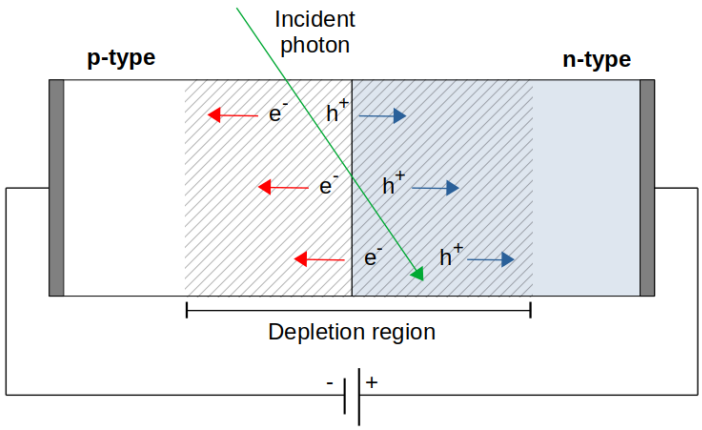



Figure 2.3: Diagram of a simplified $p$-$n$ junction. Incident radiation eventually results in electron-hole pairs, which are swept toward their respective contacts by an applied electric field. The movement of charged particles constitutes a basic electrical signal.

The depletion region has very attractive properties for radiation detection. It is this area that forms the active region of the detector. While initially very thin, applying a voltage across the material extends this to a thickness ($d$) as:

$$d = \left(\frac{2\epsilon V}{eN}\right)^{1/2}, \tag{2.5}$$

where $V$ is an applied 'reverse bias', $N$ is the net impurity concentration, $\epsilon$ is the dielectric constant, and $e$ the electron charge. If the $p$ side is made negative with respect to the $n$ side then the junction is reverse biased, enhancing the natural potential difference across the junction.

Given the range of gamma rays in the detector, the depletion region is typically extended across the whole crystal. The steady equilibrium of the depletion region is only significantly disrupted by incident radiation, with electrons and holes swept toward their respective contacts by the applied field. This motion constitutes a basic electrical signal for measurement, with charge converted into a voltage pulse by a preamplifier.

For the coaxial HPGe detectors used, a $n^+$ (heavily doped $n$) contact is made with lithium diffusion on the inside, high-purity $n$-type in the middle, and a $p^+$ outer contact made by implanting boron. These contacts are dead layers that do not contribute to the active region of the detector.

The time taken for electron-hole pairs to reach the contacts is the charge collection time. This depends on carrier mobility in the applied field, and their position of origin. The difference in electron and hole mobilities is only a factor of ∼2-3 [25, 27], with holes being slower. These are almost constant in low fields, but reach a saturation velocity in larger fields.

The shape of the signal's rising edge depends on the position of the pair, as one almost always arrives sooner than the other. Since these detectors rely on full current integration of the pair, integration times must therefore be long enough to accommodate both.

Of course, not just any semiconductor material will work. For practical use, it must have good carrier mobility and no traps to stop them reaching contacts. This means using a high purity material in as perfect a crystalline state as possible. Combining this constraint with the need for a high Z material that is manufacturable to such specifications - while remaining commercially viable - leaves few candidates. These are materials like silicon, germanium, and the newer Cadmium Zinc Telluride (CZT). The large crystal size, excellent carrier mobilities, and larger Z really set apart the germanium detectors used in this research. The much smaller band gap of 0.7 eV [25] does mean that thermal excitations constantly produce noise, so a HPGe detector must be cooled to around 77 K with liquid nitrogen. As discussed in Section 2.2.1, this is one of many concerns for pragmatic, large-scale deployment of RIID systems due to the cost of maintenance. Scintillation detectors therefore remain a crucial mainstay of the nuclear industry.

### Scintillation detectors

Since their inception, NaI scintillation detectors have become a cornerstone of gamma spectrometry, quite literally setting the standard. Manufacturing processes are refined to the point that very large crystals are possible while remaining relatively inexpensive. Scintillators such as NaI are insulators, so while the creation of charge carriers is the same process as with semiconductor detectors, the charge collection differs. Unfortunately, to the detriment of energy resolution, there is a rather long and inefficient signal chain involved.

Following absorption of an incident photon, energetic electrons are ejected from an atom in the crystal. Again, excitations are induced as it passes through the band structure of the crystal, exciting electrons from the valence band into the conduction band. These electrons rapidly reduce to the lowest conduction band energy, preferentially via phonon interactions. In this case, de-excitation is allowed to occur across the forbidden gap, resulting in the emission of photons as scintillation light. This light has energies near, if not in, visible wavelengths. This process is highlighted in the insulator schematic of Figure 2.2.

The issue with this process is that these photons have the perfect energy for immediate reabsorption, drastically reducing the light yield as a consequence. Crystals therefore need to be as transparent as possible to their own the scintillation light. By including an 'activator' the wavelength of these photons may be increased.

An activator dopant provides additional states in the forbidden gap of the crystal. With these, an electron at the bottom of the conduction band will preferentially lose energy via phonon interactions to reach an activator level, which generally sits just below the conduction band. De-exciation from one of these sites will emit a photon with energy below the threshold necessary for reabsorption. A thalium dopant is often used in NaI with concentrations of $\sim$0.1%. De-excitation through these sites produces scintillation light of 410 nm compared to the $\sim$320 nm of intrinsic NaI [13]. The choice of dopant is somewhat limited to what is chemically viable for substitutions. For example, lanthanum halide detectors such as $LaBr_3$ have a cerium activator, shifting the $\sim$360 nm [28–32] emission spectrum well below the band-gap energy.

Scintillation light, now able to pass though the crystal, must be converted into a measurable signal. This task typically falls to a Photomultiplier Tube (PMT), depicted in Figure 2.4. These are important to understand as they are often the cause of many transient effects that RIID algorithms have difficulty with, such as gain shifts.

Photoelectric absorption at the photocathode of a PMT converts the scintillation light into photoelectrons. The PMT is paired with the crystal to maximise efficiency, with the

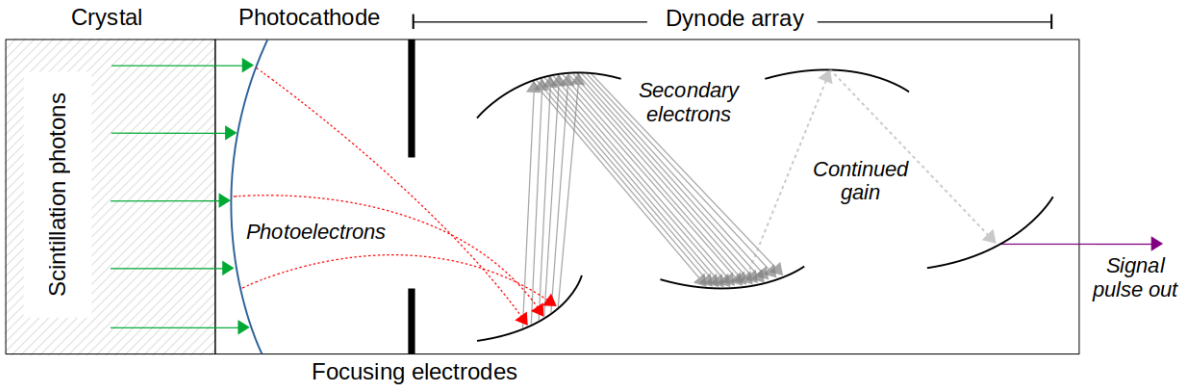



Figure 2.4: Diagram of a Photomultiplier Tube (PMT), used to convert scintillation light into a measurable signal. The small number of photoelectrons is multiplied through a dynode array, with an accelerating potential between each dynode.

photocathode well matched to visible or UV scintillation wavelengths. Efficiency is diminished somewhat by the workfunction of the surface barrier, which is a potential barrier between the cathode surface and vacuum. An electron must therefore exceed this minimum energy in order to escape the surface. The position of the electron will increase the amount of energy needed, as those produced deeper in the photocathode suffer losses via collisions during migration to the surface. A compromise exists between ensuring a photocathode is thick enough to absorb scintillation light, while remaining thin enough to allow photoelectrons to escape. The result is often a very thin (extending to ∼25 nm for some semiconductors [25]) photocathode that is semi-transparent, with the conversion quantum efficiency limited to a demonstrated maximum of around 20-30% [25].

A low surface barrier makes thermionic noise relatively unavoidable. Electrons in the conduction band have an inherent kinetic energy of ∼0.025 eV. However, the tail of the energy distribution allows the upper extreme to exceed the potential barrier, contributing to a spontaneously induced signal. This is roughly $\sim 10^2$ m$^{-2}$ s$^{-1}$ depending on the characteristics and bias of the PMT [25, 26].

The semi-transparency and low conversion efficiency mean that the signal consists of very few ($\sim 10^2$) photoelectrons at this stage. A dynode array beyond the photocathode multiplies the number of electrons, initially directed by focusing electrodes. The first dynode has a positive bias relative to the photocathode, accelerating ∼1 eV photoelectrons to strike its surface. Appropriate dynode material permits the re-emission of multiple electrons as they are excited by the passage of the accelerated photoelectrons. Most are trapped, but enough escape to contribute to the yield. A series of dynodes achieves gains of the order $10^6 - 10^{10}$.

The final number of electrons is often linearly proportional in its amplification, and much of the timing information is maintained. Interestingly, emission times of electrons are 0.1 ns or less [25], so it is the electron trajectory that becomes the dominant factor for timing characteristics.

The number of electrons propagated through the PMT is discrete and subject to random fluctuations. Assuming the formation of photoelectrons is a Poisson process, an average total of $N$ electrons will have a standard deviation $\sqrt{N}$ from purely statistical fluctuations. With $N$ being large, the signal is approximately Gaussian, and produces wider peaks in the energy spectra. This important effect is known as Gaussian Energy Broadening (GEB).

The relatively low light yield and a particularly poor conversion efficiency at the PMT contribute to a sub-optimal energy resolution. Scintillators are therefore far more limited by counting statistics than semiconductor detectors. Newer scintillation crystals such as $LaBr_3$ are slightly better, improving on NaI's resolution of ∼6% for the standard $^{137}Cs$ peak to resolutions of ∼2-3%. By comparison, HPGe can achieve resolutions below 0.5% to produce excellent quality energy spectra. These resolutions are calculated as the Full Width Half Maximum (FWHM) of a peak divided by its centroid energy.

Note that increases in environmental temperatures will induce more thermal noise in the PMT, providing extra energy for escaping the dynodes. This increases the yield of secondary electrons to be amplified, changing the overall gain. Increasing the accelerating potential between dynodes will also produce more secondary electrons, resulting in another gain shift. While the exact processes may differ between detectors, all energy spectra can suffer from these effects that are very detrimental to RIID.

### 2.1.3 Energy spectra

The features of energy spectra may be understood by bringing together properties of the detectors and the interaction mechanisms involved. Most are summarised in the exaggerated representations of Figure 2.5. Arguably, the most important feature of an energy spectrum is the full energy peak. This would ideally present as a delta function at a single, well-defined position. In reality, this peak is influenced by charge collection statistics — typically most prominent in scintillators [25, 27] — along with electronic noise and drifts in operating parameters during an experiment.

Of course, the interaction mechanisms of Section 2.1.1 often result in only partial deposition of the full energy. The finite size of the detectors makes the escape of secondary electrons and photons unavoidable.

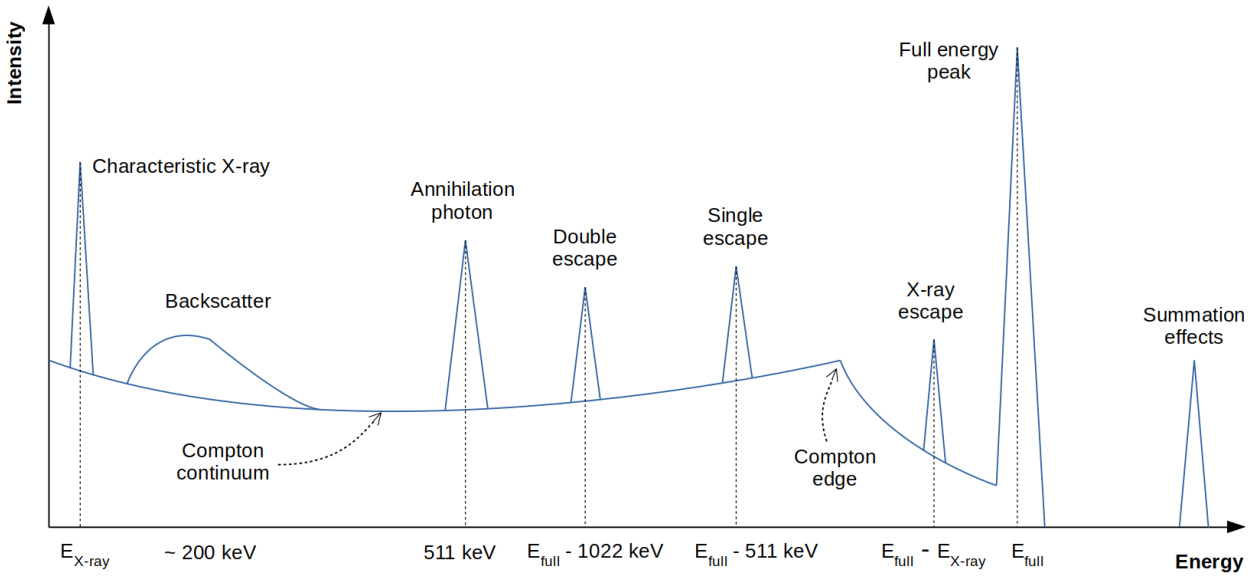



Figure 2.5: Schematic of key features that may be present in an energy spectrum.

There are times when an inner vacancy left after photoelectric absorption will not be filled by a free electron from the crystal. Instead, this vacancy may be filled by an electron from a higher orbital of the excited atom, emitting a characteristic X-ray. This is a desirable property when using Moseley's law [33] to identify elements, but in these detectors the X-ray may escape, presenting as another peak at $E_{full} - E_{X-ray}$. Alternatively, the excitation energy may be transferred directly to an Auger electron. However, the mean free path of an electron in these crystals is incredibly short and will deposit its full energy without escaping.

One of the most recognisable and significant features of a spectrum is a consequence of a photon escaping after a single Compton scatter. The recoil electron may have a range of energies from virtually zero up to a maximum following backscatter[2]. Equation 2.3 describes the energy loss after a Compton scatter. The range of potential energies is a continuum as the deflection angle ($\theta$) varies from almost zero, where there is virtually no change, up to a maximum energy that is imparted to the recoil electron after scattering through $\pi$ radians. This maximum is seen as the 'Compton edge' of Figure 2.5. Of course, multiple scatters can increase the energy deposited beyond this maximum, closer to the full energy peak.

Following pair production, the electron will usually deposit its full energy, but the annihilation photons have a lower interaction probability and thus a greater chance of escaping. Both annihilation photons have an energy of ≈511 keV, so if one or both photons escape the

[2]Not to be confused with the cause of the 'backscatter peak' where photons are scattered back into the detector following interactions with surrounding materials

result is seen as a 'single' ($E_{full} - 511$ keV) or 'double' ($E_{full} - 1022$ keV) escape peak.

Summation effects can be caused by isotopes that emit multiple gamma rays in cascade during the decay. These are effectively in coincidence and arrive at the detector almost simultaneously. The textbook case of $^{60}$Co is a prime example, with the summation of 1.17 MeV and 1.33 MeV gammas forming a sum peak at 2.5 MeV. This is known as a true coincidence summation. Random events where multiple photons arrive at the same time from different nuclides or extraneous sources are also possible, but generally only becomes a consideration at high activities.

To this point, features of the energy spectra have originated from interactions within the detector. There are always external processes to be considered. The surrounding materials such as shielding and detector housing may influence spectra in several ways. For example, a Compton scattered photon may be redirected back into the detector. The result is an important feature known as the backscatter peak, typically presenting around ~200 keV on the existing Compton continuum. It is formed due to a relatively large range of scattering angles close to 180 degrees (beyond ~110°) [25] resulting in a small range of energies. It presents as a largely localised feature because of this, but remains less of a peak and more of a continuum, since energy lost before entering the detector still has the distribution of Equation 2.3.

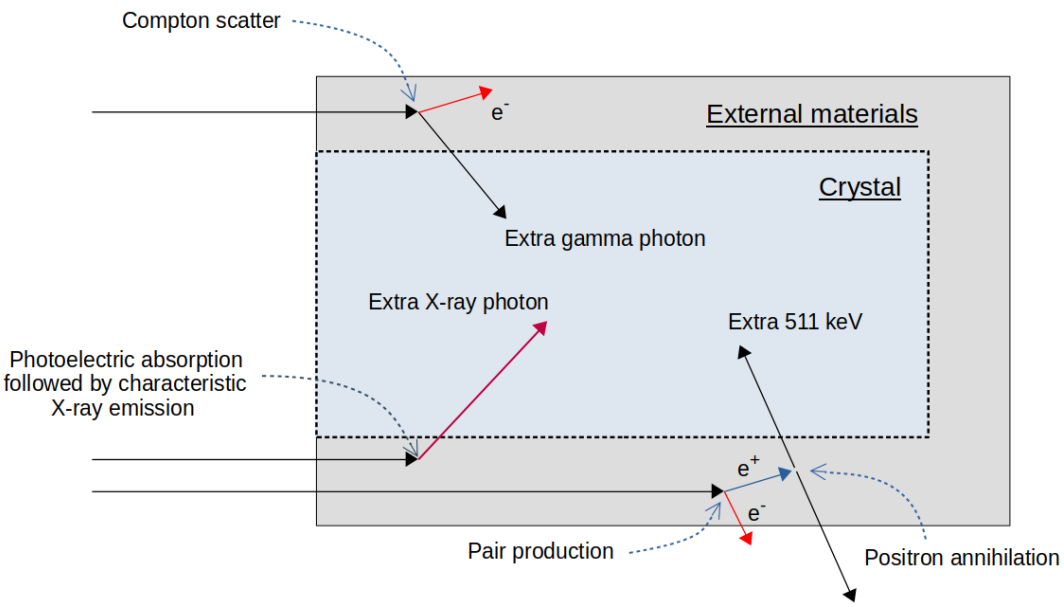



Figure 2.6: Diagram of external interactions that result in other key features of the energy spectrum. The external materials are not simply the detector itself, but anything in surrounding environment such as walls, benches, and shielding.

Characteristic X-rays are also fairly common if source photons are well matched to K-shell transitions, resulting in additional low-energy X-ray peaks. Of course, the reverse is also true, where X-rays produced at the periphery may leave the active region without interacting, resulting in an X-ray escape peak.

Pair production in the surrounding materials is always possible and the annihilation photons may enter the detector, which further contribute to a peak at 511 keV. This can be especially undesirable if annihilation photons from $\beta^+$ decays are a focus of an investigation, such as those involving coincidence measurements. Note that it is usually impossible for both annihilation photons from a single event to enter the detector at the same time to result in a 1.022 MeV peak since the annihilation photons are emitted back-to-back.

Under circumstances where electrons are likely to reach the detector, such as in a vacuum, internal conversion in the decay chain of a source can cause peaks close to the energy expected from the gamma emissions. One notable example is $^{137}$Cs. After a $\beta^-$ decay, the resulting $^{137}$Ba nucleus will either de-excite by emitting a photon, or, more rarely, transfer its energy directly to an electron through internal conversion. This fast electron will be emitted with an energy $E_{full} - E_{binding}$ depending on the energy state, and will produce discrete peaks based on their origin (eg: K and L shells give 624 keV and 656 keV emissions respectively [24]). An even rarer mechanism is pair production directly from the excitation energy of the nucleus, but as expected this only occurs at very high excitation energies. Note that this is a phenomenon of itself and different to spontaneous pair production from high energy photons.

Beyond the sources of interest there may also be environmental sources of background radiation such as Naturally Occurring Radioactive Material (NORM) or even cosmic rays, though under normal experimental conditions these are far too energetic to interact at any significant rate if at all. Due to these being sources like any other, the result is that extra peaks and all their associated features underlay the data of interest. In-depth discussions of NORM are confined to Sections 2.2.2 and 3.2.3.

For radiation detection and measurement there are many mechanisms and physical effects that go into creating a measurable signal. These often interconnected effects result in a range of features, building up the nuclear fingerprint to identify the sources present. With so much information packed into a spectrum, it becomes challenging for RIID algorithms to separate them out and relate certain features to any one source. The next section discusses the aims of RIID and how the effects described here can cause them to fail, ending with what approaches researchers have previously adopted.

## 2.2 Radio-isotope identification

The subject of Radio-Isotope IDentification (RIID) algorithms often covers techniques designed to identify radioactive material using gamma spectrometry. Ideally these are automated as far as possible to remove some of the burden placed on trained experts. In the following sections, the challenges of effective RIID are discussed, as are various approaches that have proven more successful in the past.

### 2.2.1 Aims of RIID

Humans generally outperform algorithms without contest when it comes to characterising radioactive material [8]. Trained spectroscopists are frequently required to conduct secondary analysis to resolve ambiguous or unreliable predictions [5]. They are capable of identifying multiple sources with low resolution systems and, most importantly, can accommodate unexpected changes. Addressing this need to handle unexpected perturbations to the profile of spectra is crucial to narrowing the current disparity in performance.

For security in particular, agencies rely heavily on low resolution systems such as NaI and PolyVinyl Toluene (PVT) detectors [5]. This contributes to the fact that a vast majority of all alarms are labelled as 'nuisance' alarms due to NORM and medical isotopes [3, 7]. One solution would be to use high resolution HPGe detectors everywhere and have vehicles or packages sit between detectors all day to collect excellent spectra. Obviously this is completely impractical, highlighting the need for considerations outside of the technical (Sec. 2.2.2). Any system must keep in mind the realities of deployment, where the commercial availability and robustness of existing hardware cannot be ignored. For example, small collection times with unfavourable detector coverage will result in relatively few counts, and smaller features that may be used as key identifiers are quickly lost to noise. Ideally then, a RIID system aims to work within the constraints imposed by practical use.

Of course, bespoke solutions will always outperform the generalised. However, if the underlying techniques are flexible, specialisation can be more easily incorporated by building upon a generalised solution. A good RIID system will therefore be adaptable to the needs of the situation. For example, by tweaking the final layer of a neural network, its purpose can change. With this, a classification problem such as isotope identification could be repurposed to solve a regression problem to estimate activities. Whatever the case, any information conveyed to the user should also be clearly interpretable by a non-expert and indicate a level of confidence [34].

There are always compromises to be made for the pragmatic deployment of RIID systems. It is an unavoidable reality that the algorithms must aim to handle perturbations in the spectrum profile, be flexible to the goals the system, and be fast enough to handle a reasonable throughput. Understanding operational constraints and working toward these broader aims is key to producing truly effective algorithms.

### 2.2.2 Technical considerations

The question of 'what radioactive source is present?' is a simple concept that RIID attempts to solve. As discussed, addressing the broader aims of what a RIID system should be capable of already introduces several practical limitations. However, the fundamental problem systems face is the sheer number of unknowns. So much can change the profile of a spectrum that it becomes impossible to explicitly code a comprehensive solution. The following discussion considers technical and physical effects that cause many algorithms to fail.

Starting with what is known, commercial viability dictates the landscape of RIID systems almost entirely, with PVT, NaI, and HPGe detectors being the most common. This is good for algorithm development because it introduces a certain level of hardware standardisation, where the detector characteristics are well known. Efficiency, resolution, and operational constraints can all be accounted for and included as part of the algorithm if necessary.

The greatest of all unknowns are the physical effects that may change unexpectedly over time, or between measurements. These transient effects include gain shift, where fluctuations in temperature or applied bias will cause several issues. Over small fluctuations the energy calibration will change, while at extremes there may be several detector-specific effects. For example, a HPGe detector at room temperature is entirely dominated by thermal noise, and lower voltages would cause resolution to suffer as information carries are increasingly lost to incomplete charge collection.

On the subject of the detectors, common issues with the electronic signal chain can include pile-up or dead time. Of course, for RIID this is generally a non-issue because they are only non-negligible at high activities. Security applications would immediately flag this for inspection, and any other application can simply reduce the solid angle by moving a little further away. While high activities may not be a practical concern, low activities certainly are. In order to collect a spectrum as quickly as possible, the detector should be as close as possible. However, this introduces additional features from closely timed decay emissions called summation peaks. Accurately representing these in simulations became a significant focus of this work, as discussed in Section 3.2.4.

Other experimental practicalities must also be accounted for, such as setting low energy discrimination to cut off low-level noise. Even the detector size will change the profile of the Compton continuum and how prevalent escape peaks are.

External influences from the environment can also have a large role to play. Shielding in particular is a huge problem for RIID. As discussed in Section 2.1, any material between the source and detector typically causes a significant loss of information. The low energy peaks are attenuated to the point of being lost, and backscatter imposes a large continuum around ∼200 keV that changes the profile considerably.

Finally, there is the issue of dealing with unknowns that can not be reliably predicted. The background radiation for example. Contributions from $^{238}$U, $^{232}$Th, and $^{40}$K are foreseeable assuming secular equilibrium and typical concentrations, but these nuclides are embedded in various materials surrounding the system and are inherently shielded to some extent. This makes even NORM difficult to predict with much accuracy because low-energy attenuation will depend on how deeply nuclides are embedded and in what material. On top of this, the profile is further altered by variations in backscatter for different RIID conditions.

There may then also be non-NORM sources present in the background. For example, the experimental data used included $^{137}$Cs from a source cupboard ∼10 m away. Negligible for NaI, but HPGe spectra demonstrate a clear enough photopeak among the $^{238}$U and $^{232}$Th emissions. Another peak at 2.2 MeV with associated escape peaks was also present due to a neutron bath in the same laboratory[3]. This highlights the possibility of activations further contributing to the background, and is particularly true of active interrogation methods [35] where a sample is intentionally bombarded with neutrons or photons to induce fission.

Collecting all of these considerations together, it becomes clear why the question 'what radioactive source is present?' has a non-trivial answer. RIID systems need to be able to handle changes to the spectra due to transient effects and the local environment. The same collection of sources can produce drastically different spectra, and not accommodating certain effects is why many algorithms fail. Some of these techniques are described in Section 2.2.3. By keeping these technical considerations in mind, the root of any difficulties faced often becomes clear.

---

[3] An Am-Be source is located at the centre of a water tank. The $\alpha$ emitted during $^{241}$Am decay is captured by $^{9}$Be, resulting in $^{12}$C+n. The fast neutron quickly slows to thermal energies and is captured by $^{1}$H in the water to make $^{2}$H. The deuteron binding energy is 2.2 MeV, which is released as the gamma seen.

### 2.2.3 Algorithmic methods overview

The vast majority of algorithms need a library of spectra or tabulated data to compare against. The main RIID approaches have historically covered: manual identification, simple library comparisons, Region of Interest (ROI) methods, template matching, and peak finding algorithms [6].

Manual identification by experts will always be required at some stage, whatever the application. Here, a trained spectroscopist will analyse spectra by identifying prominent photopeaks, making their own corrections for branching ratios and efficiencies. Crucially, the spectroscopist can use deductive reasoning and draw on experience to handle the unexpected. This is the most time- and labour-intensive method, but necessary nonetheless.

Library comparisons attempt to automate the most basic part of what an expert does, where the centroid of each peak is found automatically and compared to a library, ignoring additional information such as area or FWHM. This is extremely sensitive to the calibration. Any drifts will quickly throw off the identification, as will any missing peaks due to high background, shielding, or a failure to find important peaks.

ROI methods find the total number of counts in a series of small regions. Identifications are made by considering the difference between these and the net counts of a baseline background. Any regions with elevated counts are then checked against expectations for different isotopes. Unfortunately, overlap between isotopes is an issue. Specified ROIs are stored for every nuclide of interest, but since they cannot overlap the library size is quickly limited[4]. The upside is that the method is computationally cheap, so it is often used for gross count alarms instead.

Template matching uses the whole spectrum, fitting the experimental spectrum with every template contained within its library to look for the most similar. There are many templates where factors like detection medium, shielding, and distances are varied for each isotope. Of course, this is also limited in that source mixtures vastly increase the number of templates needed due to many possible combinations, making it difficult to scale into a more comprehensive library.

Finally, there is peak finding and characterisation. Much like the basic library comparison, this attempts to emulate the expert's use of key information. Smoothing is applied before automatically finding and fitting peaks to get the number of counts, FWHM, and centroid. These data are used together in an extension of the centroid-only library comparison. However, the matching of several peaks to a single isotope becomes a greater challenge because

[4]One example for security is that there is clear overlap in the 378.5 keV and 413.66 keV peaks of $^{177}$Lu and the 375.0 keV and 413.69 keV peaks of $^{239}$Pu

the algorithm needs to know relative intensities and the detector efficiency curve in order to use them. This is again very sensitive to calibration shifts.

The common theme when examining any of these approaches is that they are rigid in their scope of applications. They are sensitive to transient effects and often unable to adapt to situations outside of narrowly defined conditions. On top of all the code needed for comparing data or looking up libraries, it becomes a monumental task to be able to explicitly handle all the possible combinations of unknowns. So how best to handle different activities, or shielding, or summation, or escape peaks? These questions have driven interest in machine learning approaches, which generalise without being explicitly programmed to do so.

### 2.2.4 Machine learning in RIID

Neural networks are the most promising type of machine learning for the RIID problem space when it comes to gamma spectrometry. While models such as random forest, k-nearest neighbour, and support vector machines are sometimes seen, they appear to be less consistent and less accurate in a like-for-like comparison [21]. These models look for clusters of correlated features which usually requires preprocessing to manually extract them. Data can also become biased depending on the algorithms used. As such, the discussion that follows focuses on only neural network based approaches.

#### Early work 1991-2002

Algorithmic methods for the RIID challenges have been around for as long as gamma detectors [12]. It was not until the early 90's that a machine learning model was first applied to nuclide identification by Olmos *et al.* [19]. This was around the time that processors were up to the task of handling something with the complexity of a neural network. The common theme throughout almost three decades of development is the near exclusive use of fully-connected Artificial Neural Networks (ANN), shown in Figure 2.7 (see Section 2.3 for details). This initial work states that *"Preliminary results are good enough to consider these adaptive structures as powerful and simple tools in the automatic spectrum analysis"*, paving the way for further development. The work looked at NaI and Ge(Li) for combinations of four simple radioisotopes, taking the raw 1024-channel data as input.

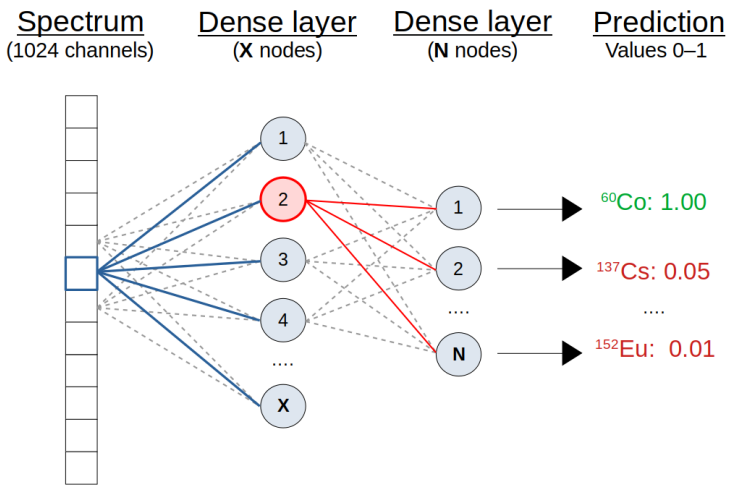



Figure 2.7: Example Artificial Neural Network (ANN) structure for gamma spectra. Every channel's intensity in a spectrum is connected to all $X$ nodes of the first fully-connected or 'dense' layer. The $X$ outputs are, in turn, connected to the $N$ nodes of the next layer, before finally making a prediction. Since every connection has trainable parameters that must be optimised, ANNs rapidly increase in complexity with the number of layers and the number of nodes per layer.

The original paper was quickly followed up with a deeper look into lower resolution performance [36]. This time the focus was classification into distinct mixtures of radioisotopes, finding accuracies comparable to those of algorithmic methods. A further study looked into the issue of gain shifts due to temperature or voltage transients [37]. Small shifts were added in post-processing, and demonstrated that by including drifted spectra in the training sets, the models could be made more generalised.

The Olmos *et al.* work was built on by several groups in the late 90's [38–41]. Keller *et al.* [38] focused on applying an ANN to waste handling that included some RIID for the gamma spectra. Interestingly the network was a very simple linear ANN that had no hidden layers; the input nodes were fully connected to the output nodes. Spectra rebinned to 512 channels were used as training data in an attempt to minimise the number of training parameters. However, this meant very good input data were necessary for training because the model became sensitive to any deviations from the ideal template.

Vigneron *et al.* [40] took a different approach to keep the input vector small. The 4096-channel HPGe spectra were reduced to just six input features with Principle Component Analysis (PCA), greatly cutting down the number of connections needed between layers. Training and test sets were highly specialised and had limited available data, possibly driving their advocation for bespoke solutions.

Continuing this group's work, Pilato *et al.* [42] applied the ANN to finding activities. A set of 12 predefined Regions Of Interest (ROI) were chosen, with the aggregate counts used as input features. Pre-processing involved removing background, centring on averages, normalisation, and a final projection onto orthogonal space with Singular Values Decomposition. A heavily involved pre-processing approach is seen to have carried forward from their predecessors, becoming almost commonplace in the following decade.

One immediate example is the ANN used by Yoshida *et al.* [43] from Japan, which also limits input dimensionality by extracting features manually. A peak finding algorithm is applied to smoothed data. If an energy found is within some tolerance of pre-selected dominant emissions, the feature is set to 1 rather than 0. The input vector therefore corresponds to whether or not a peak was found for each energy. The idea was that the ANN would learn input patterns of energies and relate them to the radioisotopes present. This approach worked well for calibration sources and uranium ore, however, the results stem from a very small training set with good statistics from a HPGe detector. It is therefore difficult to assess how much this approach relies on having good peak finding and calibration.

**Development 2002-2016**

After brief period of no publications, a paper from Kangas *et al.* [44] from 2008 stands out as significant. This was the first to look at RIID for PVT-based radiation portal monitors, seeking to reduce the number of nuisance alarms. The ANN was to be used in tandem with the typical catch-all gross-count method rather than being a replacement. The paper is also one of the first to use MCNP [45] and GADRAS [46] to simulate spectra that could augment a limited set of real data.

Peak search algorithms are often confused by overlapping peaks and unexpected features, so Chen *et al.* [47] took the PCA feature extraction idea from [40] and updated the transform. The closely related Karhunen-Loeve (KL) transform was used instead (See Ref. [47] for details) because it is reportedly better than the Fast Fourier Transform for finding weak signals buried in noise [48]. Applying the KL transform to spectra, they took the first 64 coefficients and used those as the input vector.

Building on the majority of previous work highlighted here, Medhat *et al.* [18] used an ANN to find relative activities of isotopes in natural sources. Only the most dominant contributing radionuclides were used for the identification. They followed the Yoshida *et al.* [43] example and took only the calibrated peak positions as features for input data. The optimised networks showed relatively good agreement with the measured values, though it

consistently overestimated activities.

Varley *et al.* [49] are one of the first UK groups seen to use neural networks for RIID. Training data were augmented with MCNPX [50] simulated data, though geometries and sources were greatly simplified. PCA was used to contend with variations in background radiation. While some predictions can be very uncertain, the ANN still typically outperformed conventional methods for estimating depth and activities of point sources.

Returning to PVT portal monitors, Felipe *et al.* [51] sought to classify types of material rather than any specific radioisotopes. The groups were defined as: Industrial, Special Nuclear Material (SNM), Medical, and NORM. Data are preprocessed when a gross-count alarm is triggered, again working in tandem with existing systems. This involves selecting intervals to cut down on the raw data size. The fully-connected ANN resulted in satisfactory results, being capable of classifying spectra into standardised groups.

In a similar vain to this, Bobin *et al.* [52] looked at well-type NaI based portal monitors. The aim was for real time processing for RIID and relative activity estimation. To achieve this, Bayes' rule [53] is applied to find radioisotope contributions to the overall spectra. A library of independent spectra was created using measured spectra, combining them in a few different ratios for testing. The results were promising but run times were longer than expected, varying significantly with spectrum complexity and NORM rates.

Up until this point, machine learning was mostly confined to specialised academic and commercial development. However, the next few years saw the beginning of an explosion in machine learning applications across a diverse range of research fields. This is crucially due to the accessibility of open source tools. The fairly steep learning curve is a small price to pay for these powerful libraries that would otherwise require a great deal of effort to implement from scratch; another limiting factor on the complexity of the early models following Olmos *et al* [19, 36, 37]. The physical sciences in particular are constantly performing pattern recognition in some form or another, so it is natural that a drive for collaboration with computer sciences has resulted in machine learning becoming a widespread tool. Recent increased interest in machine learning for RIID comes as no surprise [16, 17, 20, 21, 54].

### Recent publications post-2017

The perfect example of this accessibility is seen in a paper by Kamuda *et al.* [16], where Google's TensorFlow [55] is utilised. This is carried through to later papers [20, 54] where changes to model architectures became far quicker and easier to implement. What is new here is the focus on spectra containing relatively few counts ($10^3$-$10^5$ gross counts), where previous

work tended to have training data with good statistics. An impressive 32 radioisotope classes were trained for, though all training and test data were simulated with MCNP [45]. While perhaps not the best indicator of real world performance, using primarily simulated data allowed for a large training set of $10^5$ spectra. This is significantly more than any previous work. The core $10^3$ simulated spectra were augmented with random perturbations such as gain shifts. The paper found relatively good insensitivity to gain shifts due to it being included in the training data, as suggested back in 1992 [36].

In a follow-up paper the same methods are applied with the objective of calculating relative activities in NaI spectra [20]. For the built-in background profiles they switched to GADRAS [46] to simulate training data. This time the simulations were run long enough to ensure a 'perfect' case, and then combined in different ways with the inverse transform method [56] to create mixed-radioisotope spectra. With few exceptions, the single isotope ID performance was very good. Even with significant background contributions, the confusion matrix showed the majority of test data to be correctly identified over 80% of the time. However, ANN accuracy fell apart when training on data with different energy resolutions (7-10% at 662 keV simulated). Again though, all data used for training and testing were simulated, making it difficult to comment on real world performance. The trouble this ANN had with resolution changes, even under ideal simulated conditions, says a lot about generalisation challenges, and identifies it as another key area for further study.

In contrast to Kamuda *et al.* [20], Kim *et al.* [17] had a paper accepted a few months prior showing their ANN handling resolution differences fairly well for PVT spectra. Of course, the resolution of the two detector types is incomparable, but it is interesting to see them including several Gaussian Energy Broadening functions in the MCNP6 [45] simulations to diversify their training data. The training data were simulated, but real data were taken for testing with a few mixed into the training set. This became important, finding there to be an 18% accuracy drop when real data were excluded from the training set.

One inherent limitation of ANNs is that they treat every channel as an independent feature. In 2018, the first probes into the viability of a variant known as the Convolutional Neural Network (CNN) were seen. The details of these networks are presented in Section 2.3, but the key benefits are that not only is computational expense reduced, but the networks also consider neighbouring channels.

### CNN papers post-2018

The first to suggest using CNN for the broader RIID problem space was Kamuda *et al.* [54]. A well developed ANN from Ref. [16] in 2017 was compared with a basic CNN. All NaI spectra used throughout were simulated with GADRAS to allow them to investigate the 29 radioisotopes from ANSI recommendations [34]. The two networks were compared based on how well they did with basic RIID, then on variations in source distance and FWHM parameters. It was found that in this ideal case of purely simulated spectra, the CNN outperforms the ANN. However, when predicting simulated spectra that had real background the CNN did slightly worse. They suggested that ANN may be better for dealing with unknown background patterns. With varying stand-off distance, CNN generalises slightly better. For predicting spectra with 10% resolution at 662 keV after training on 7%, both performed poorly.

The following year, Liang *et al.* [21] also implemented a CNN for RIID purposes. Training data were simulated for NaI using MCNP to look at only four radioisotopes. Each spectrum contained $10^8$ events, with measured background randomly added to create a training set. A Hilbert transform was applied to create 32x32 images from spectra by projecting them onto a 2D plane (See Ref. [21] for details). Since none of the training data had any gain shift, a 20-channel shift was enough to cause identification errors. They went on to compare to an ANN that had a 128 feature vectors from a wavelet transform, along with a couple of other basic machine learning model types. In general, the neural network types are far more consistently accurate, with CNN slightly better than ANN in this particular comparison. However, it is difficult to judge here since there is almost no information given about the other models.

In parallel to this thesis, Daniel *et al.* [22] used GEANT to train a CNN to identify five calibration sources in a high resolution CdTe-based system. The proceedings show a similar approach where they investigated the viability of using purely simulated data to train models before testing on real data. They found that false identifications were mostly due to a small proportion of the sources. Primarily focused on dose rates, the system was able to achieve accuracies of 90% on spectra containing as low as 1000 counts from the source. All sets were calibrated, with minor gain shifts included. No other perturbations were included in the data through shielding or energy discrimination, with sources kept far enough so as not to have summation. Interestingly, a more modular approach was taken where each source had its own network for predictions.

### Machine learning summary

The last three decades have seen steady progress in applying neural network models to the RIID problem space. Paying attention to the origin of each group shows how ubiquitous RIID is among the international community. A diverse variety of methods are applied to HPGe, NaI, and PVT-based radiation portal monitors for RIID specific applications in security, decommissioning, and public health & safety.

Early work by Olmos *et al.* [19] opened up the possibility of using neural networks by applying an ANN to identify four basic radioisotopes in 1991. Since then, various techniques have been developed to scale this up to identify 32 radioisotopes [16] (though considerably fewer in other publications). Until 2018, the developed techniques unanimously revolved around reducing the number of input features given to the models. Unfortunately, ANNs still quickly become extremely complex and computationally expensive, significantly limiting the number of nodes and hidden layers that can be implemented. Several interesting approaches include Principle Component Analysis [40, 49] and the closely related K-L transform [43] for extracting underlying signals from background noise. Others rephrase the problem, preforming feature extraction manually [16, 19, 42, 47] and provide those as input data rather than the raw spectra.

Following preprocessing, many have made an effort to investigate deeper issues such as gain shifts, background contributions, and variations in energy resolution. Sensitivity to all of these is generally mitigated somewhat by including wider diversity in the training sets. Many have limited data sets however, so sensitivity is difficult to compare. Gain shifts are of course much more of a problem for input data based on ROI or peak search methods. It is only recently that such variations in detector resolution became a concern, with Refs. [54] and [20] looking into what now appears to be a problem area.

Creating more comprehensive datasets to combat real world differences has been a driving force behind the switch to primarily simulated data. This was always MCNP or GADRAS and at larger stand-off distances to negate summation, which they cannot reproduce well without manual intervention. The use of GEANT4 in this work (and recently that of Ref. [22]) provides the flexibility needed to investigate timing information that would otherwise be time consuming to include in the more restrictive frameworks like MCNP. An increase in papers using *only* simulated spectra for training and test data unfortunately leaves only speculation as to how well the models would perform under practical deployment.

What every paper up to 2018 has in common is the use of fully-connected ANN models in some form or another. With TensorFlow's release in late 2015, other network possibilities

opened up without too much effort on the part of the researcher. The use of Convolutional Neural Networks (CNN) was suggested in Ref. [54]. CNNs are better for many reasons, but one in particular addresses multiple issues highlighted throughout ANN usage. Rather than every channel being an independent feature, local channel information is also considered. By applying the techniques seen for ANNs, and of course developing new CNN specific techniques, it was an interesting challenge to see how far they could be taken. With so little work done with CNNs, it remains a very new research area that certainly merits more in-depth study beyond this thesis.

## 2.3 Convolutional Neural Networks

Machine learning now constitutes a huge field in computer science and is a vast topic. The following sections detail only the relevant concepts and techniques, set in the context of their use for RIID applications. As discussed, neural network based approaches have historically been relatively successful. The recently suggested variant known as the Convolutional Neural Network (CNN) became the focus of this research, so discussions here are confined to this model type.

### 2.3.1 General concepts

#### Models and problem definition

The more classical approaches described in Section 2.2.3, however complex, are always following sets of explicit instructions written by the developers. There is no scope for dynamically handling unexpected perturbations. Machine learning differs in that models learn generalised rules born of experience, training to recognise and analyse patterns in datasets without being explicitly programmed. For gamma spectrometry and RIID, this means providing diverse example spectra ('samples') from which to learn.

There are many different types of model for machine learning applications depending on the data available and the desired outcome. These may be loosely broken into 'supervised' and 'unsupervised' problems. For RIID, the task can be framed as that of a supervised problem, where the target outcomes are known. For example, if it is known which sources are present in each sample spectrum, the spectra may be appropriately labelled to provide a goal for the model while it learns. For the interested reader, there is an excellent review [14] of machine learning in physics that covers various model types.

Unsupervised learning is an interesting problem space where none of the target outcomes are known, so the model is left to infer what data should be clustered into groups by looking at similarities between variables. Unlike supervised learning, there are no predefined groups provided, and neural networks may also be used to tackle such problems.

Target outcomes for supervised learning are typically either regression or classification. The latter is the focus of this work, where the goal is to identify which sources are present in a spectrum. However, as discussed in Section 2.2.4, RIID models have also been applied to regression problems in the past [18, 42]. Classification problems may be summarised as the following types:

1. Binary
    (a) 1 source
    (b) Present (True) or Not present (False)
2. Multi-class, single-label
    (a) 1 source (label)
    (b) $N$ possibilities (classes)
3. Multi-class, multi-label
    (a) $k$ sources (labels)
    (b) $N$ possibilities (classes)

Each predefined group for classification is known as a 'class'. In RIID this corresponds to the full set of sources that may be present in a spectrum. It may be as simple as a binary outcome where there is only one source, with the two classes being 'present' or 'not present'. Slightly more complex is the case where there are $N$ classes to create a 'multi-class' problem, where the goal is to choose one-from-many. For example, if all spectra were to contain only one of the possible sources, a multi-class solution will associate each class with a unique source.

An RIID system that identifies only one of the possible sources as being present is fairly limited. As the aims of RIID discuss, it must be able to handle the presence of multiple sources. Including this in a single model therefore requires extension to a many-from-many problem. Target outcomes are further broken into 'labels', each corresponding to a class.

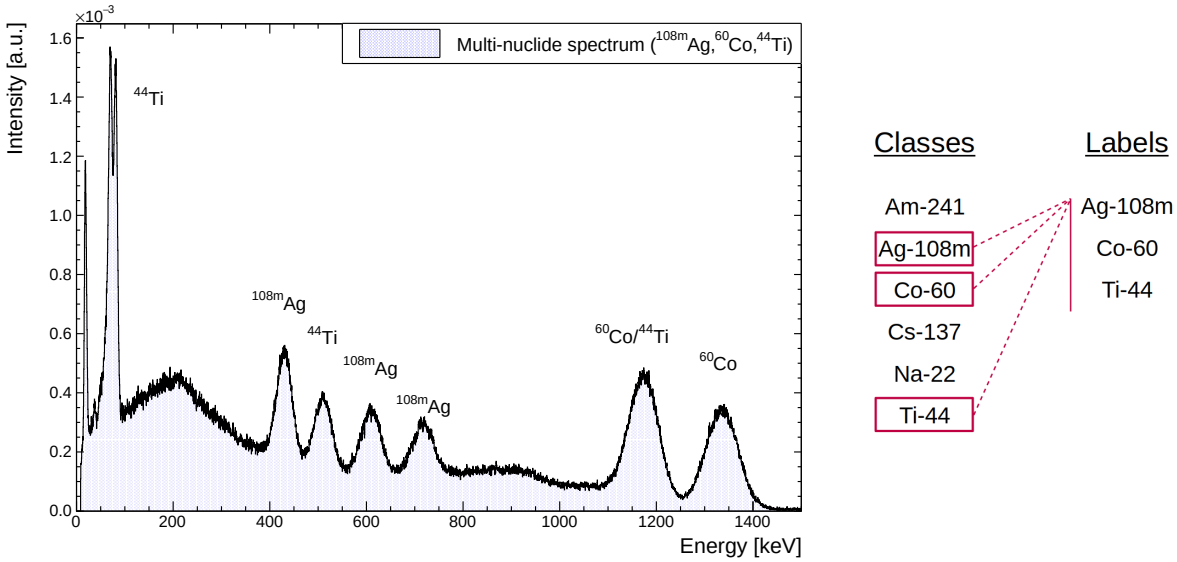



Figure 2.8: Diagram demonstrating class and label concepts. In this case, 'classes' are the possible sources that might be present in a spectrum, while 'labels' denote which of these are actually seen.

This concept is demonstrated in Figure 2.8. For example, if the classes are defined to be `[Am, Ag, Co, Cs, Na, Ti]`, then this spectrum may be labelled by content as `[Ag, Co, Ti]` if all three sources are present. The arbitrary combination of which sources are present can therefore frame RIID as a 'multi-class, multi-label' problem. As the focus of this research, all discussions are put into the context of this problem type.

### Training a model

With the problem defined and an appropriate model type chosen, it comes time to train the model. Any model requires data to be broken into three distinct sets: *training*, *test*, and *validation*. To be able to judge performance on an 'unseen' case, a test set is withheld from the model during training. Full datasets are typically split into training and test sets randomly while ensuring no sample is duplicated across both sets. Generally, the training set is further divided to include a selection of samples withheld for validation. This validation set is used during training to monitor for undesirable effects such as 'over-fitting', where the model ceases to generalise and simply starts to memorise the training spectra. Section 4.1 details how these sets are preprocessed for the purposes of this research. An example of the learning process is shown in Figure 2.9.

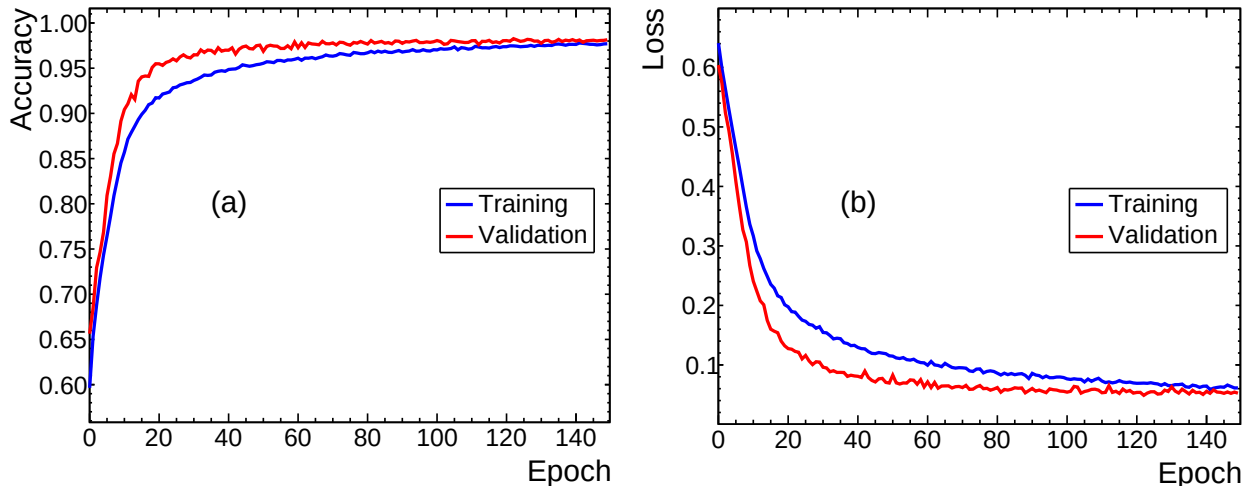



Figure 2.9: Example of relatively good learning, where the training and validation curves of accuracy (a) and loss (b) metrics begin to converge over several epochs. Through monitoring these curves, many problems with the optimisation or datasets are revealed.

Models are trained iteratively over a series of 'epochs', which is a single pass over all available training data. The learning process over an epoch is performing a numerical optimisation at its core, so while training there is some loss or cost function to be minimised. For example, this is much like minimising the $\chi^2$ when fitting a Gaussian to peaks in a spectrum. There is no universal loss function that works for everything, but all loss functions quantify a difference between the prediction and the true value. Like many models, losses are calculated here using the predictions made by a set of nodes much like the one shown in Figure 2.10.

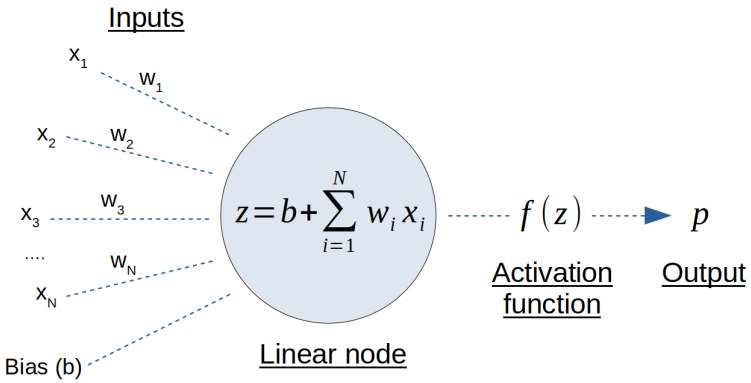



Figure 2.10: A single node of a fully-connected layer used to make a prediction. Every output ($x$) from the previous layer is connected with a trainable weight ($w$) and combined linearly. An activation function, in this case, encodes the probability of a source being present to a value between 0 and 1, with some offset defined by the bias ($b$). Validation samples follow optimisations using the training samples.

This node of the fully-connected final layer makes a prediction about a single class. The product of inputs of the previous layer ($\mathbf{x}$) and their associated weights ($\mathbf{w}$) are combined linearly before being passed through an activation function ($f(z)$). A bias ($b$) is just a constant included to shift the output so that the activation function may fit better. The loss ($l$) for this class is calculated from the difference between prediction ($p$) and the target outcome ($y$).

Scaling this up to a classification problem with many classes, the cross entropy loss ($\mathbf{L}$) serves this purpose. Passing a single spectrum through the model, the loss may be calculated as

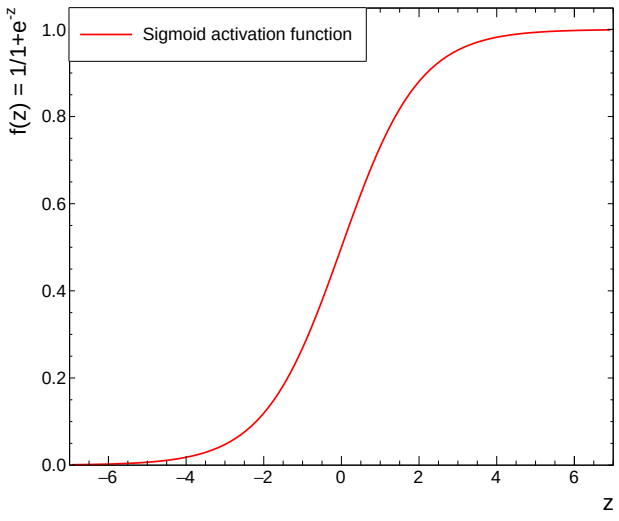
$$L(\mathbf{y}, \mathbf{p}) = -\mathbf{y}.\ln(\mathbf{p}) - (1 - \mathbf{y}).\ln(1 - \mathbf{p}), \tag{2.6}$$

where $\mathbf{y}$ is a vector of target label values, or which sources are present, and $\mathbf{p}$ is a vector of predictions[5].

The benefit of using logarithmic operations for classes is that $L \to \infty$ when wrong and $L \to 0$ when correct. However, if the probabilities were to be taken directly (i.e. $p = z$), there would be an issue when $p \to 0$ since $\ln(p) \to -\infty$. To counteract this instability, certain activation functions may be included in the process.

For the multi-class case, a Softmax [57] activation is fine as a categorical cross entropy function, where the predictions are scaled to (0-1) while summing to unity. For the multi-label case, where each class is independent and not mutually exclusive, a Sigmoid function is required instead, treating each class as (0-1) individually. This is defined as

$$f(z) = \frac{1}{1 + e^{-z}}. \tag{2.7}$$




[5]Specifically the probability of belonging to the positive case, where the source is present.

Using Sigmoid as the final node activation defines the Sigmoid cross entropy loss function, more commonly referred to as the 'binary cross entropy' loss.

$$L(\mathbf{y}, \mathbf{z}) = -\mathbf{y}.\ln(Sigmoid(\mathbf{z})) - (1-\mathbf{y}).\ln(1-Sigmoid(\mathbf{z})). \tag{2.8}$$

In this way, probabilities are bound as $0 < p < 1$, and the two halves of the equation tend toward zero depending on the result. Taking one label for example, $L \to 0$ as $z \to \infty$ if correct ($y = 1$, or positive) and $L \to +\infty$ if incorrect ($y = 0$, or negative). Overall, the sum of the loss vector is simply taken to provide a total error. The most important aspect of this is that a cross entropy loss will heavily penalise confident predictions that are actually incorrect.

Where the 'learning' comes in is that the derivative of the loss function is then used to make changes to weights and biases using 'backpropagation'. Values are traced backwards from prediction to input to determine how much of the loss is due to each part of the network, adjusting weights as necessary for the next iteration.

With the general topics of problem definition, datasets, and loss function covered, the details of specific model architectures may be discussed. Taking the neural network approach, an ANN may be set up as shown previously in Figure 2.7. The issue with these is that the number of weights, biases, and other trainable parameters rapidly increase with the number of nodes in each layer. For example, a 1024 channel spectrum fully-connected the 1363-node layer of the Kamuda *et. al.* [16] model already has ∼1.4 million basic trainable parameters at a minimum. Computational expense quickly increases out of control, which is one reason the variant known as the convolutional neural network was suggested as an alternative.

### 2.3.2 CNNs for RIID

Originally designed for feature recognition in images [58], the core of a CNN optimises successive 'filter' sets that are convolved with an input sample. CNN convolution layers train only a set of small (typically 3×3) filters.

As the example of Figure 2.11 shows, this is fundamentally very different to an ANN. The trainable parameters are now just values in a filter, coupled with a single weight and bias for connection to the next layer. These filters perform feature extraction, with context now considered by including information from neighbouring pixels. Applying the same operation to gamma spectra, where many important features are localised over channels, is very beneficial.

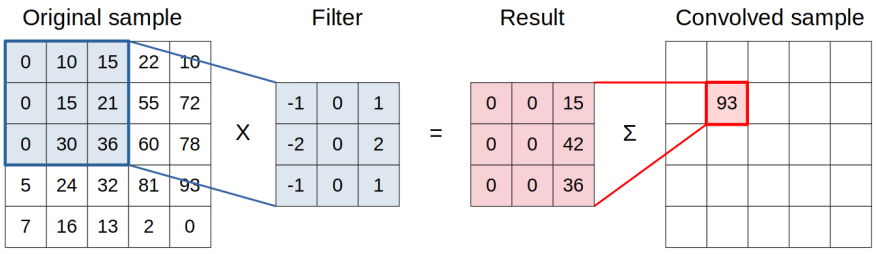



Figure 2.11: Applying a trainable filter to build up a feature map of new abstractions of the original.

By convolving an input spectrum with filters, new abstractions of features are created. More complex features are learnt by building upon multiple layers of convolutions. For example, the first layers generally end up performing basic operations such as edge detection. The idea of using a CNN for RIID allows responsibility of feature extraction to be placed on the model. This not only automates a lot of the manual preprocessing, but pattern recognition through these convolution filters will not rely on any calibration or standardised input (within limits). A filter that performs leading edge detection will work for the 662 keV $^{137}$Cs peak whether it is in channel 100 or channel 1000. For gamma spectra, the special 1D convolution operation may be used instead. To demonstrate several basic operations, Figure 2.12 shows a section of the full model used as a baseline in the results of Section 5.3.

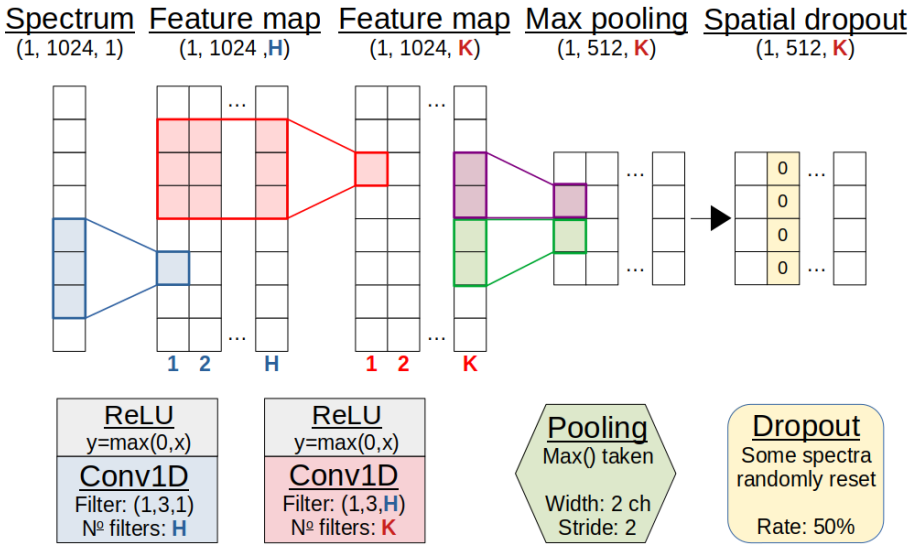



Figure 2.12: Representation of the convolution module. An array of intensities represents the spectrum. This trains first $H$ filters, then $K$ filters in two convolution layers to emphasise key features in the spectra. Pooling and dropout then reduce the dimensionality and combat over-fitting. For demonstration purposes, the shapes exclude a dimension for batch size.

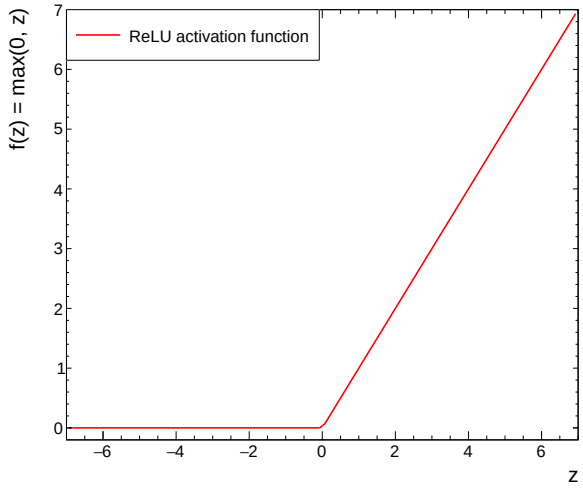
The module shown is connected to a raw 1024-channel input spectrum. Expanding the concept to multidimensional arrays (tensors), the shape of each layer is shown. The input spectrum has shape `(1, 1024, 1)` as `(height, width, depth)` to follow the image analogy. For an RGB image the depth would be 3, where each pixel was a combination of three colours. For spectra, much like greyscale, each channel has only one number: counts.

In this case, the first layer convolves each channel of the input spectrum with a three-channel wide filter. The result of sliding this filter over the spectrum is a new abstraction of the original. By applying several filters, the output tensor forms a 'feature map' which contains the result of different convolutions stacked together. This can be done multiple times if desired, as seen in the example.

Activation functions are not only used for encoding the predictions. Their primary purpose is to introduce non-linearity in networks. Without these, layers of a network are simply performing the dot product between input and weight tensors, which is just a linear operation that effectively makes it one layer deep. Since these networks learn by building on the previous layers, this renders the architecture relatively pointless.

A Rectified Linear Unit (ReLU) activation function is typically applied after the convolutions to introduce non-linearity between layers. This is much faster than legacy functions [59] and may be summarised as

$$f(z) = max(0, z). \quad (2.9)$$




After the convolution layers, the scale of the tensors has increased by a factor of $K$, or the number of filters. A pooling layer reduces the size of feature maps to reduce the number of trainable parameters and memory required. This operation applies to each spectrum in the feature map independently, taking the maximum of two adjacent channels. The pooling

kernel shown takes the maximum intensity of adjacent channel sets and has a 'stride' of two. Size and stride can be adjusted to have larger pools for broader sampling, or even adjusted to have overlapping pools. It is worth noting that there is a recent tendency to remove pooling layers since their inception [60], instead changing the stride of the convolution layers.

Pooling also summarises the features that are seen in the feature map. In this way, the next Conv1D operations are applied to these higher level and more abstract features, rather than caring about something that is precisely positioned. This is how the network builds on previous layers to learn new features that are progressively less sensitive to position.

The final important operation shown is that of spatial dropout. In ANNs, some of the layer outputs are randomly selected and 'dropped' by resetting them to zero to introduce noise into the output values of a layer. The idea is that this breaks up coincidental patterns that are not related to the inputs, preventing the network from memorising irrelevant patterns. In CNNs the structure demands a slightly modified approach, where entire spectra are randomly selected from the feature map to have all intensities reset to zero. This has a similar effect in reducing over-fitting, but also serves to promote independence between adjacent spectra in the map [61], which can end up strongly correlated.

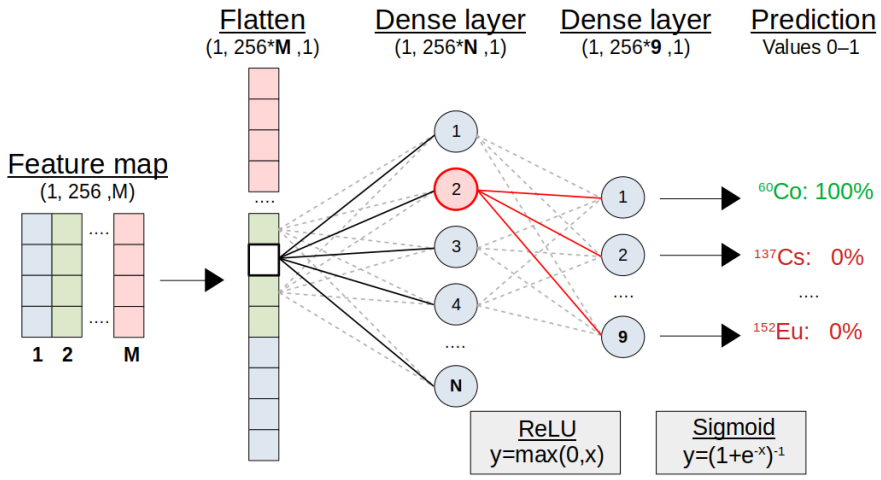



Figure 2.13: Representation of the classification module. The convolution module output is flattened into a vector. A small 'fully-connected' ANN makes final predictions of present sources, with a 0-1 probability through a Sigmoid activation.

Of course, the features extracted by the CNN are not a prediction by themselves. A CNN actually contains a small ANN as a part of its architecture, with feature maps provided to a final classification module through a fully-connected layer. An example of this is shown in Figure 2.13.

The feature map made from the final $M$ filters here must be flattened into a vector for compatibility, laying every feature map end-to-end. For example, four 256-channel feature maps become a vector of 1024 intensities. Each element is then fully-connected to the first ANN dense layer. While this is potentially a large number of elements, recall that the number of channels in each spectrum of the feature map has been reduced through pooling, and that fewer filters may be used for investigating patters between higher level features. The result is that a far smaller ANN may be used for classification, though these two layers still constitute the vast majority of trainable parameters.

The ANN can have any structure, though often a basic two-layer setup is used. Nodes of the final layer must equal the number of classes so that a final Sigmoid activation can provide the probability of each source being present in the input spectrum. This is of course to accommodate the binary cross entropy loss for the multi-class, multi-label problem.

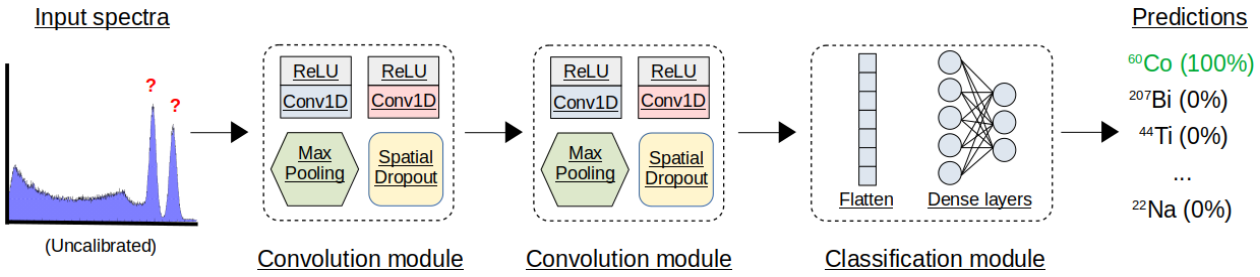



Figure 2.14: Simplified CNN structure. The basic sequential model consists of two modules, each performing 1D convolution operations. The resulting tensor is flattened into a vector to interface with a small, fully-connected ANN. The final output layer provides the probability of each source being present in a gamma spectrum.

Figure 2.14 demonstrates how all these elements may come together to build up a CNN for RIID tasks focused on classification. The key benefit is that this is extremely flexible. To change predictive goals, the only thing that needs to change is the ANN output. The features extracted in the early CNN layers will typically remain consistent across similar applications, so all that matters is what the ANN then interprets from those features. This lends itself to things like transfer learning and other advanced methods discussed in Chapter 7 to make the range of applications extremely diverse, despite the simplicity of the architecture.

So with a model providing some numbers to work with, the question of performance is raised. Methods of quantitatively evaluating how well a model performs on a given task are now required. These provide metrics to track improvements in model iterations, and enable direct comparisons between different models.

### 2.3.3 Performance evaluation

The evaluation of multi-label classification is best described by building upon binary concepts. Indeed, the loss function used even breaks down each class into a binary classification anyway. A final Sigmoid activation (Eq. 2.7) encodes the probability of each source being present in a spectrum. Typically, models are trained on a decision threshold of 0.5, with greater probabilities defining a 'positive' (1) classification. If the source is actually present, it is then correctly identified as a 'True' Positive (TP). Correspondingly, all other possible outcomes are shown in Table 2.1.

Table 2.1: Simple binary classification outcomes.

| | | Actual label | |
|---|---|---|---|
| | | 1 | 0 |
| Prediction | 1 | True Positive (TP) | False Positive (FP) |
| | 0 | False Negative (FN) | True Negative (TN) |

These are the basic metrics of any binary classification, with the grid sometimes known as a confusion matrix. The multi-label performance can be summarised as such by taking the average across all classes. Of course, the decision threshold may change and effect the performance of a model. To investigate this, True Positive Rate (TPR) and False Positive Rate (FPR) pairs may be plotted for each threshold, defined as,

$$TPR = \frac{TP}{TP + FN}, \quad and \quad FPR = \frac{FP}{FP + TN}. \tag{2.10}$$

The result is known as the Receiver Operator Characteristic (ROC), and it provides an excellent indication of how performance varies with threshold. The TPR, or 'recall', is a measure of ability to find all sources present, while FPR is the probability of a false alarm when put into the context of RIID. The rates hold varying importance depending on the application. In security, for example, it is perhaps better to accept an increase in FP rates if FN rates are reduced; better to have more nuisance alarms than allowing more illicit material to remain undetected.

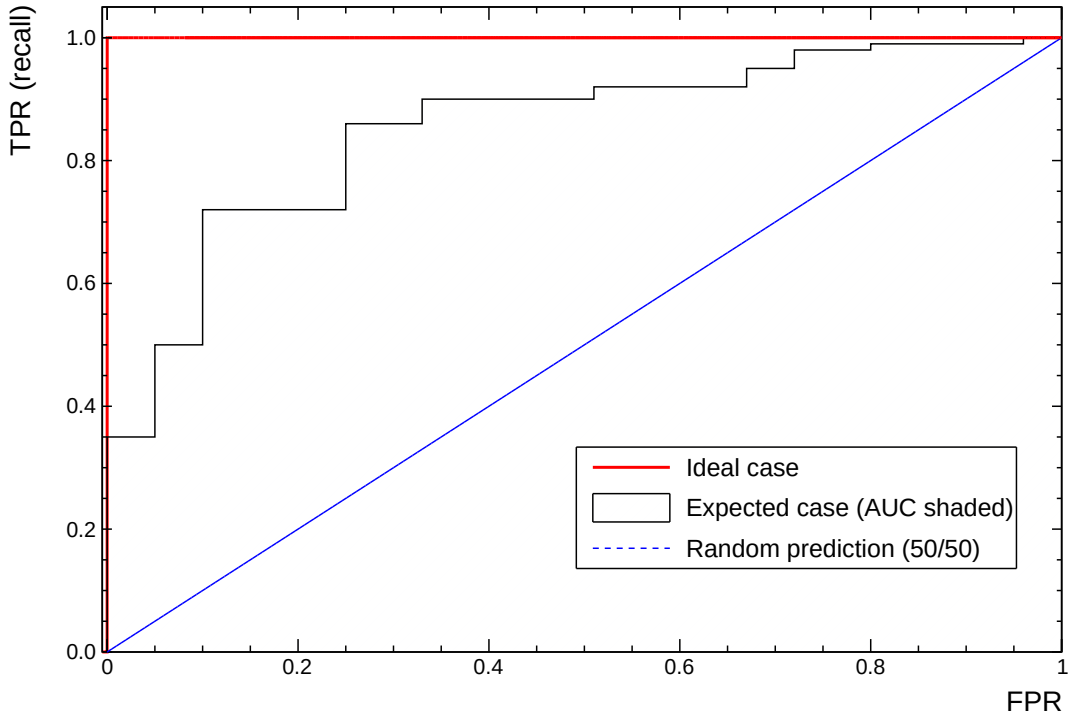



Figure 2.15: Example plot of Receiver Operator Characteristic (ROC). The ideal case tends toward (0,1) where all predictions are correct. A more realistic curve is also shown, staying above the $y = x$ boundary that demonstrates performance achieved by randomly assigning labels. The Area Under Curve (AUC) is taken as the shaded region.

The ROC curves of Figure 2.15 are extended to the multi-label case by considering each label individually, then together as an average. This information may be summarised in a single value by the Area Under Curve (AUC) as,

$$AUC = \int_{-\infty}^{\infty} TPR(t)FPR'(t)\ dt. \tag{2.11}$$

A closely related metric using these binary outcomes is 'precision', or how relevant each prediction is.

$$precision = \frac{TP}{TP + FP} \tag{2.12}$$

When paired with the TPR, a precision-recall (PR) curve may also be plotted to provide a measure of model skill. Figure 2.16 demonstrates exaggerated cases.

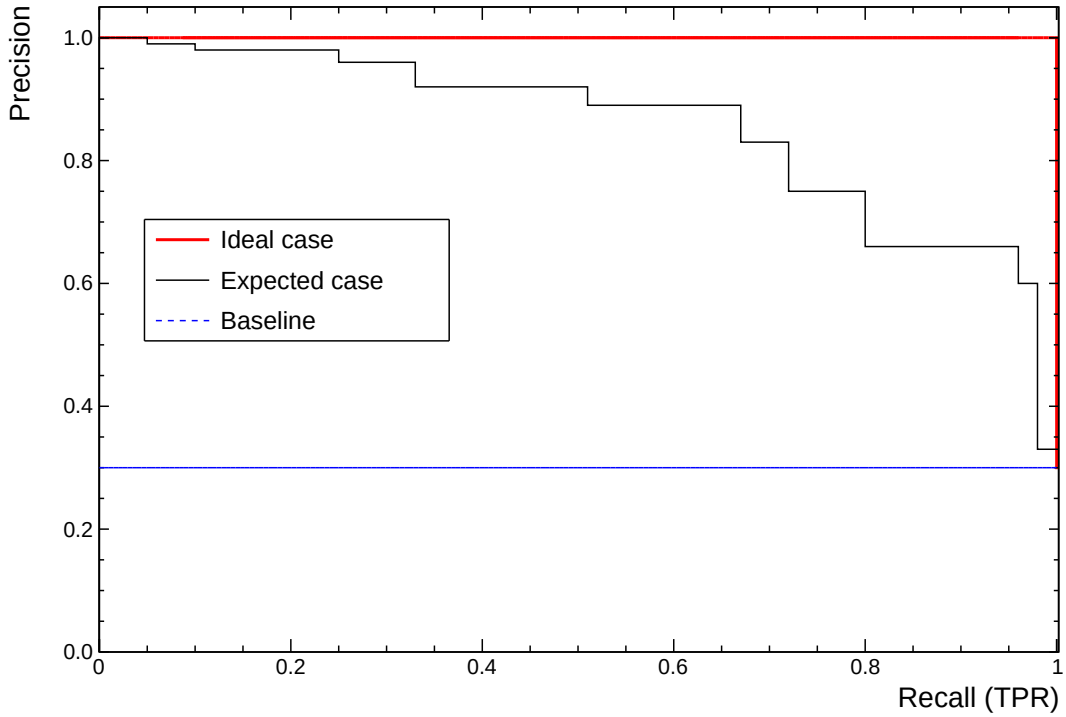



Figure 2.16: Example Precision-Recall (PR) curve demonstrating several cases. Ideal curves tend toward (1,0), while a baseline of randomly assigned predictions is defined by the fraction of positive samples in the dataset.

Both the PR and ROC curves may be applied to the multi-label classification problems with each class overlain to show contributions. This proves extremely useful for the RIID application because it is a clear visual indicator as to which sources are more challenging for the model.

Quantitatively the AUC and precision are good indicators, but when models are similar enough, the variation in performance due to minor effects like varying random seeds can obscure any differences. When every model shows a 95-99% precision it is difficult to tell if improvements are being made. The 'perfect' classification rates are, therefore, used as a complementary measure of RIID performance given the real world applications. Predictions are considered perfect when every possible source is correctly identified as either TP or TN for an individual spectrum. Performance drops if even one label is misclassified. With this, differences between similar models becomes much more exaggerated and very sensitive to predictions where a single source is consistently incorrect.

# Chapter 3

# Simulation of gamma spectra

## 3.1 Overview

A review of the literature in Section 2.2.4 shows a clear trend toward using simulated datasets when developing machine learning models for RIID. More recently, some have even gone so far as to use *only* simulated data for the speed and flexibility it offers [16, 54]. One drawback of doing so is that real world performance is somewhat speculative. The importance of understanding the scope of training and test data used for machine learning models is discussed in detail though Chapter 4. While experimental data were collected throughout this research, the advantages of using simulated spectra were also fully exploited to establish a rapid development environment. To this end, a generalised gamma spectrometry simulator was developed using radiation transport codes.

### 3.1.1 Radiation transport codes

There are two main challenges when it comes to fully describing a physical system: complexity at scale, and random elements. There are many systems that can be solved through theoretical or empirical models. However, issues always arise with scale, and any moderately large system quickly becomes extremely complex. In nuclear structure, simple systems containing a few nucleons may be described well, but add too many nucleons and finding exact solutions becomes impractical.

Random elements may then combine with this complexity to create near-impossible problems. Physics problems especially must often describe inherently stochastic processes. Radioactive decay is a prime example. It is impossible to predict when any specific nucleus will decay, but an informed prediction of the probability of decay can always be made based on

the underlying statistical distribution. The idea of breaking a problem down into manageable steps and sampling from known distributions is the foundation of transport methods.

For simulating radiation, all particles involved are transported individually. Note that 'particles' in this context encompasses anything moved through the geometry, including nuclei and photons. Interactions with matter are randomly sampled using the appropriate distributions at every key step. For example, a radioactive source may randomly decay at some time taken from an exponential distribution. A photon is then emitted in a direction sampled from an angular distribution, moving through a material with some probability of interacting. If it does interact, there are different interaction mechanisms, so one is randomly chosen with weights based on factors like the photon energy. This process continues until the photon disappears or leaves the simulation geometry.

Overall, the more individual particles that are transported on a micro scale, the better the macro approximation of the result. The advent of these methods is attributed to Los Alamos National Laboratory (LANL) in the 1940's, developed alongside computational techniques and technology to form the Monte Carlo N-Particle (MCNP) [45] code so widely used today. In physics there are now many such codes for general radiation transport, with GEANT [15], OpenMC [62], Serpent [63], and GADRAS [46] typically used in published work.

### 3.1.2 Simulation aims

The choice of radiation transport code is dependent on the task at hand. In this work, a simulator was required to accurately reproduce gamma spectra. This includes modelling of:

- Gamma interactions at all levels (photoelectric effect, Compton scattering, pair production)
- X-ray fluorescence/escape peaks
- True coincidence summation peaks
- Nuclides in arbitrary excitation states
- Multi-isotope sources

Originally, the core of the simulator was MCNP for version 6.1's incorporation of photon models that were developed for MCNPX [50]. However, the famously rigid interface to the MCNP black box proved unsuitable for using timing information without requesting access to the source. Several codes were also written to automate certain input fields such as the

source definition, where every emission and associated relative intensity must be explicitly defined by the user.

The GEANT4 toolkit was later chosen instead to provide the flexibility needed, with the open source code encouraging customisation at every level of radiation transport. The toolkit contains powerful fundamentals to be built on, with any higher level functionality left as an exercise to the user.

### 3.1.3 Geant4 basic concepts

Having improved the existing FORTRAN based GEANT3, a 1994 formal proposal (RD44 [64]) resulted in the development of a C++ based successor. The significant change to C++ allowed developers to exploit more advanced software-engineering techniques and object-oriented technology. All simulations here use GEANT4 version 10.6 and are compiled with CMAKE. For reference, the relevant nomenclature and key concepts are now described.

#### Particle propagation

Once some initial 'primary' particles and an environment are defined, the radiation may be transported through the simulation geometry. This is driven forward by the lowest level of iteration, known as a `Step`. Each step is composed of a `pre-step point` and a `post-step point` (Fig. 3.1).

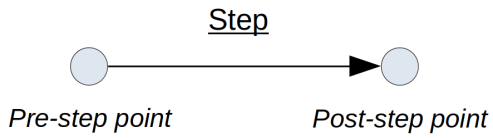



Figure 3.1: The `Step` is made up of pre- and post-step points.

A `Track` is then made up of many steps, with each track considered in turn and propagated through the geometry (Fig. 3.2).

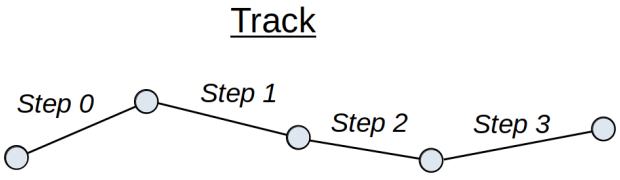



Figure 3.2: A `Track` is made up of multiple linked steps.

All tracks are collected under a single `Event`, which is the level at which initial conditions are defined.

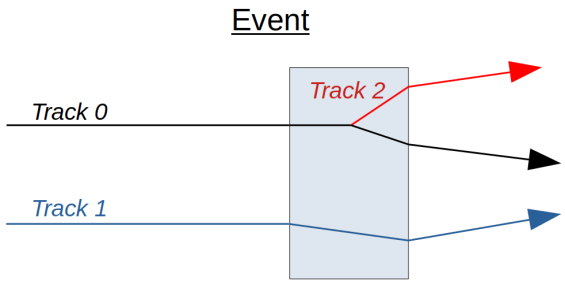



Figure 3.3: The `Event` contains a track for every transported particle, with a new track defined for any secondary particles created.

In the event shown in Figure 3.3 there are two primary particles, assigned to Tracks 0 and 1. If these tracks happened to interact and create secondary particles, new track objects are defined (Track 2). Any secondary tracks are added to the list and dealt with once the original is propagated through to completion. Finally, the `Run` collects all events together as the highest level of iteration.

All of these are C++ objects that contain useful information. For example, the energy deposited is held in the step object, with times and positions held in associated step point objects. Every collection of objects is deleted at the end of its parent iteration however, so there are certain data that must be collected before that happens. To illustrate, all steps are deleted at the end of the track, so recording step lengths would need to be done on a step-by-step basis.

To recap, the `Run` locks in the simulation models, geometry, and parameters and is made up of many `Events`. Each event is retrieving a primary set of particles that are to be propagated through the simulation geometry by a series of `Tracks`. These tracks and those of any secondary particles are driven forward on a step-by-step basis in the lowest iterative unit of the simulation, the `Step`.

Understanding this chain of Run→Event→Track→Step and keeping it in mind allows an intuitive knowledge of the various classes, their order, and how they interact. Computational expense increases further down this chain, toward the step propagation. It is therefore a good idea to retrieve and process information as high up the chain as possible.

### Core structure

Every GEANT4 simulation consists of a main function that sets up a *Run Manager*. As the name suggests, this handles the executions of each class and ensures they are in the correct order. The main function also initialises three mandatory classes:

1. *Detector construction*: Defines the full simulation geometry, along with any 'sensitive' regions.

2. *Action initialisation*: Defines the mandatory *Primary Generator Action* sub-class and all other *User Action* classes.

3. *Physics lists*: Defines all particles and physics processes to be included in the simulation. This is the heart of the simulation, determining how every particle should interact with the simulated geometry.

The primary generator action defines all initial particles to be propagated at the beginning of an event. User action classes are simply hooks for the kernel, permitting any additional code to be included for custom functionality. The User Actions relevant to this simulator are the *Run Action* and *Tracking Action*, but almost all boil down to:

'at the beginning/end of a Run/Event/Track/Step, do something'

Collection of relevant data may be carried out during any of these stages, but the use of *Sensitive Detectors* was more appropriate. Code written for a sensitive detector will only run when a step enters volumes designated as being 'sensitive', such as the active region of a crystal for example. This has several advantages, primarily convenience, customisability, and modularity within complex geometries. Any data deemed relevant is stored in a *Hit* object to be used later. Beyond this there are custom classes written for handling timing information and multi-isotope sources. The vast majority of these data are used to fill ROOT histograms set up at the start of the run action.

Note that the physics models and libraries defined in *Physics Lists* can influence the result, and must be chosen with care. GEANT is primarily a high energy physics code, with the 'standard' packages best covering ranges up to 100 TeV with approximations and assumptions made below 1 MeV. The regions of interest here are below 3 MeV, so the 'low-energy' packages were used to include better data that also include atomic effects down to eV range. The Penelope data [65] were chosen over Livermore due to great care dedicated to the low energy regions and the inclusion of positrons. Both are mixed datasets of analytical, parametrised, and experimental data.

## 3.2 Simulation techniques

### 3.2.1 Detector models

The focus of this research is on generalisation rather than specialisation. Several models were therefore created for the most ubiquitous detector types, which include commercially available HPGe and NaI detectors.

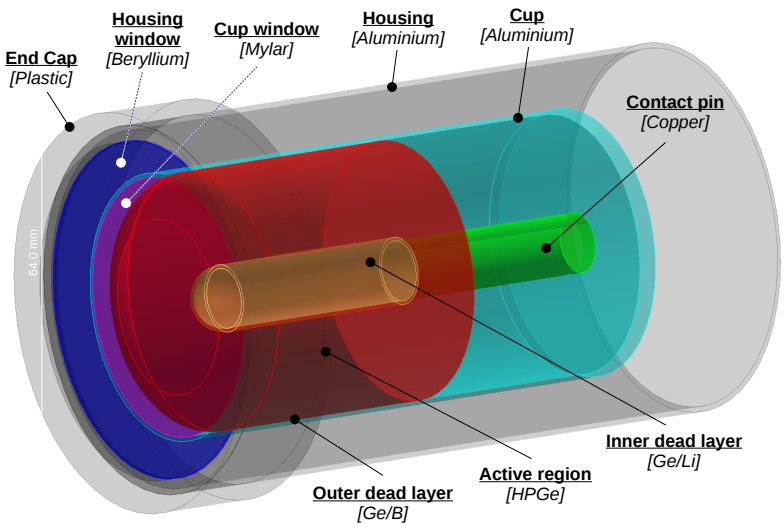



Figure 3.4: Model of the HPGe detector used.

The HPGe model in Figure 3.4 was used for all preliminary validation of the simulator. The superior energy resolution allows more subtle effects to be verified against real data. With poor resolution detectors such as NaI, these features are often obscured in gamma spectra.

A 3"×3" NaI detector model is also shown in Figure 3.5. Note that while the preamplifier and PMT are excluded, the material closest to the crystal is modelled in detail for more realistic behaviour at lower energies. For example, this includes attenuation of X-rays through the aluminium housing and minor backscatter effects. The same is true of the HPGe model, where there is a lot more material surrounding the active region.

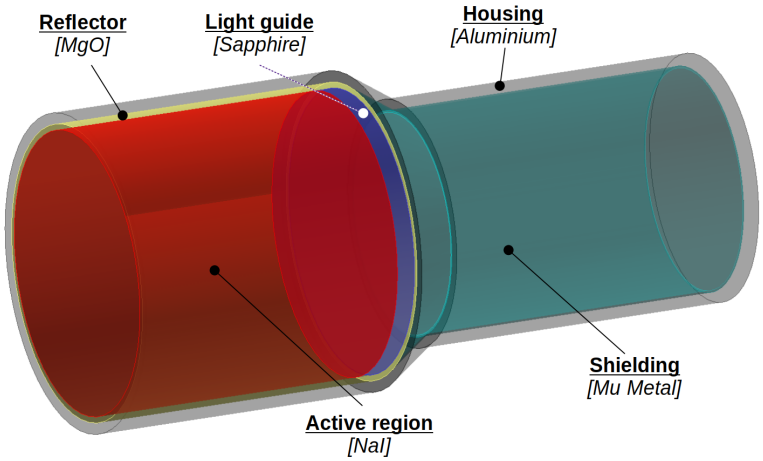



Figure 3.5: Model of one of the NaI detectors used.

### HPGe details

The geometry base for the HPGe model is that of an Ortec PopTop HPGe, model code GMX-10190. The manufacturer diagram kindly provided by Ortec specifically mentions the detector used, but this is taken to be representative of all detectors of the same model. The crystal volume is the only non-trivial part of the geometry in regard to the number of constituent components, but much like an MCNP model it can be built up from simple shapes.

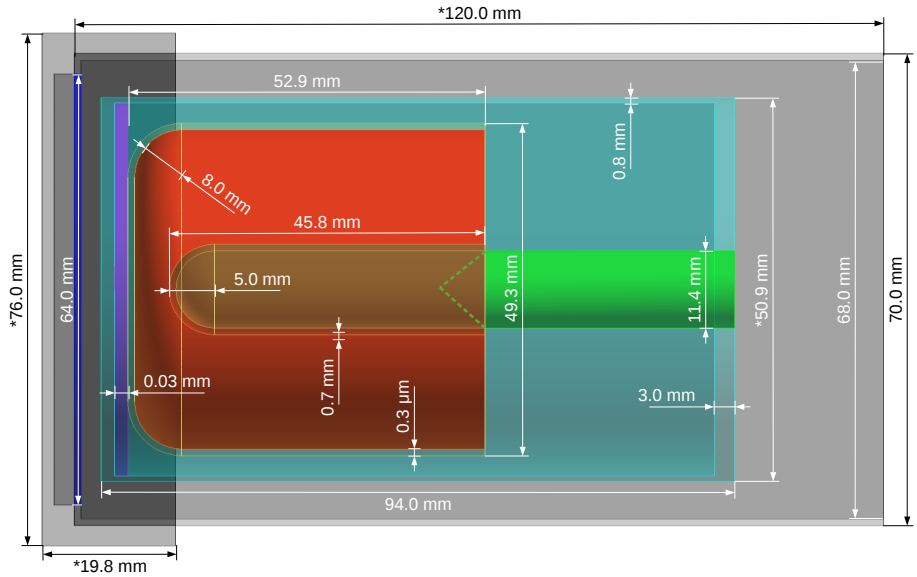



Figure 3.6: HPGe detector dimensions used for the simulated model.

The full crystal required seven cylinders, two tori, and two spheres to define all the rounded edges of the 'Bulletized' shape. A cryostat cup is kept under vacuum to inhibit the any thermal conductivity with the surrounding air. The Ge/Li inner dead layer takes 0.7 mm off the potential active region and is considered to be pure natural Ge. The same is true of the 0.3 $\mu$m Ge/B outer dead layer. While extremely thin, the outer dead layer is still included in the simulation geometry in case energies below $\sim$25 keV became important. The only volume scored (i.e. in chich particles are explicitly processed and recorded) in terms of energy deposition is the active region of the crystal, shown in red in these figures.

The crystal is mounted within an aluminium cryostat cup. The inner cup diameter is assumed to be flush with the crystal surface. Note that this is one of several free parameters that must either be estimated or physically measured, but the majority of dimensions are accurate to the manufacturer specifications. The corresponding physical geometry of Figure 3.4 is shown in Figure 3.6 with the main dimensions labelled.

The cup thickness is 0.8 mm except at the crystal face, which is 0.03 mm of aluminium re-enforced with a 0.03 mm mylar layer. This is to minimise attenuation of incident radiation while remaining strong enough to withstand the pressure differential between atmosphere and the perpetual vacuum conditions within.

Surrounding the cryostat cup is the detector housing. The length is arbitrary with the condition that it must extend beyond the 94 mm cryostat cup and leave a 3 mm air gap between the cup face and beryllium window of the housing. The housing is also aluminium and specified to be 1 mm thick, while the beryllium window is 0.5 mm thick.

Attached to the end of the detector housing is a plastic end cap. Since this is omitted from the specifications the dimensions come from physical measurements with a digital micrometer. From the experimental side, the inclusion of the cap has a very minor effect on the observed amount of backscatter below 60 keV for the $^{241}$Am spectrum.

### 3.2.2 Reference data

To build the simulator and ensure the various physical processes are being properly represented in the simulated spectra, corresponding real data were collected experimentally. These reference data were taken using the detector specified in the manufacturer specifications, inviting the assumption that the geometry is similar enough for meaningful comparison to simulation.

A NHQ 214M power supply provided a $-2499$ V bias with no variation seen over a 24 hour period. The signal was amplified with an Ortec 571 amplifier, using a 2 $\mu$s shaping time with

a manually adjusted pole-zero to ensure signal quality. The signal is digitised with an Ortec Easy-MCA-8K Multi-Channel Analyser (MCA) for use with the Maestro[1] software to collect spectra.

A general purpose calibration was created to be used as the default, containing data from the available sources: $^{241}$Am, $^{108m}$Ag, $^{133}$Ba, $^{207}$Bi, $^{60}$Co, $^{137}$Cs, $^{152}$Eu, $^{22}$Na, and $^{44}$Ti. A 12 hour background spectrum was also recorded with distinct peaks used for completeness in Figure 3.7. This background was split into a series of two hour acquisitions to ensure no gain drifts had occurred before summing.

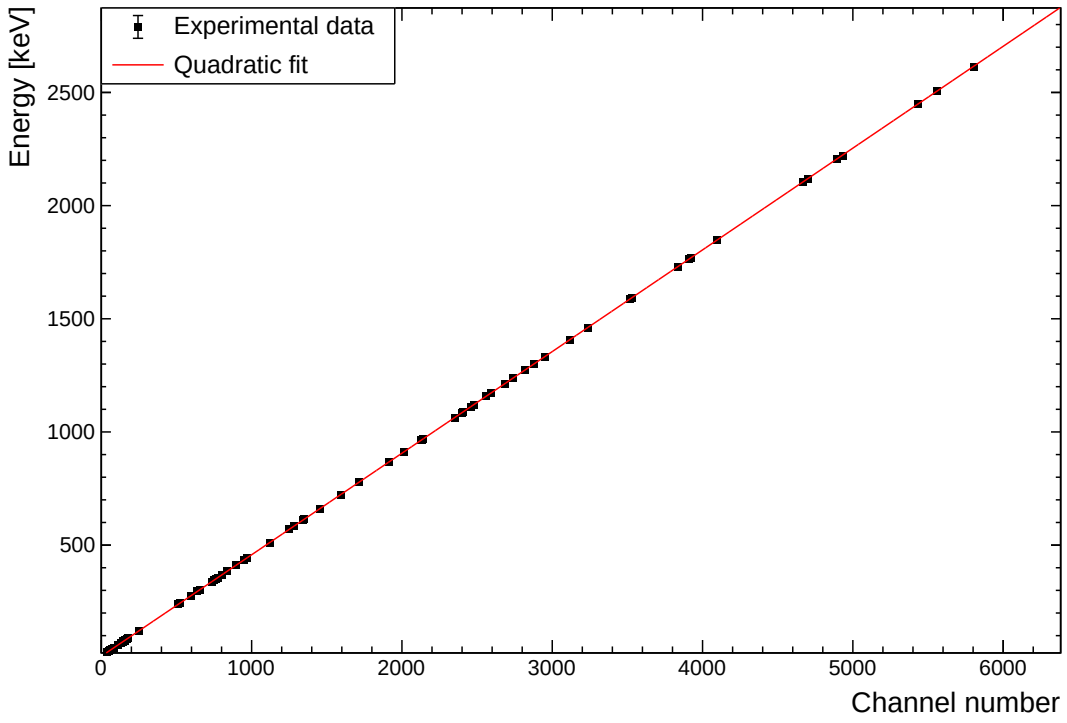



Figure 3.7: General purpose calibration curve to be used for the reference set of HPGe gamma spectra.

This calibration was applied to all spectra after the data were extracted form the Maestro *.Spe* format and converted to a suitable ROOT format. The appropriately scaled background was subtracted from the reference spectra. Using a custom N-Gaussian fitting ROOT macro, the peaks of interest were fit with a linear background component. The calibration covers the 3 MeV range well, mostly thanks to the abundance of significant transitions in the $^{152}$Eu decay scheme and environmental background data. Of course, the NaI calibration contains

[1]Version 7.01, licensed to The University of Birmingham, AMT Inc, copyright 2012

fewer data due to poorly resolved peaks. All peaks were refit once calibrated to obtain initial Gaussian Energy Broadening (GEB), which is one of the core simulation challenges.

### 3.2.3 Core challenges

#### Gaussian energy broadening

The full energy peak is arguably the most important feature in an energy spectrum. Ideally, this would be a delta function at distinct energies. The reality of physically measuring energy deposition introduces uncertainty and degrades the resolution. Photopeaks will contain contributions from charge collection statistics, electronic noise, and even drifts in operating parameters while an experiment is running. The shape of the peaks generally conforms to a Gaussian as described by Poisson counting statistics [25, 27], but the detail varies between detector types. To reproduce the energy spectra seen in experimental data, Gaussian Energy Broadening (GEB) must be incorporated into the simulation. This is implemented using a sensitive detector.

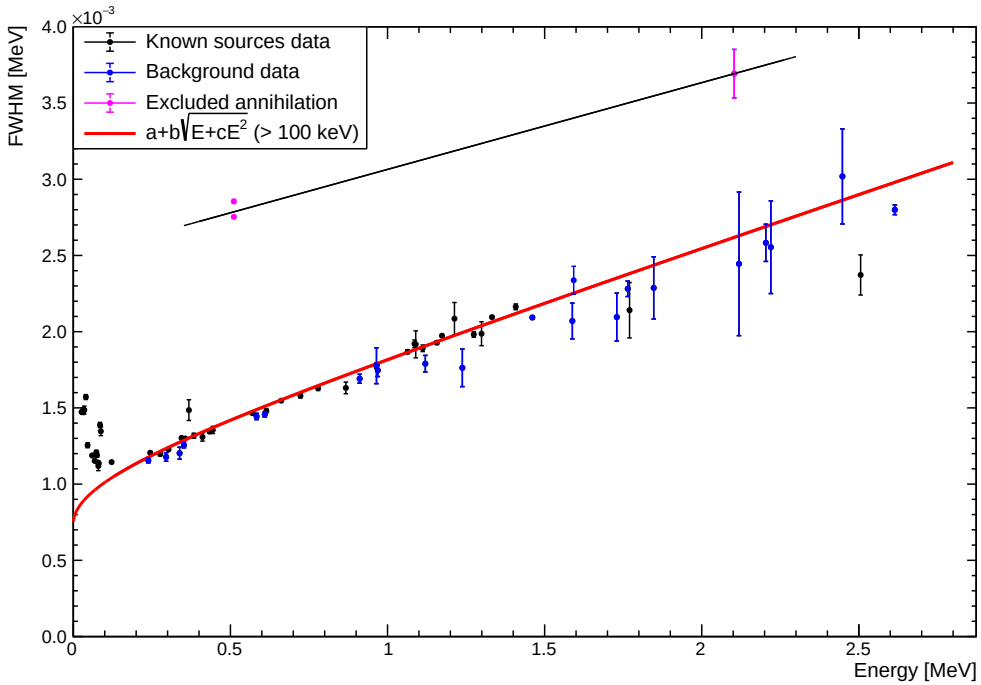



Figure 3.8: FWHM calibration for use as Gaussian energy broadening parameters. Note the divergence at the low energies (<100 keV) from multiple contributions that appear to be single peaks given the detector resolution, and the influence of annihilation photons.

At the start of every event a sensitive detector is initialised. If a step is made within the volume assigned to be 'sensitive', the stepping manager invokes the `ProcessHits()` method to record the relevant data. In this case it will only run when there is a track that steps through the crystal volume, constructing a series of `G4VHit` objects to hold the information. These hits are stored together in a *Hit Collection*, which is simply a vector that gets filled over each event. To construct a naïve picture of energy deposition, the energy lost with each step in the crystal may be summed.

At the end of the event (which is high on the simulation's iterative chain) the hit collection is looped over, and the total energy deposited per track sent to an exact energy histogram. For the realistic detector response with GEB, this total is also sent to a custom function that samples from a Gaussian distribution,

$$f(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{(x-E)^2}{2\sigma^2}}. \tag{3.1}$$

This emulates the expected Gaussian energy broadening. Here the the expectation value ($E$) is the exact energy, with the width ($\sigma = fwhm/2\sqrt{\ln(2)}$) coming from the approximation,

$$fwhm = a + b\sqrt{E + cE^2}. \tag{3.2}$$

There are many random number generators available. The `CLHEP::RandGauss` generator was chosen from the *Randomize* general header for convenience. Once the sampled value is returned, it is recorded in another energy histogram. Note that the parameters $a$, $b$, and $c$ are experimentally determined. Using the reference data, a good range of energies can be used for this width calibration. The data in 3.1 are the result of fitting Equation 3.2.

Table 3.1: GEB parameters used for simulations.

| Detector type | a | b | c |
|---|---|---|---|
| HPGe | $7.4(2)\times10^{-4}$ | $8.3(4)\times10^{-4}$ | $6.9(10)\times10^{-1}$ |
| NaI | $-1.2(1)\times10^{-2}$ | $6.9(1)\times10^{-2}$ | $-5.6(9)\times10^{-2}$ |

The justification for using Equation 3.2 follows from considering the signal chain and making practical assumptions. The full uncertainty will be a combination of the intrinsic width ($\omega_I$), information carrier production ($\omega_P$), incomplete charge collection ($\omega_C$), and electronic noise ($\omega_E$) [27]. These are combined in quadrature to give,

$$(\omega_{Total})^2 = (\omega_{Intrinsic})^2 + (\omega_{Production})^2 + (\omega_{Collection})^2 + (\omega_{Electronic})^2. \tag{3.3}$$

The approximation of Equation 3.2 is a more complete description from Hurtado *et al.* [66], and is used in MCNP to the same effect. Earlier suggestions include $\sqrt{a^2 + b^2 E}$ [67, 68] which is simplified to the $a + b\sqrt{E}$ that Genie2000 [69] uses. Note that these only describe the fit to full energy peaks that do not involve annihilation photons, which is another challenge.

### Annihilation photons

As a result of $\beta^+$ decay, a positron is emitted from the nucleus. It will rapidly slow to almost zero kinetic energy through losses via Coulomb interactions, and will readily annihilate if in the vicinity of an electron. As a consequence of the annihilation, two photons are emitted back-to-back with a total energy of 1.022 MeV in the centre of mass frame. The photons will have almost 511 keV each. However, because the $\beta^+$ has some residual kinetic energy and the (presumably) bound $e^-$ has orbital momentum, there remains a net momentum that must be conserved. This results in the two photons having $511+\delta$ keV and $511-\delta$ keV. Annihilation peaks therefore appear broader than expected in the real spectra as shown in Figure 3.9.

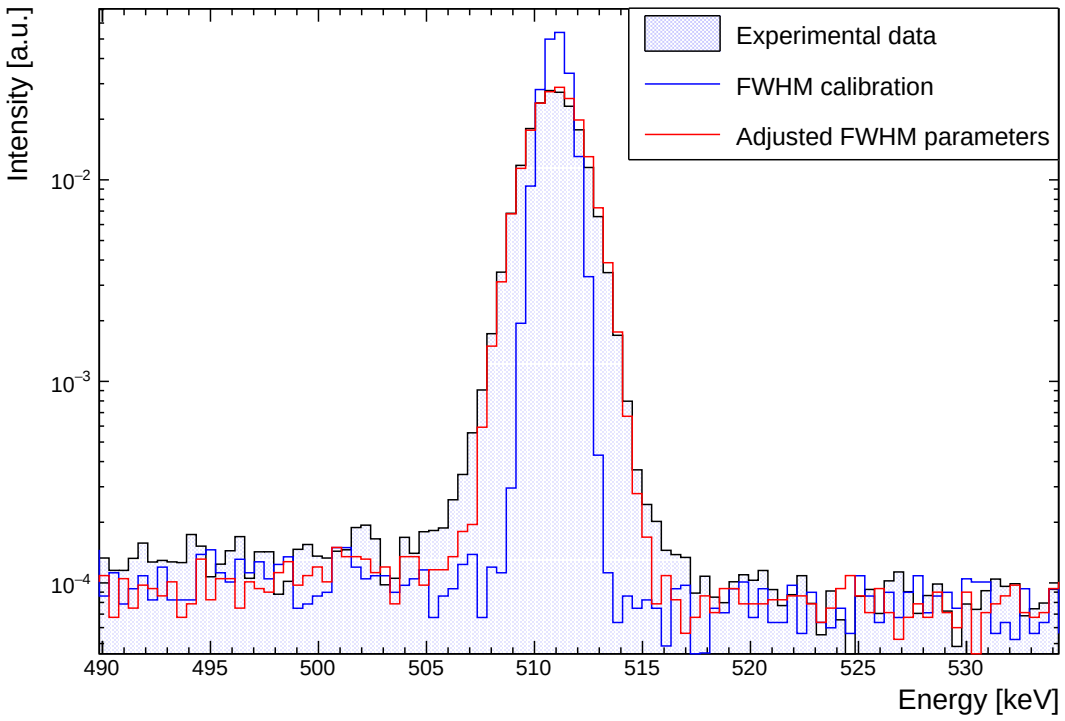



Figure 3.9: Correcting for the broadening of the annihilation peak in $^{22}$Na due to a residual net momentum of $\beta^+$ and electron.

The solution is to provide a second parameter set for peaks that involve annihilation photons (i.e. the full 511 keV and any first escape peaks). For the purposes of this simulator, special consideration is given to the 511 keV full energy peak in particular due to its prevalence in many gamma spectra. An average of several measured values is currently used instead of the GEB parameters when this specific energy is deposited in the active region. While less of a concern for the first escape peaks over the current ranges of interest, these will certainly become more relevant for full energy peaks above 2 MeV. This can be implemented by flagging annihilation tracks, and sampling the energy from an alternative parameter set given by the broader trend shown in Figure 3.8.

### Decay chains

When dealing with a radioactive nuclide, GEANT4 will follow the decay chain for as long as no stable isotope is reached. This can create unrealistic spectra given typical experimental time scales. Any lifetimes must be considered prior to initialising any secondary tracks, with chains limited based on a user supplied half-life.

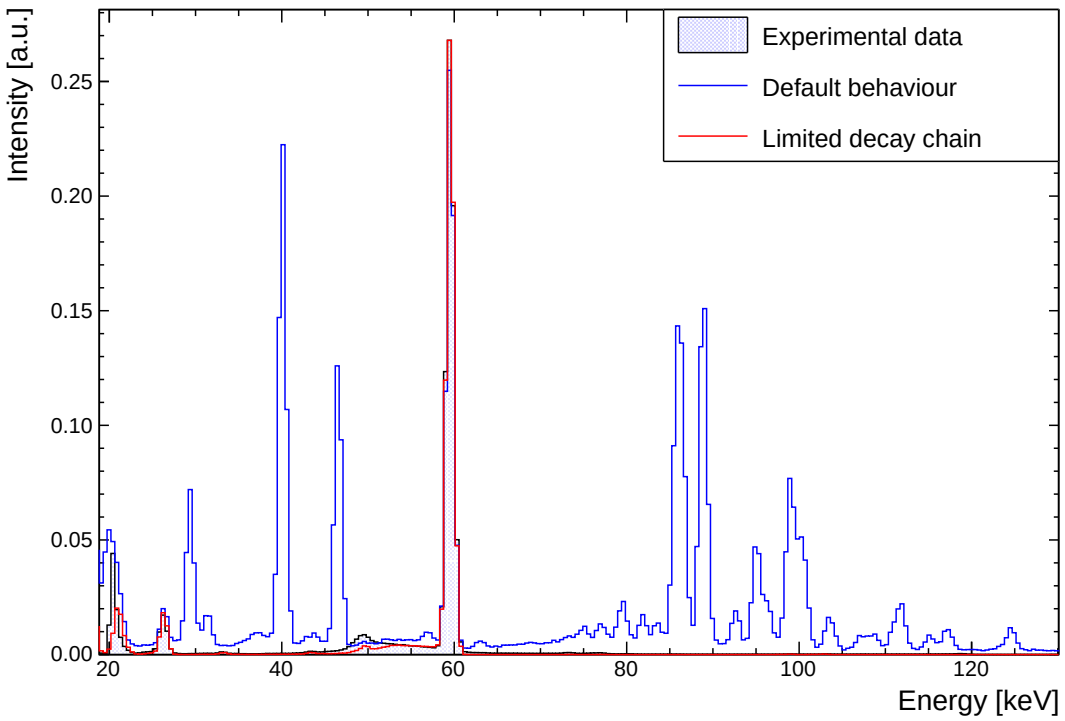



Figure 3.10: An $^{241}$Am simulation both with and without the decay chain limit. These are compared to real data (black line, light blue fill) for reference.

A tracking action class is mostly used to collect a lot of event-relevant information, but it is also a good level to do things like limiting long decay chains. The $^{241}$Am source is a perfect example of the issue. It has a 435-year half-life, alpha decaying into $^{237}$Np which itself has a ∼2 million year half-life. Figure 3.10 demonstrates the result when GEANT is allowed to follow the decay chain of the unstable $^{237}$Np, resulting in many photopeaks that are not seen over realistic collection times.

It was found that the track is the highest iterative step where some conditions can be enforced based on lifetimes. There are no methods in any of the physics lists that allow something similar to be set, so a function was needed to manually kill or continue tracks. This may be done by overriding the track status enumerators of the *G4TrackStatus* header. Note that this can also be done under the stacking action, but there was no practical reason for involving another user class. Any new track produced may now be considered through the process outlined in Figure 3.11.

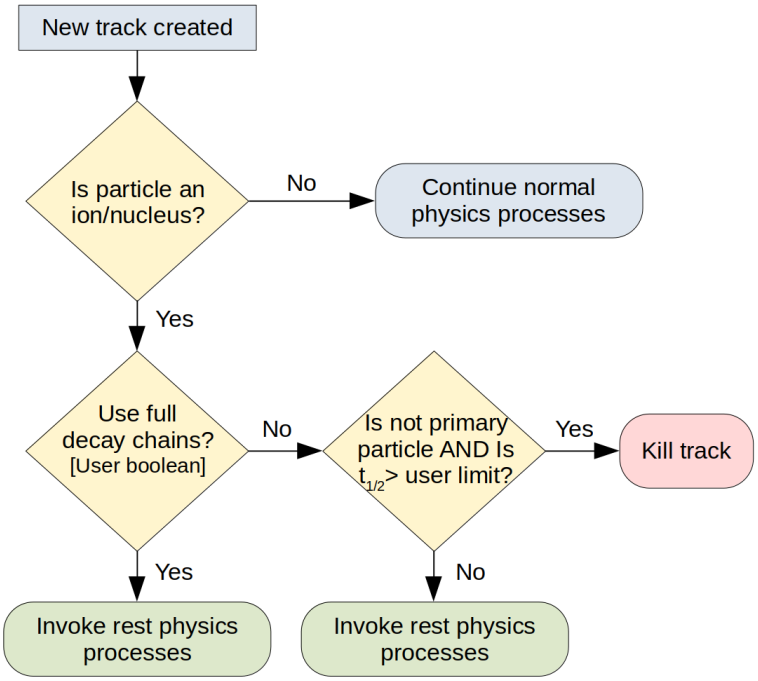



Figure 3.11: Process for limiting long decay chains by looking at the particle lifetimes of a newly created track object.

While lifetime is used as a condition, it is calculated from a user supplied half-life out of preference. A default half-life of $1{\times}10^6$ years is included in case a limit is wanted but not specified. The result of toggling these conditions is seen with the limited decay chain spectrum (red) shown in Figure 3.10.

### X-ray escape

The absence of expected escape peaks are particularly noticeable in the $^{241}$Am and $^{44}$Ti spectra due to prominent low energy emissions. The escape peak occurs when X-rays produced near the crystal surface and leave the active region without interacting. This falls under the atomic relaxation model in GEANT, which defines a threshold or 'range cut' for producing these X-rays. Any secondary particles with insufficient energy to travel at least the range cut value will not be produced. By default this is set to 1 mm for all particle production throughout the simulation geometry.

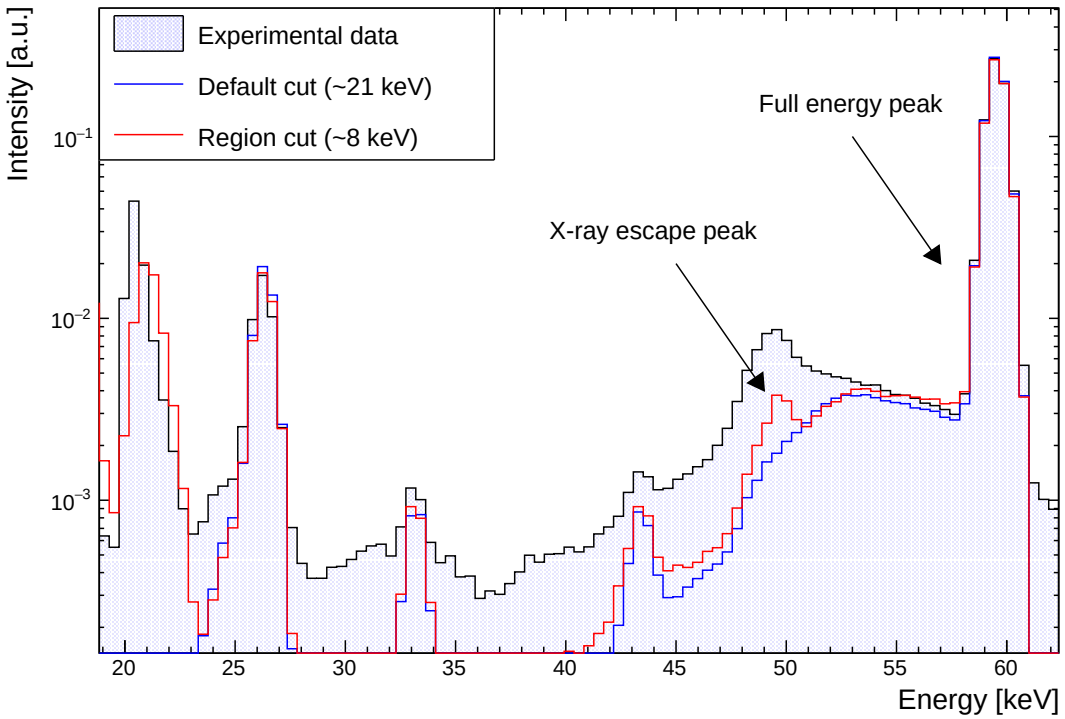



Figure 3.12: Example of X-ray fluorescence in the assigned region with reduced production cuts. The expected escape peak is now seen ∼10 keV below the 60 keV full energy peak of this $^{241}$Am spectrum. The simple simulation environment also demonstrates a difference in backscatter and low-energy background that can be extremely difficult to generalise.

Reducing the range cut for calculating X-ray fluorescence everywhere is computationally expensive, especially in complex geometries or at low energies. All these extra particles will then also require the full stepping and tracking treatment to propagate them through the simulation. To mitigate this as much as possible, the full crystal (i.e. active regions and dead layers) have been assigned to a special 'region'. This allows for custom production cuts to be

specified for the crystal alone, leaving everything else as a generalised compromise between speed and fidelity. This is based on consideration of dead layer attenuation expectations, distance/composition of surrounding materials, and most importantly the significance of the impact on the final spectrum.

### Multi-isotope sources

All examples so far have contained a single source. A method is needed for simulating spectra that contain the contribution of multiple isotopes. There are two ways to approach this. The first and preferred method of simulating a multi-isotope source is to choose one from a list of possibilities event-by-event. This was exactly what was happening in the original MCNP implementation, where sampling starts with the nuclides before continuing with the transitions of the chosen nuclide.

The second option is to put every nuclide needed in a single event. This has two problems: relative activities become difficult to implement, and memory usage can increase dramatically.

To reproduce different relative activities, some nuclides still need to be sampled more often than others, and will come with significantly more overhead than just emulating the MCNP approach.

On the memory side, this is primarily a limitation of hardware. At the end of each event, any data not considered relevant is discarded to keep memory usage low. Stored information for every object therefore becomes a concern when there are too many source nuclides in a single event, each needing to be decayed and tracked. Obviously this matters if interaction between two decay chain isotopes is needed, but practically it is far better to treat each decay as utterly independent. Fortunately the way the custom time handling class (Sec. 3.2.4) works, the number of nuclides will not be a problem in either case.

**Implementation** Going the event-by-event route, the idea is to sample from a list of the source nuclides based on activity and decay that nuclide for the current event. This is a simple and has very little overhead each event. All sources are defined at the start of each run, with only random sampling happening once at the beginning of each event. A struct is defined to hold the necessary nuclide information for convenience. This contains: Z, A, excitation energy, and activity.

The first three are simply used to define the primary particle. A `G4ParticleDefinition` object is returned after looking for the ion in a `G4IonTable`, then given to the `G4ParticleGun` in the primary generator action class. Storage and sampling the list of nuclides is moved out

of the primary generator action and into another class for better modularity.

Each nuclide is added to a vector with some activity (default 1.0), which is just a probability in this context and can thus be an arbitrary representation of the relative chance for a decay. For example, two sources with activities 50 Bq and 100 Bq can be given as 50 and 100, 1.0 and 2.0, or 0.33 and 0.66 if normalised. A cumulative probability is calculated and normalised to 1 to accommodate whatever format is easiest.

A basic loaded die method is implemented to choose a source, using the cumulative probability. For the example case a `G4RandFlat` may choose a nuclide based on the returned 0-1 value. If the random number is within the range 0-0.33, return the first nuclide, otherwise return the second when inside the 0.33-1.0 range. Naturally occurring radioactive material is one such example where multiple nuclides are required.

### NORM and the environment

As discussed in the challenges of RIID (Sec 2.2.2), Naturally Occurring Radioactive Material (NORM) is primarily from the $^{238}$U, $^{232}$Th, and $^{40}$K decay series. This corresponds to contributions from ∼25 distinct radioactive nuclides. It is difficult to characterise not only because of the large number of possible transitions from long decay chains [70], but also due to factors such as low activities, other sources, local activations, and all the usual artefacts that go along with gamma spectrometry (i.e. annihilations, fluorescence, Compton, etc...) [27]. For example, the $^{40}$K prominence at 1461 keV [71] results in a large Compton continuum that obscures and disrupts other low-intensity peaks.

There has been some work to simulate NORM to varying degrees of success [72–75], with most agreeing that the profile may be reasonably represented using a few dozen of the most probable transitions. The simplification was convenient for studies using MCNP, where all of these must be explicitly entered by the user. For GEANT simulations, all transitions are inherently considered by sampling from decay libraries.

Fortunately, in both the $^{238}$U and $^{232}$Th decay chains, all secondary nuclides have much shorter lifetimes than the original. This means that any natural material is typically in secular equilibrium, so the activity of any nuclide in the chain can be considered that of the progenitor. Ratios for simulating the three primary nuclides becomes much simpler with this assumption.

To investigate NORM simulation, a series of six spectra were collected sequentially with live times of 7200 s, splitting up the total 12-hour run to guard against unforeseen circumstances and gain drifts.

To aid in identifying photopeaks, a set of expected energies and intensities was compiled. A decay-only FISPACT-II [76] model covering several million years for each decay series, followed by extra code to extract the relevant information, listed relative intensities across entire decay chains. With this, every significant contribution to the spectrum was identified, finding two exceptions to be peaks at 662 keV and 2.22 MeV. Figure 3.13 shows a clear example of how the local environment can alter the NORM profile. This 662 keV peak is due to many $^{137}$Cs sources present in the source cupboard ∼10 m away. The 2.2 MeV and associated escape peaks are the result of a neutron bath housed in the same room, as explained in Section 2.2.2.

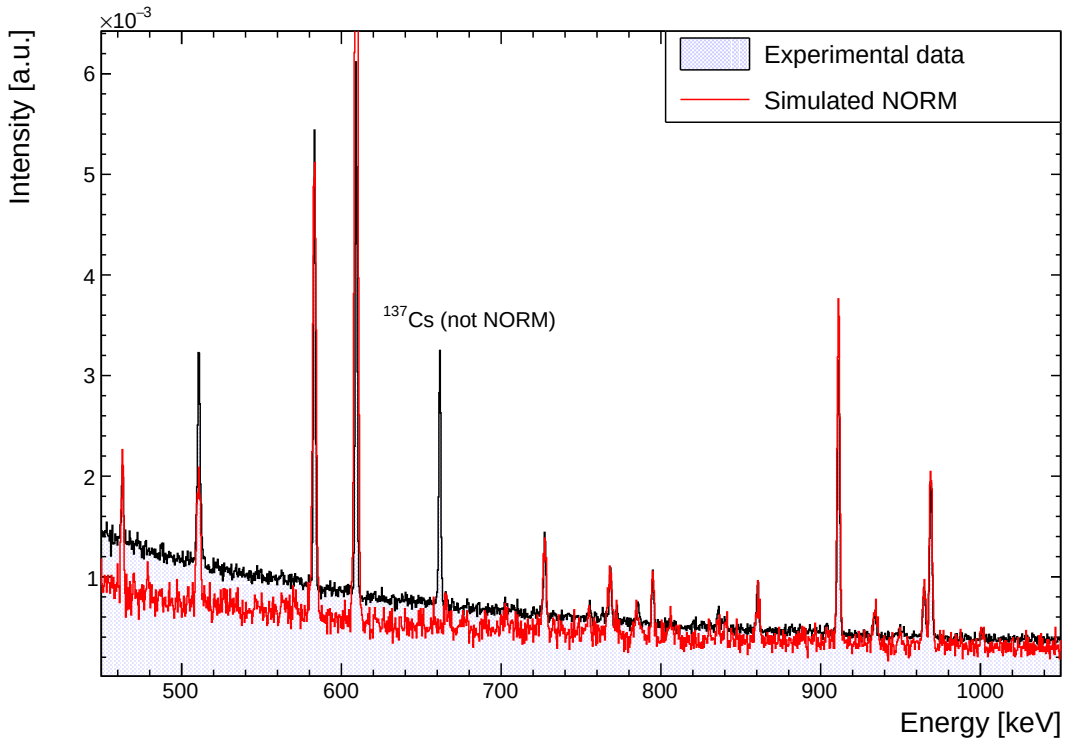



Figure 3.13: Example of extraneous sources contributing to the background profile. The $^{137}$Cs seen in the experimental data present is absent from this point-source NORM simulation.

In general, the mid- to high- energy peaks are represented fairly well, but it is clear that in Figure 3.13 there is divergence toward the low end. The primary cause of this is the amount of backscatter seen.

When the simulation was simply a point source at 10 cm in front of the detector, the <400 keV region contained almost no backscatter and fully captured the expected peak intensities. In reality, NORM sources are embedded in building materials and the ground,

meaning that there is likely to be significant attenuation of these over-represented low energy peaks. There is also a strong probability that the photons entering the detector have been scattered into the path of the crystal, resulting in a large continuum seen below 300 keV. As mentioned, there have been many suggestions for dealing with this. One paper [73] suggests placing the crystal 1 m above a disc of soil 20 m in diameter and 50 cm thick. The scale of this makes it difficult to simulate in any reasonable amount of time. Others [74, 75] showed that encasing a detector in a small concrete enclosure was accurate enough to simulate their environment. This was more computationally favourable and was tested here, with the result shown in Figure 3.14.

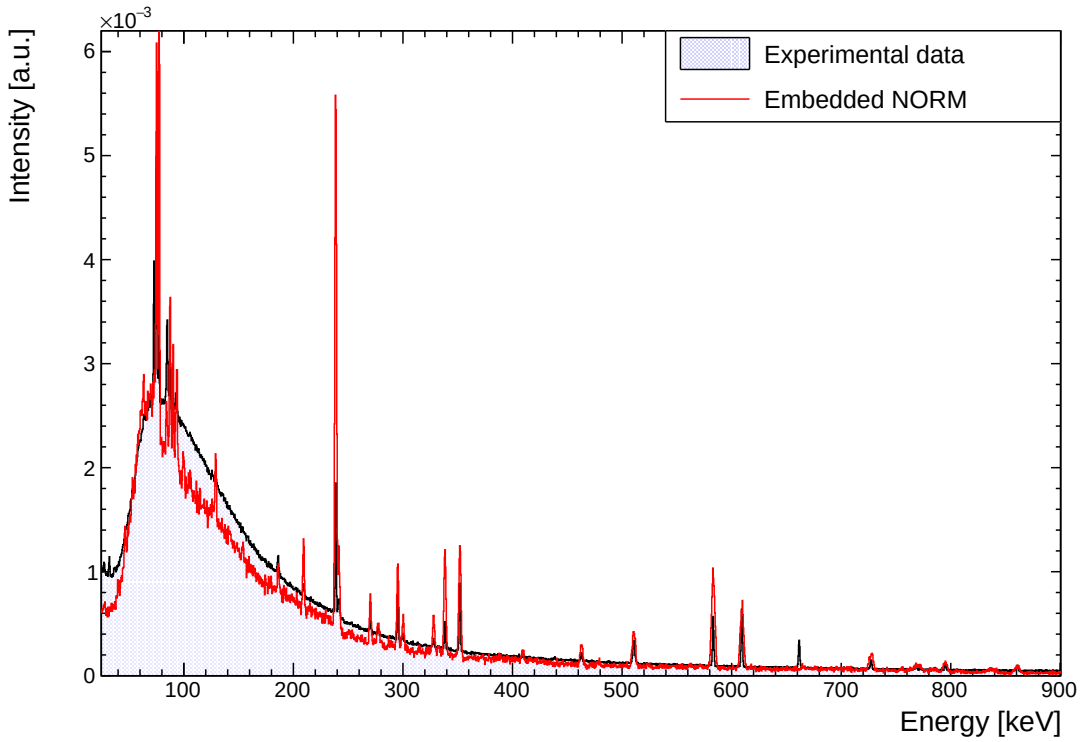



Figure 3.14: Example of one attempt to generalise the background profile, this time with NORM sources homogeneously embedded in surrounding concrete. Several peaks remain either over- or under-represented in the low-energy region because of the dominance of backscatter and attenuation.

This demonstrates the importance of backscatter and having an attenuating material. Here the NORM sources were homogeneously distributed throughout a surrounding concrete box. Unfortunately the backscatter and several peaks are still poorly reproduced. If absolutely necessary, it may be possible to parametrise a function that approximates the profile, which can be fit to real data and later incorporated into the simulated spectra.

At this point, iteratively making arbitrary changes to the simulation geometry meant that an over-specialised simulation was being developed for a specific background profile, which fundamentally goes against the application-oriented goal of the research. This, coupled with long simulation times, encouraged the conclusion that it would be impractical to simulate background. Of course, this is only a major concern for high resolution detectors, as systems with poor resolution smear out the features into a smoother profile.

It would be better to define a basic NORM template that contains a reasonable amount of backscatter. Alternatively, one would obtain an experiential sample from the site to use instead. Ideally though, during pre-processing of test data, a background subtraction may be performed to remove the complication entirely, allowing machine learning models to be trained on purely simulated data. Since this research seeks to include a variable background intensity, a real sample is used as template for a realistic compromise.

### 3.2.4 Timing

As alluded to in Section 3.2.3, accurate timing information is important because it expands the scope of the simulator. For energy spectra this means the inclusion of summation peaks, but with timing information comes the ability to emulate coincidence measurements with detector arrays. In developing this functionality, challenges related to timing precision became apparent. For the generalised solution needed here, methods were developed to work in the background for any nuclide without user intervention. This section discusses how these methods enable summations to be accurately included in the simulations, and how the problem of timing precision was resolved.

#### The true coincidence summing problem

Initial $^{60}$Co simulations in GEANT contained the notable peak at ∼2.5 MeV where MCNP simulations were lacking. In MCNP, the supporting codes were only providing transition probabilities from data libraries. The fact that this peak was barely seen is unsurprising given the $2\times10^{-6}$ relative intensity. More supporting code could be written to accommodate this with the techniques described by Britton *et al.* [77], but this would require calculations of the system geometry and a significant amount of user input. In GEANT, true coincidence summations are inherently included in the 'per-decay accuracy' approach to radiation transport. The problem is not that there are no summations as with MCNP, but rather too many.

Following a decay in GEANT, the entire series of secondary photons and particles is considered part of the same event. Two photons created years apart may enter the same detector

and deposit energy. Given no context, a decay will often result in spectra containing many spurious summation peaks. This is a consequence of using a naïve energy deposition model, where anything that enters the crystal is summed and recorded as a single, overall energy. This effect is very apparent in a $^{44}$Ti spectrum at a 5 cm stand-off shown in Figure 3.15.

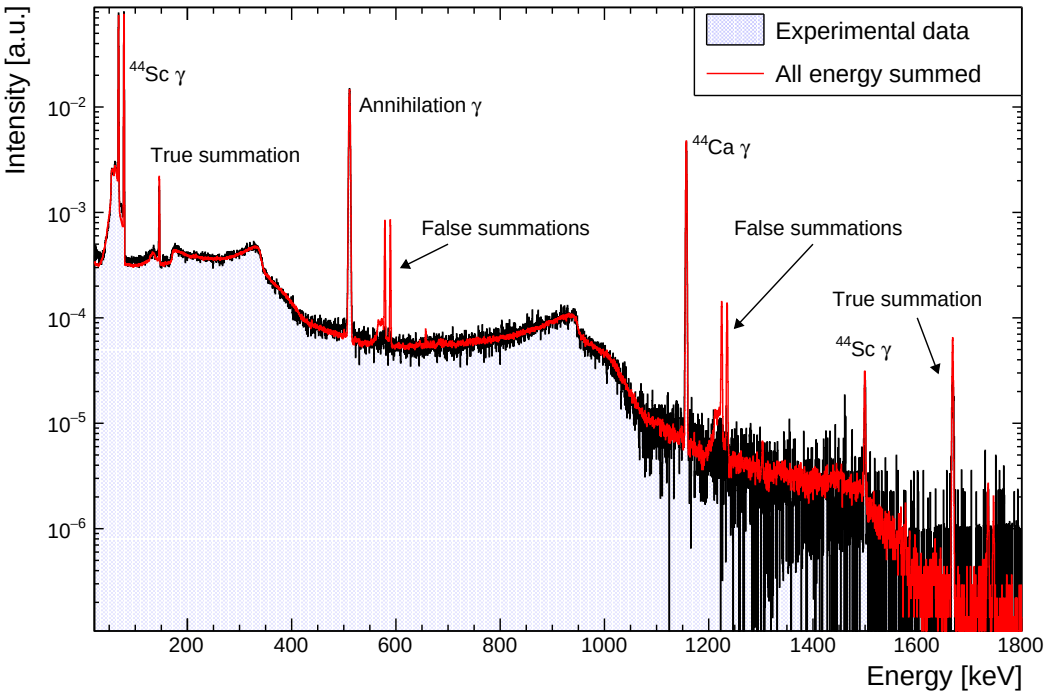



Figure 3.15: Spurious summations between the 68/78 keV and the much later 511 keV or 1157 keV emissions of the $^{44}$Ti decay chain. In more complex decay chains there can be many more false summations.

Even here, the summations can present significant features that need to be addressed. In more involved nuclides and multi-nuclide sources, the complexity of the profile can increase even more dramatically. In this case, the $^{44}$Sc emissions (78/68 keV) will occur *hours* before the annihilation photons and $^{44}$Ca de-excitations, as shown in the decay scheme of Figure 3.21. To put this into context, the experimental system used here has a conversion time of around 2 $\mu$s, and is certainly fast enough to process the $^{44}$Sc pulses before those related to $^{44}$Ca are produced in a single decay.

A strategy of comparing a list of independent tracks and summing only where appropriate solved this problem. These 'independent tracks' must be reconstructed from collections of individual steps before making comparisons. Methods were then implemented to find the arrival time in the scope of the event, with a threshold set for considering two tracks as time

independent. Unfortunately, this highlighted other issues with handling timing information that also needed solving later (see the precision problem in Section 3.2.4).

### Track reconstruction

When a step is detected within a volume registered as 'sensitive', a `Hit` object is created to store any relevant information. To demonstrate, the energy deposited by the photons labelled Examples A and B in Figure 3.16 is found by collecting all steps within the volume, and then reconstructing them into the two separate, independent tracks.

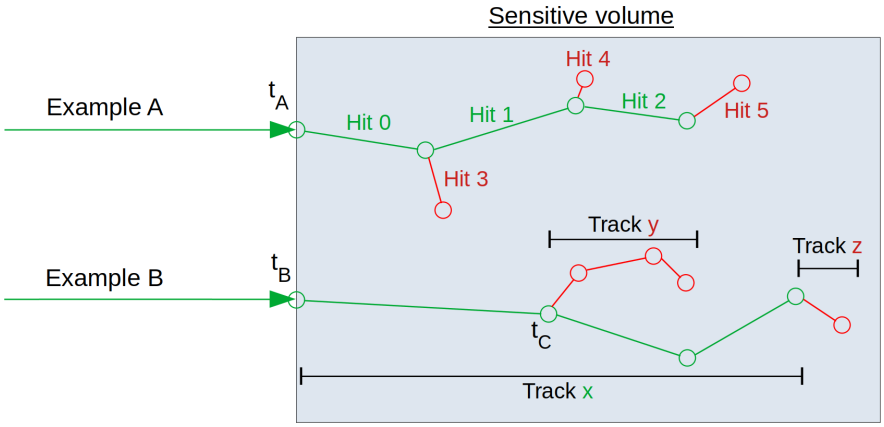



Figure 3.16: Basic case to demonstrate the track reconstruction, with primary (green) and secondary (red) particle steps shown. Examples A and B are two independent tracks that must be reconstructed from Hit objects. In Example B, the parent-child chains are followed through these Hits, and may be linked to combine the energy deposits of tracks $x$, $y$, and $z$. Summation between independent tracks is based on the entry times $t_A$ and $t_B$.

Looking at Example A, there will be three hits for the photon, followed by a hit for each secondary electron. As far as GEANT is concerned, these are three separate tracks. The task of the reconstruction is to link the energy deposits appropriately. In Example B, tracks $x$, $y$, and $z$ must be summed to capture the full independent track.

The hits in the collection are always in ascending order of 'track ID'[2], which is just a unique integer assigned to the track object. Unless the memory allocation or physics processes are modified, all secondary particles created are added to the stack in order of production, while the track itself is propagated to completion. Thus, a 'parent ID' is always available and can also be stored in the hit. To reconstruct, the hit collection is looped over at the end of the

[2]If not, the data are stored in a C++ map anyway so that it will not matter what is done to the stacking action, the time data are automatically arranged in track order.

event, with parent-child chains followed back to a common track seen entering the volume. The process for each hit is outlined in the flowchart of Figure 3.17.

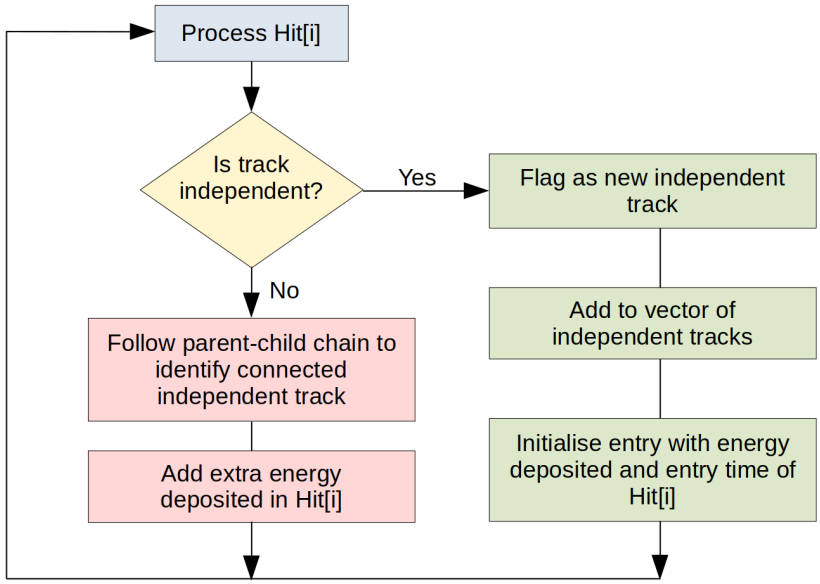



Figure 3.17: Basic reconstruction of Hit data into independent tracks that are entering the crystal. All energy deposits are added to that of the relevant independent track.

This reconstruction allows for multiple secondaries, and chains of secondaries, to all be recombined into one independent track. Once these independent tracks are defined, timing information can reveal correlations. If two of the tracks enter the crystal within some time window they are summed, otherwise they are left separate. The final set of energy deposits is added to the final histogram, sampling from a Gaussian for the energy broadened version.

By reconstructing independent tracks and summing as appropriate at the end of the event, the computational expense is minimised. If conditions were enacted step-by-step as the hits were created, simulations would be slowed significantly. As far as possible, the `ProcessHits()` method in the sensitive detector class should be doing the bare minimum, simply collecting relevant hit information.

It must be noted that these tracks take time to deposit their full energy, but it is typically in the order of picoseconds for most small crystal sizes [25]. Any other choice of definition for deposition time will simply add a more or less constant offset and is not of practical concern in this particular system, where the summation window is taken to be is as large as 2 $\mu$s.

If a more rigorous look at the precise timings is needed, then further complications arise. For example, the branch at $t_C$ in Figure 3.16 has a secondary electron which goes on to scatter around. The time recorded for this new track starts at zero and must be considered relative to

the parent track (i.e. the photon). This is true of every secondary, requiring methods to store the parent-child chains of each independent track so that the steps may be looped over to find a total deposition time. For the purposes of this research, the additional computational expense was considered unnecessary given the speed at which energy is deposited, so only the entry points $t_A$ and $t_B$ are taken for comparison. In either case, a broader look at the 'global' time for the entire event is required, and the issues with doing so are discussed in the next section. Figure 3.18 shows the result of the reconstruction to determine track independence in terms of true coincidence summing.

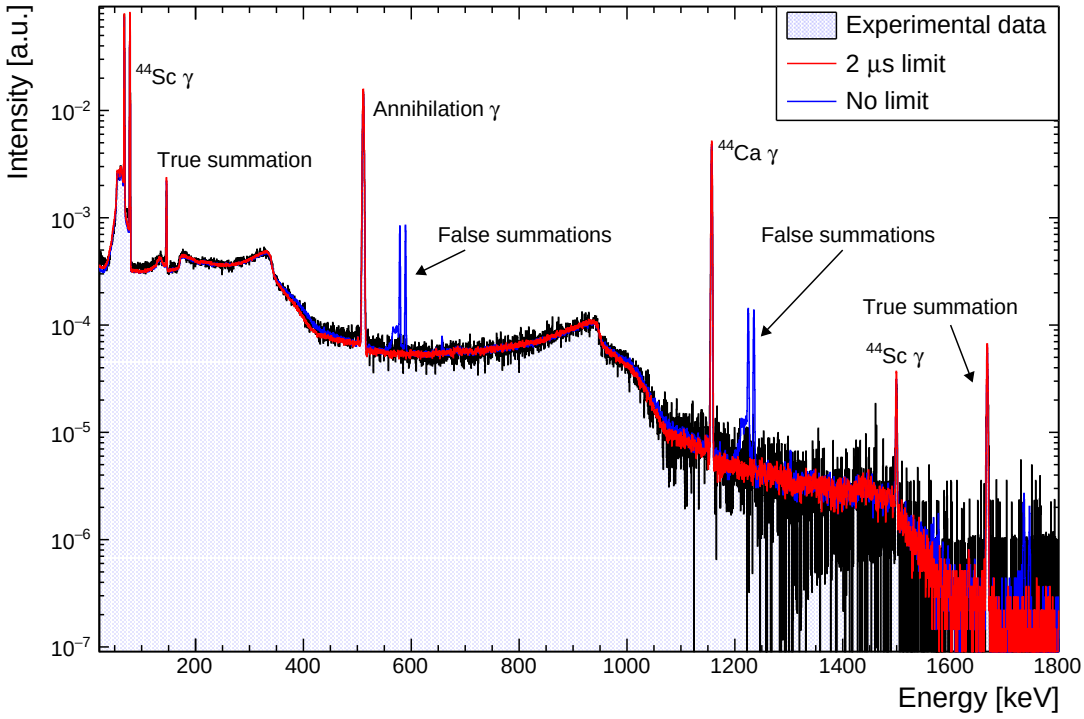



Figure 3.18: The times for tracks entering the detector are checked against an acceptance window, and treated as independent if outside this time window. This removes the previous false summations while retaining true summations.

One simple test of this is to collect spectra at various stand-off distances for comparison with the $^{60}$Co summation. The result is shown in Figure 3.19, with good agreement to the expected trends from geometrical calculations.

The concept of time in GEANT is actually tracked through several definitions. The most important are those of *Global* and *Local* time. The global time is set to zero at the beginning of the event, with all times calculated relative to this point. A local time is calculated relative

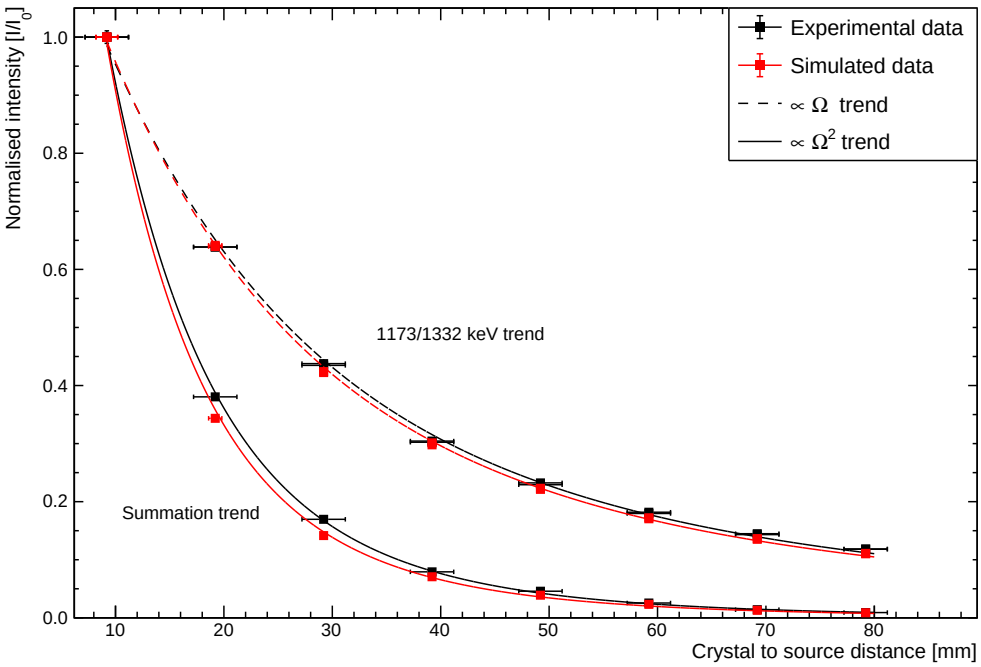



Figure 3.19: Comparison of photopeak intensities as the position of a $^{60}$Co source is varied. Note the summation peak following the expected solid angle ($\Omega$) summation trend.

to the start of each track instead. This means that for local times, there is no continuity in times recorded for secondary particles.

If only local times were taken for comparison, everything is likely to be summed because photons reach the detector extremely quickly. The global time *must* be consulted for consideration of state lifetimes. For example, this means the $^{44}$Sc (78 keV) will not sum with the $^{44}$Ca (1157 keV) photon, which is created hours later. This was the case seen in Figure 3.15. Unfortunately, this was not as simple as querying the global time for each track or step due to the huge numbers involved. This led into the next generalisation challenge to be addressed: timing precision.

### The timing precision problem

A combination of a picosecond base unit and very long lifetimes results in truncation to `double` precision, which can be extremely detrimental to timing information. For example, any hope of nanosecond precision is certainly lost when comparing two times in years. The solution is to reset the global time as appropriate while decay chains are being followed, and find differences by calculating times relative to the closest common reset point.

As mentioned previously, the issue with using the local time is that it is zero for the start of every new track. This leaves the global time as the only usable reference point to put a decay in the context of the whole event. Unfortunately, when it comes to radioactive decays there is a vast range of lifetimes, anywhere up to billions of years. While the base unit for GEANT may be picoseconds, the C++ double precision means that as soon as stored times exceed ∼16 minutes they start to round to the nearest picosecond. For a year the smallest unit becomes ∼32 ns. It quickly becomes unreasonable to look at any sort of detail of $^{44}$Ti for example, which is used in undergraduate experiments for investigating a 154 ns excited state half-life [78]. The issue is an 87 year lifetime offset from the initial $^{44}$Ti decay to $^{44}$Sc. This is demonstrated in Figure 3.20, where unmodified global times are compared directly as soon as each track is created, leading to the clear artefacts seen.

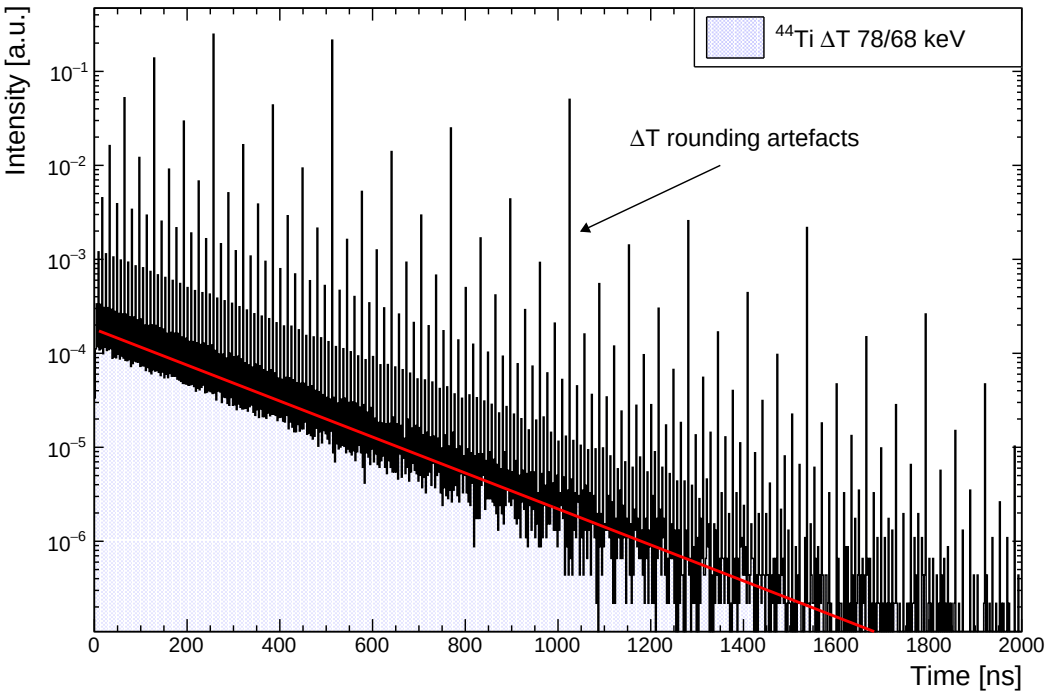



Figure 3.20: Demonstration of the loss of precision at large global times using the 154.8(8) ns state in $^{44}$Sc. The time logged measures the lifetime of this first excited state, with the exponential fit showing an incorrect 157.3(5) ns half-life due to rounding artefacts.

One way of addressing this rounding problem is to reset the global time to zero when the first photon is created, keeping the magnitude small enough to avoid non-negligible rounding. This is fine as a specialised solution, but a method was required for generalised use that would work with any nuclide given by the user.

### A new timing class

The original approach was to simply reset the global time to zero whenever a decay results in a change in proton number. Recorded times are then always relative to the populated state of the new nuclide. This is flawed in that there may be long lived states, and that the global time has already been updated. The current implementation therefore looks at particle lifetimes *before* enacting physics processes, and then decides whether or not to reset.

To account for long-lived states, the current reset conditions trigger when either a new nuclide is created or a long lifetime is seen. These prevent too much time building up between resets, and provides conceptually easy markers for debugging and ensuring physical accuracy. The $^{44}$Ti example is continued here to demonstrate reconstruction when there are multiple nuclides, while having relatively few emissions for simplicity (Fig. 3.21).

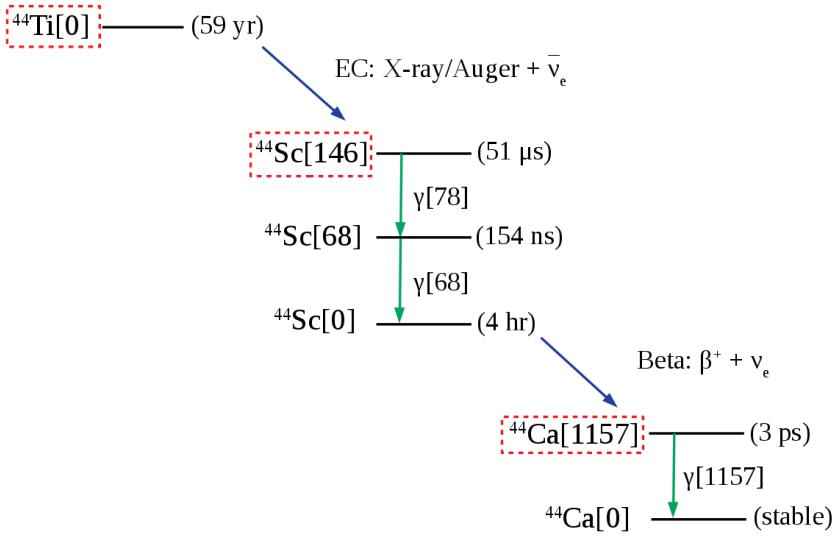



Figure 3.21: A simplified $^{44}$Ti decay scheme, showing reset points (dashed red) for the global time. All time differences are calculated to the nearest reset point rather than the primary particle, keeping the magnitude of stored values small.

To find global time differences with the precision needed, a custom class was made with a static method to allow a `std::map` to persist across multiple instances. This means that information can be easily accessed and updated at any level of the event.

The map always consists of a <*key*> and <*data*> pair, and was chosen for the standard methods that come with it and the fact that everything will be sorted in key order, regardless of when the entry is made. In this case the key is the track ID, so changes to the stacking action do not effect the timing methods. The data are stored as an object containing parent ID, global time, local time, and whether or not the track is a reset point.

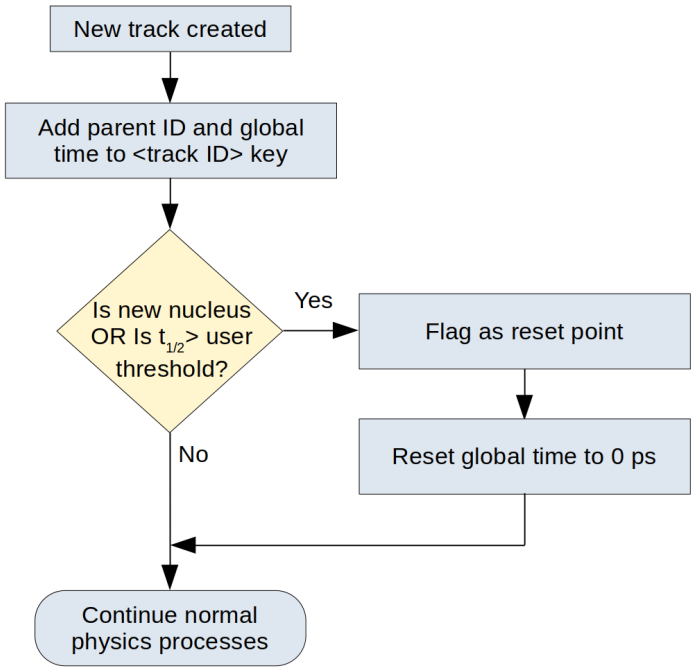



Figure 3.22: Adding timing data for each new particle track, resetting the global time to zero at key points to manage magnitudes of stored times. The conditions for doing so may be redefined as needed since the handling of timing information is self-contained.

The method outlined in Figure 3.22 is simply noting any reset points (red boxes Figure 3.21) where relevant, and filling up the necessary information along the way for use at the end of an event. The crucial part is zeroing the global time. This keeps all of times at a manageable magnitude because they are generally relative to a populated state, not the parent nuclide. Again, with how tracks are added to the stack, all necessary information is available to the sensitive detector should it be needed mid-event. Adding calculations so low in the simulation chain would result in poor performance, but the option is there.

Once reconstructed, independent tracks are checked to see if they should be summed together based on some acceptance window as outlined in previous discussions. This heavily relies on the methods of this new class to ensure precision is maintained. Without it, a $1\times10^{6}$ yr half-life may easily round to the nearest 32 ms, far beyond the 2 $\mu$s window.

To determine whether independent tracks are to be treated as separate, two time values are needed: the difference in creation time, and the difference in arrival time at the detector. The creation time offset is calculated via a generic interface of this new class, where any two existing track IDs are given as arguments. The arrival (or travel) times may simply be taken from the local time. For the summations then, the two offsets are summed to give a total that is not calculated through detrimentally rounded values.

### Expanding simulation capability

To maintain precision in recorded time information, some form of reset was now being applied and tracked, and a generic method written to find the time difference between any two tracks. This opens up the possibility of simulating coincidence experiments and other time-sensitive simulations that would otherwise require bespoke solutions. One related project even investigated the simulation of signals for pulse shape discrimination for RIID [79].

For simplicity on the user side, two 'triggers' may be set just as they would experimentally. If the full energy deposited is within some energy window, the appropriate trigger stores the track ID. A basic example is shown in Figure 3.23 to remain consistent. The red box shows the reset point, while blue boxes are reconstructed tracks found in the crystal.

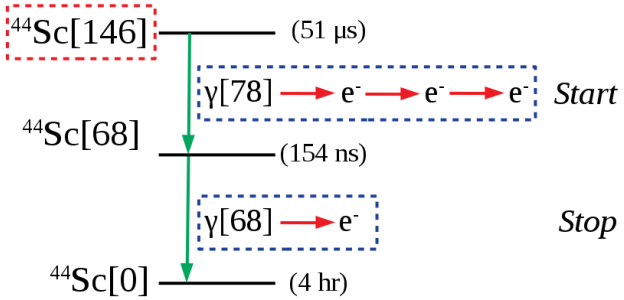



Figure 3.23: A scandium-44 example for global time reset and base offset calculation. The red box indicates a reset point, while blue boxes are reconstructed independent tracks.

In this case, two triggers would be set to 78.3±0.1 keV and 67.9±0.1 keV for a check in the sensitive detector class. At the end of the event, if both triggers were activated, the time difference between the two independent tracks is calculated. Again this may be calculated with precision for any two tracks, but by adding this trigger mechanism at a higher level, there is a clear parallel between simulation and experiment. Taking only the creation time difference for a moment, this strategy may be used to repeat the measurement of Figure 3.20. The result is shown in Figure 3.24 and contains none of the previously seen artefacts and agrees well with the literature value.

One good test of these generalised techniques is to define $^{232}$Th as the primary particle for an extreme case. This has a large, $1.4\times10^{10}$ yr half-life, and typically decays through nine nuclides and two significant branches before reaching stability. At the end of the chain, one of these branches sees $^{208}$Tl $\beta^-$ decay to $^{208}$Pb, which may de-excite through a state possessing a 294 ps half-life. The result of setting the appropriate trigger energies is shown in Figure 3.25.

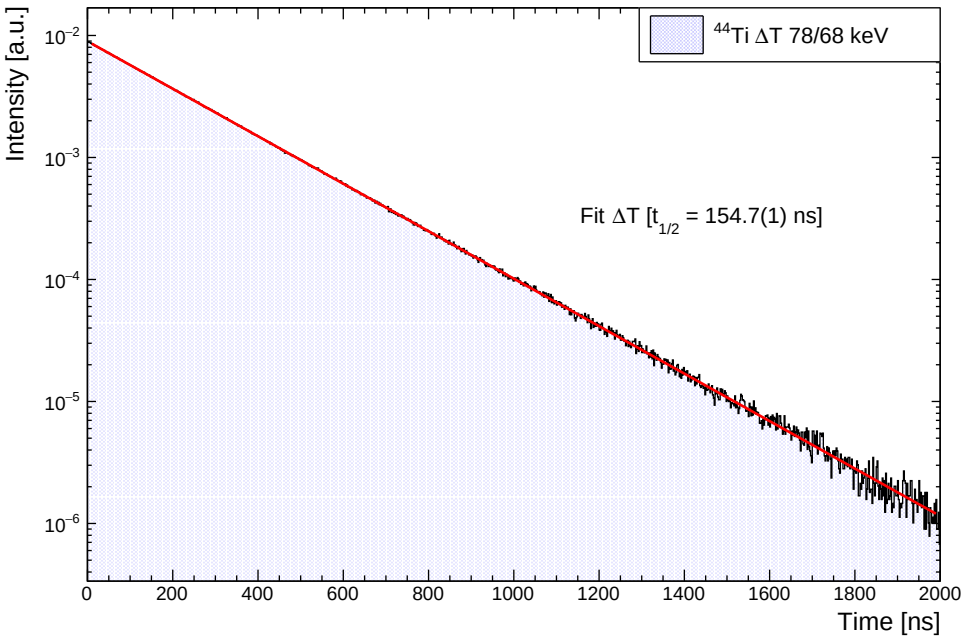



Figure 3.24: Demonstration of the time reset methodology for the 154.8(8) ns [80] half-life in $^{44}$Sc. Since GEANT samples from literature values, Figure 3.20 should have agreed this well.

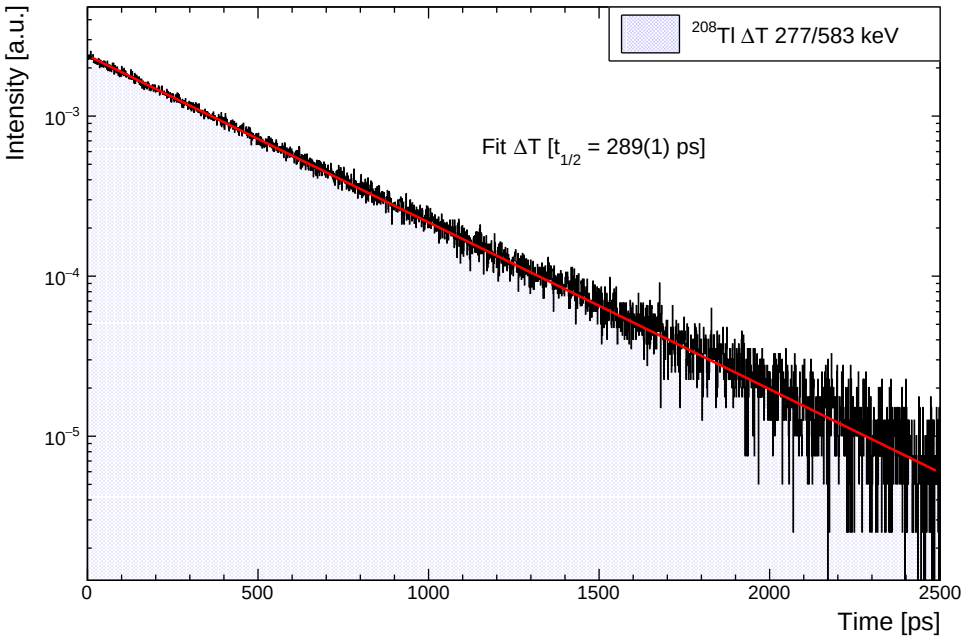



Figure 3.25: Time spectrum of a late stage excited state of $^{208}$Pb in the long $^{232}$Th decay series. The fit half-life of 279(2) ps compares well to the 294(15) ps [81] literature value, maintaining precision with no effort on the part of the user.

Making use of mother volumes and copy numbers, it also becomes very simple to make duplicates of detectors. Setting up a more modular detector construction class, arrays of detectors are easy to include. A two-detector system may be created that mirrors experimental coincidence measurements, as shown in Figure 3.26.

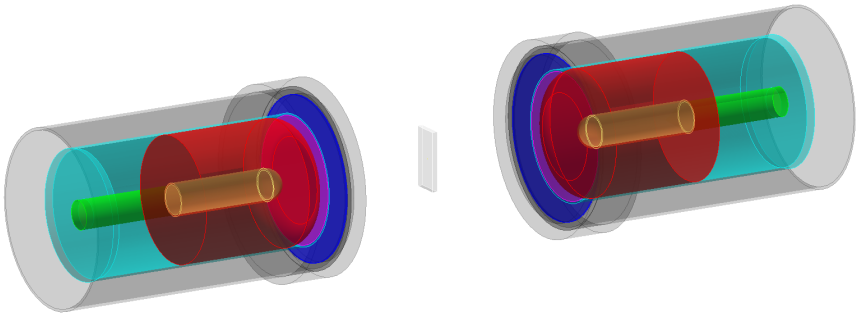


Figure 3.26: Multiple detectors with a central $^{44}$Ti source for $\gamma$-$\gamma$ coincidence measurements.

Again, using the $^{44}$Ti example for continuity, the triggers may be set appropriately and some conditions added to the sensitive detector. Time differences are recorded only when the two tracks enter separate detectors. The time calculated is then passed through a sampling function much like that used for GEB, as physically, the same principles apply.

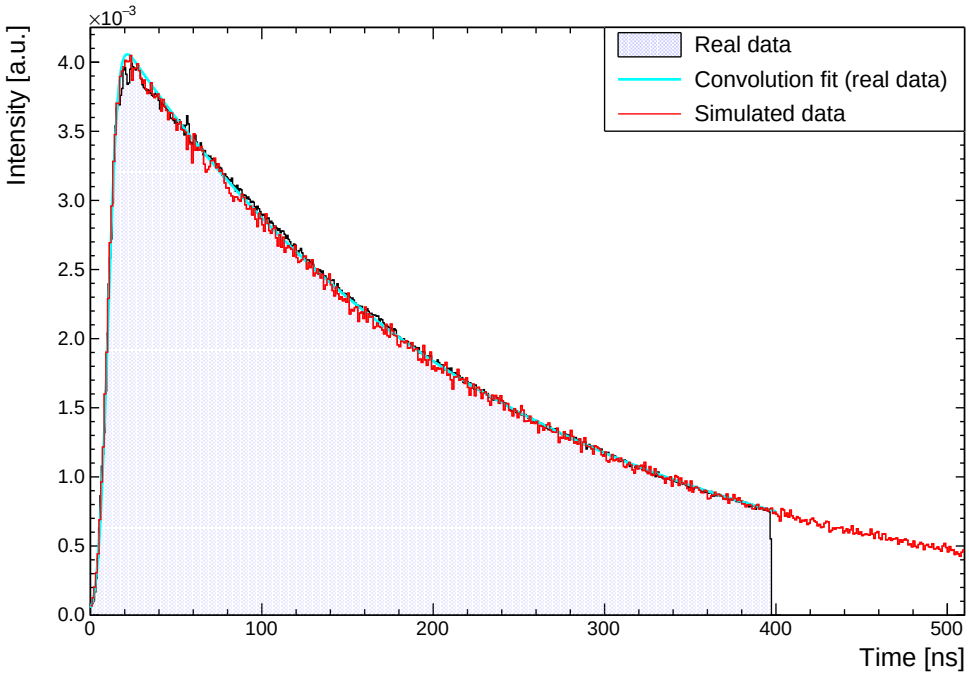



Figure 3.27: Comparison to the $^{44}$Sc lifetime measurement from Ref. [78]. The real data are fit with a convolution of a Gaussian and exponential, with the simulation in excellent agreement.

Figure 3.27 demonstrates how this compares to real data. The fit is a convolution of a Gaussian (from time resolution) and an exponential (from the excited state lifetime).

With multiple detectors and coincidence measurements possible, it follows that angular correlation experiments might also be emulated. In this setup, a plane is chosen, and the angle between detectors varied around a central source. A good example of this is the $4^+ \rightarrow 2^+ \rightarrow 0^+$ cascade of $^{60}$Co. To demonstrate a current limitation, the $^{108m}$Ag source is used here. This has the same cascade, but with the added benefit of a $6^+$ state above that. As Figure 3.28 shows, angular correlations are only partially implemented in this version of GEANT4.

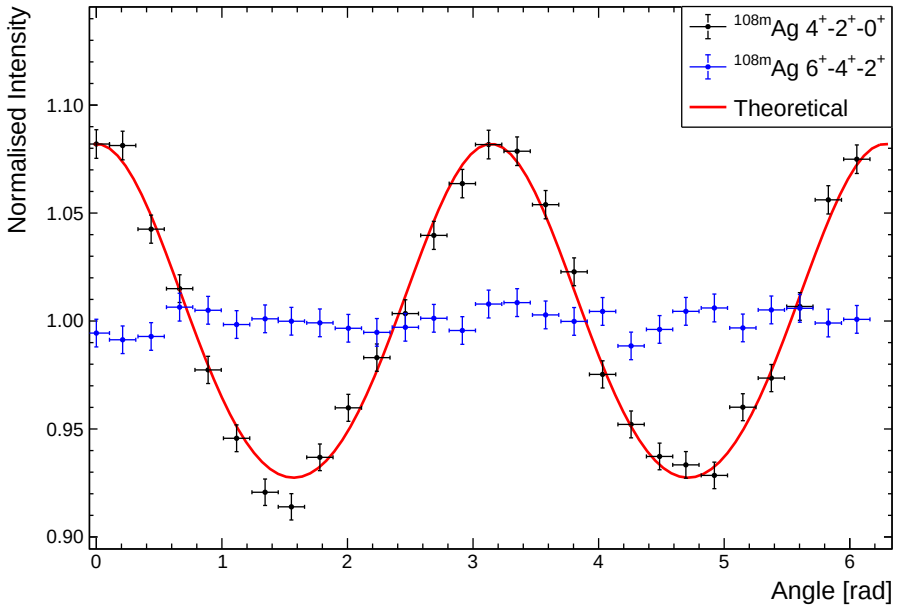



Figure 3.28: Partially implemented angular correlations of GEANT. As the $^{108m}$Ag example shows, any nuclide with the same $4^+ \rightarrow 2^+ \rightarrow 0^+$ cascade will work, but there is no generalised solution implemented for any other set of transitions. The $6^+ \rightarrow 4^+ \rightarrow 2^+$ is expected to show the same trend (though slightly supressed).

Unfortunately, the currently included implementation of angular correlations [82] are slow for anything other than the demonstration case of $^{60}$Co. As such it is not enabled for anything other than the $4^+ \rightarrow 2^+ \rightarrow 0^+$ shown here, failing to show anything for the $6^+ \rightarrow 4^+ \rightarrow 2^+$ that should have the same trend. The options are to create overly-specialised solutions, wait for GEANT development to implement it, or create the generalised technique personally. Since none of these are favourable and the extent of the correlation is negligible for RIID, no further work was carried out to implement angular distributions.

## 3.3 Summary of simulator

The simulator presented is a major focus of the research. It provides the foundation for a rapid development environment by supplying $\gamma$-ray spectra for a diverse range of applications. Research aims quickly moved toward applying machine learning techniques to the RIID problem space, with the vast majority requiring large datasets filled with example spectra to learn from. The use of simulated spectra is well matched to such techniques, with a good simulator bypassing the need for experimental data in prototyping phases. Section 2.2.4 highlights a recent tendency to adopt this approach.

Through the work presented in this chapter, a generalised spectrometry simulator was developed that could reliably reproduce the energy spectra of even high resolution detectors. All the important features are accurately reproduced, with GEB and multi-isotope sources included. This extends to the more subtle effects, such as the extra broadening of annihilation peaks, X-ray fluorescence, and true coincidence summations.

While there is a tendency to ignore the summation peaks, this work felt the effect of close source-detector geometries was important for practical RIID (see Sec. 2.2.1). To accommodate this, significant work on the handling of timing data at a fundamental level made this information readily available. The result is a simulator with extended functionality towards more in-depth investigations, such as coincidence and angular correlation measurements with multiple detectors. On a case-by-case basis, this is not too difficult to achieve. However, the generalised techniques implemented here allow for far more versatility.

Of course, no simulator is ever perfect, especially those intended to be a generalised solution. The angular correlations showcased demonstrate the partially implemented functionality of the current GEANT source. While negligible for this research, a solution would have to be manually coded for applications that require it.

The greatest source of discrepancy in the overall profile of a spectrum is always in the low (<100 keV or so) energy region. The backscatter, NORM, and extraneous background sources are so specific to the environment and surrounding material that a generalised profile becomes impossible to replicate.

With an awareness of these current limitations, the simulator was certainly good enough to closely reproduce even HPGe spectra with minimal effort. For lower resolution detectors, backscatter and NORM effects are somewhat mitigated due to most of the subtle features being smeared out. It now rapidly supplys example spectra for various RIID conditions. The next chapter details how these spectra, along with their experimentally obtained counterparts, are used to create large datasets full of examples for machine learning models to train on.

# Chapter 4

# Data preparation for machine learning

## 4.1 Datasets

Machine learning techniques always require good data to learn from. As outlined in Section 2.3.1, the data need to be split into three sets: training, validation, and testing. In this research the majority of training data are simulated spectra, with models tested against real spectra that were collected experimentally. The core set of radioactive sources used are: $^{241}$Am, $^{108m}$Ag, $^{133}$Ba, $^{207}$Bi, $^{60}$Co, $^{137}$Cs, $^{152}$Eu, $^{22}$Na, and $^{44}$Ti. Use of these readily available sources is convenient for validation of simulations, and for comparison against similar published work.

The RIID conditions chosen reflect real challenges, so variations in parameters such as shielding and stand-off distance were included beyond simple point sources. Combining several setups with every possible unique combination of sources and effects, it becomes impossible to collect a comprehensive set of experimental data. Even simulating multiple samples for each situation would require a significant investment of computational resources and time. 'Template spectra' were therefore used instead, providing excellent examples of the core problem that may be augmented later. Through techniques developed here, large datasets for prototyping are generated extremely quickly, even on the i5-5200U of this author's laptop.

### 4.1.1 Template spectra

In data science it is common practice to augment small datasets to create additional, and equally valid training samples from which to learn. One example is changing the rotation or altering the colour balance of samples in an image dataset. Here, it is almost impossible to physically collect a large enough dataset to be useful, so this augmentation concept is applied to gamma spectra too.

Fortunately, with the domain knowledge necessary to understand the physics and signal chains that produce gamma spectra, several realistic perturbations may be implemented. These are discussed in detail in Section 4.2.1. The main prerequisite is having a good example to work with, so spectra with good statistics are required to act as templates. There is typically one template for each source per varied parameter (e.g, number of shielding plates). Even multi-isotope spectra may be built up surprisingly well from template combinations [16]. Note that an environmental background template is always experimentally obtained because, as reviewed in Section 2.2.3, it is impractical to simulate and extremely difficult to generalise. For these reasons, all template spectra are calibrated to allow direct combination and comparison, with a consistent 8192 channel structure ensured between simulation and experiment over a 0-3 MeV range.

## 4.2 Preprocessing techniques

Specific details of various datasets are mostly excluded from these discussions. The techniques presented are generalised to apply to any augmented dataset built up from template spectra. The ROOT [83] analysis framework is used for all preprocessing due to the excellent infrastructure surrounding histogram objects. The templates, simulated or experimental, are built with the following process:

1. Convert template spectra to histogram objects
2. Find all possible unique combinations of the template set
3. Loop over unique combinations:
   (a) Make $N$ training samples with randomised perturbations of:
      i. Total counts
      ii. Gain shifts
      iii. Low energy discrimination thresholds
      iv. Relative activities
      v. Background contribution
   (b) Rebin each template to downsample
   (c) Normalise intensities

The detail of each step is outlined in Section 4.2.1. The idea is to start with a minimal library of the computational/labour heavy spectra, and then combine them with random perturbations to massively scale up the dataset. This is only possible because the augmentations are physically viable and there are enough degrees of freedom to allow it. The parameters that define each training sample are also logged for reference. The following sections go through these step-by-step to provide detail of the datasets used and justify any choices made. In machine learning, it is so important to understand the content and scope of the datasets used for an accurate portrayal of performance. For example, a model trained on perfectly calibrated spectra will perform well when tested on the same, but providing shifted test spectra will exceed the scope of the model and fail.

### 4.2.1 Methods

#### Templates

For real data, the Maestro *.Spe* ASCII format spectra are converted to several ROOT style file structures for flexibility. The C++ code written to do this strips the header and footer information while recording live times for background subtractions later. Data are converted to `TH1D` histogram objects and saved as individual histograms. This allows for extremely convenient comparison to the simulated spectra produced by GEANT, and makes all the functionality of ROOT readily applicable.

Experimental data inherently contain environmental background, which must be removed from the spectra to create templates containing only contributions from the source of interest. It is often the case that there are no gain shifts over the course of the experiments, so background may be scaled appropriately based on the live time before being subtracted channel-by-channel from source spectra. If this is not possible, all spectra are calibrated before the subtraction, though this introduces marginally more uncertainly having re-binned based on a calibration fit.

These background subtracted spectra become the templates upon which a dataset is built. For the sake of example, the 10 cm stand-off data for a NaI detector are used throughout. The template histogram objects are listed in Table 4.1.

It is important to assign a numerical ID to each template that is to be included in the dataset so that all unique combinations will be represented appropriately. This is also used in producing multi-isotope spectra.

Table 4.1: Example template mapping.

| Assigned ID | Template source |
|---|---|
| 1 | $^{108m}$Ag |
| 2 | $^{241}$Am |
| 3 | $^{133}$Ba |
| 4 | $^{207}$Bi |
| 5 | $^{60}$Co |
| 6 | $^{137}$Cs |
| 7 | $^{152}$Eu |
| 8 | $^{22}$Na |
| 9 | $^{44}$Ti |

### Multi-isotope combinations

There will always be multiple isotopes in real spectra. This is often environmental background, but may also be medical and industrial isotopes, or more a complex material containing many sources. It is therefore crucial that multi-isotope spectra are included in the training and validation datasets so that the models may learn the features that best distinguish one source from another. While this can be done by the simulator, the time taken to simulate many different combinations of relative activities would be counterproductive. Combining single isotope spectra in post-processing is much faster, allows for arbitrary activities, and emulates multi-isotope spectra surprisingly well [16].

Ideally, the dataset will contain a sample for every possible unique combination of sources to build up a comprehensive set. The number of unique permutations ($N_p$) of a vector containing $n$ values for a given subset (size $r$) is simply,

$$N_p = \frac{n!}{r!(n-r)!}. \tag{4.1}$$

For example, given the nine templates in Table 4.1 there are 84 possible ways to combine three sources. The order is not important, but each combination can not be repeated. Code was written to return a vector containing every unique subset of Assigned IDs, and it simply becomes a matter of looping over this to build spectra containing the required sources.

If every combination of sources was to be included, this makes a total of 511 unique combinations. This is true of all datasets used unless stated otherwise. In doing so it is ensured that every possibility for a particular RIID scenario is represented in some way, and that the model learns not to rely on features that may often be obscured by more dominant features of other isotopes. Of course, 511 spectra constitutes a very small dataset and would

only contain ideal representations of mixed isotopes that all have the same activity. This is where the augmentation idea returns. Instead of randomising rotation and colour balances in images, the spectra are augmented by including random perturbations that emulate physical effects. However, before these operations can be applied the samples must first be created, and for that the relative activity and total number of counts in each spectrum must be decided.

### Building a sample

To build up a spectrum from template spectra, the relative activity of each source must be decided. Like all predictable unknowns, it must be assumed that anything is possible within appropriate ranges. A vector of factors is created and filled with numbers randomly sampled using a uniform `TRandom` method from a 0.25 - 1.0 range. Each isotope will have a corresponding number to represent its share of the total counts.

The templates contain no environmental background. If they did, the background would be significantly overrepresented in spectra containing more isotopes. An additional factor for the background contribution is included, but capped at 25% of the total counts to prevent it from unnaturally dominating spectra. The vector of factors is finally normalised to 1, making it clear what fraction of the total counts each template is to contribute.

The number of total counts used to create a sample is randomly sampled from a logarithmic distribution. This ensures there are more samples created with poorer statistics. Of course, a reasonable minimum is enforced depending on the situation, often simply taken as 500 counts per template involved. This is sampled using the `TRandom` exponential random number generator, with the slope parameter $\tau$ defining how biased the dataset is to poorer statistics. The choice of $\tau$ is somewhat arbitrary, so a value of 0.3 was chosen such that spectra with the minimum number of counts are roughly an order of magnitude more likely to be created than those containing the maximum.

At this point a total number of counts is decided, as is the contribution of each source to the final spectrum. Normalising template spectra transforms them into probability distributions that may be randomly sampled. This goes back to Monte-Carlo methods, where sampling from known underlying distributions eventually builds up a more accurate representation of reality. The template spectra provide these distributions, and by sampling a small number of total counts, the poorer statistics of a short acquisition is emulated as demonstrated in Figure 4.1.

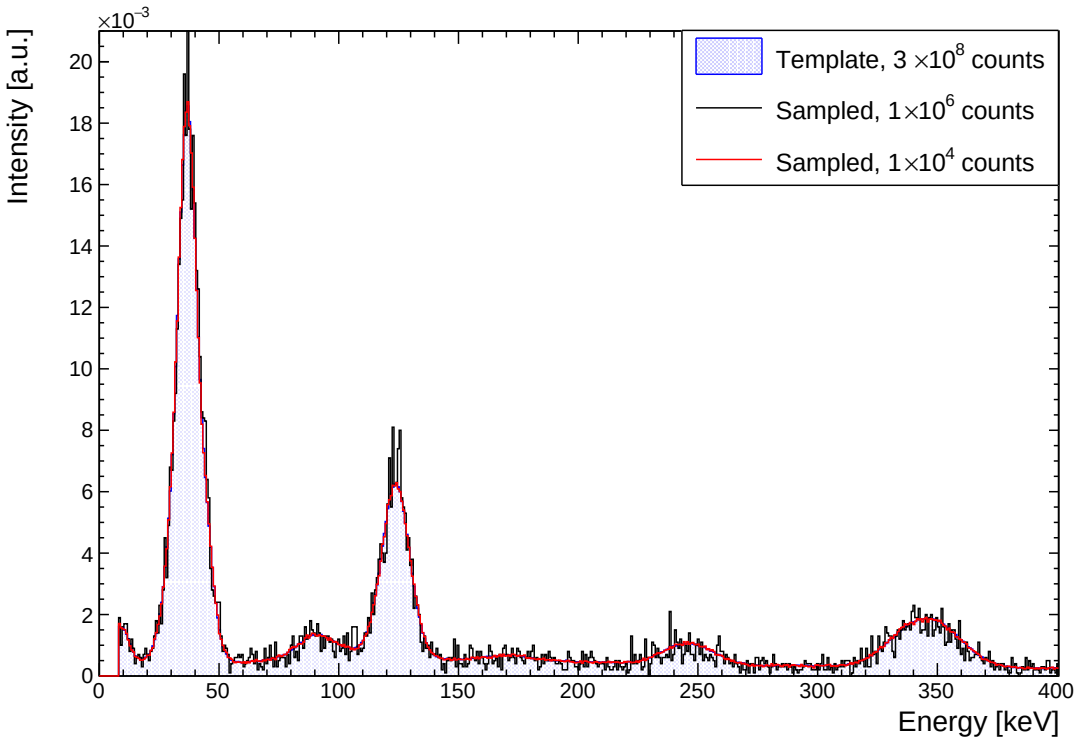



Figure 4.1: Example sampling of template spectra, downsampled for clarity. The original template for this selection of experimental $^{152}$Eu data are well reproduced with $1\times10^6$ randomly sampled counts, while worse statistics are emulated by taking only $1\times10^4$ counts. In the higher energy regions, where photopeaks are less prominent, the noise becomes even more exaggerated.

### Threshold and noise

While the spectra remain calibrated, a lower energy discrimination can be applied. This replicates the common experimental practice of setting a small ADC threshold to exclude low level noise, with any counts in bins below this threshold set to an intensity of zero. This may be seen in the real data of Figure 4.1, where the threshold was ∼10 keV. Taking the training samples to more of an extreme, a maximum energy of 50 keV is generally defined when randomising a threshold. This <50 keV cut is limited for the sake of $^{241}$Am, which predominantly emits a 59.5 keV $\gamma$-ray, while remaining large enough to remove significant peaks from several sources. The random element comes from taking the product of threshold energy and some scaling factor, typically 0.5-1.0 from a uniform random distribution for the purposes of these data. An example of the potential information loss is shown in Figure 4.2.

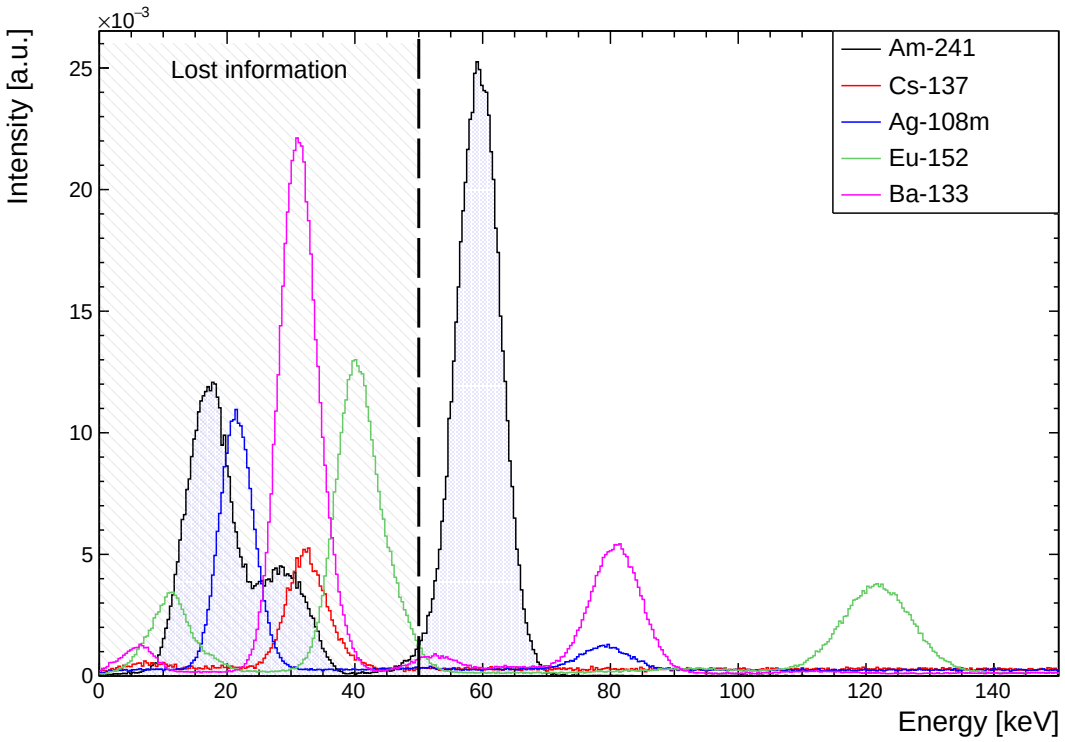



Figure 4.2: Example of the loss of information with 50 keV low-energy discrimination. Simulated spectra are used for demonstration purposes. Note that these low energy peaks are typically the most dominant features, making their loss significant for RIID. The $^{241}$Am spectrum is seen here in its entirely, hence the 50 keV maximum.

This mostly effects $^{108m}$Ag, $^{133}$Ba, $^{137}$Cs, and $^{152}$Eu, all of which have distinct photopeaks below 50 keV. This can make it particularly difficult to distinguish $^{137}$Cs from other sources. Losing the 32 keV peak leaves only one peak in the spectrum to use as a feature, much like $^{241}$Am.

Gaussian noise may also be added to spectra. This is rarely used due to the inherent noise in the spectra created by randomly sampling templates. Poor statistics is mimicked well by sampling a low number of total counts from templates anyway. However, if required there is the option to include even more noise via a channel-by-channel multiplication of counts based on randomised sampling of a Gaussian. The function in Equation 3.1 will have a mean of 1.0 in this case, with some width defined by the user. An example is shown in Figure 4.3 for a $^{60}$Co spectrum.

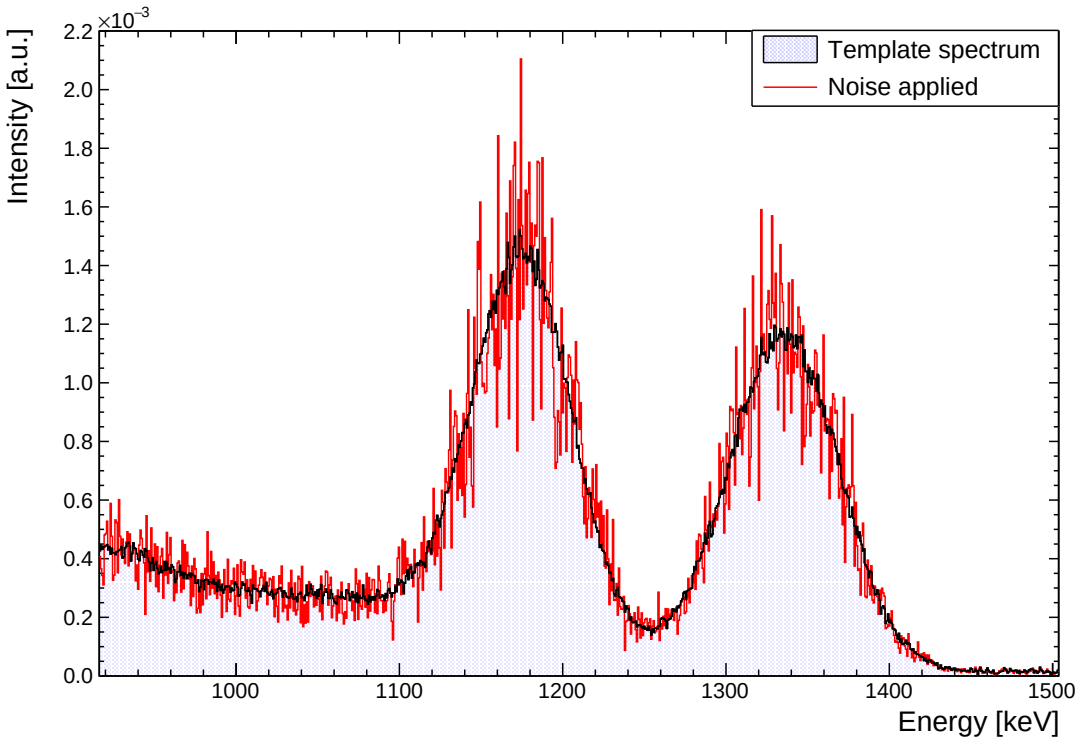



Figure 4.3: Gaussian noise added to a $^{60}$Co spectrum with width $\sigma = 0.2$ as an alternative for emulating poor statistics.

### Gain shifts

One of the most challenging transient effects that every algorithm has difficulty with is gain shift. There are drift corrections that can be applied online or in post-processing [84], but the datasets used in this work are provided to models uncalibrated. The models simply receive a vector of intensities with no information about the corresponding energies. It therefore forces the learning of more general features in the profile of gamma spectra without relying on matching energy to intensities. This goes far in mitigating the fundamental problem of varying energy calibrations, where algorithms such as library comparisons would constantly need to correct the calibration over time.

Simple variation in the applied calibration changes only the energy assigned to each bin. This is a problem for emulating gain shifts, as demonstrated in Figure 4.4. The 2× calibration factor has merely reassigned bin edges to the new energies as expected. Introducing a gain shift for these datasets requires energy to remain both consistent and irrelevant, decoupling the bin and its intensity. Since the model receives an array of intensities, changing the calibration changes nothing. The photopeaks must move position in the array.

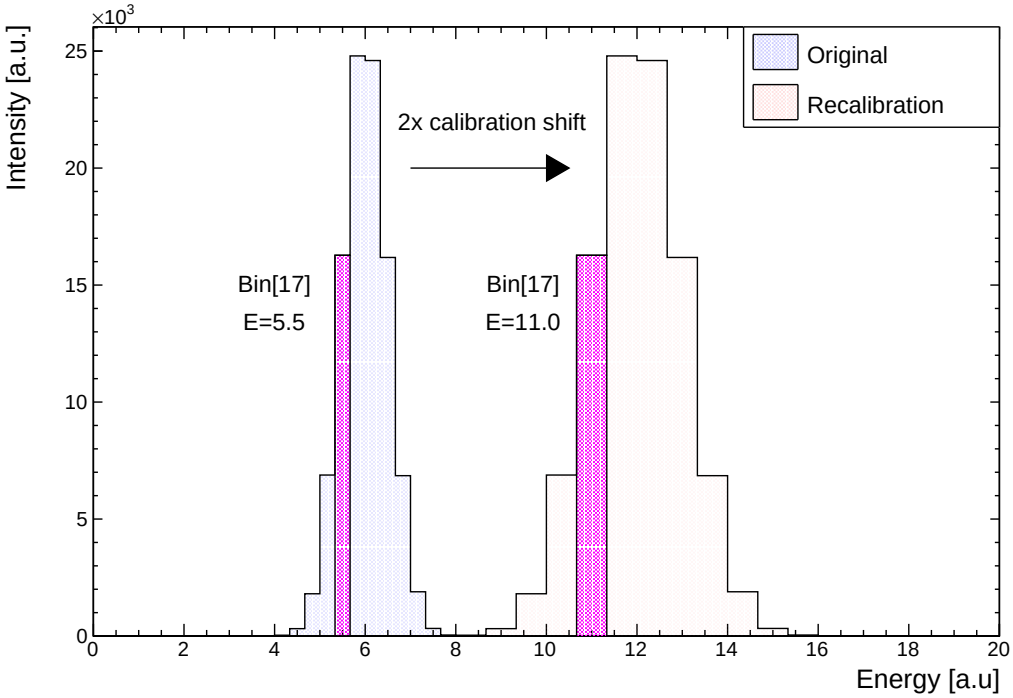



Figure 4.4: Calibration change, altering only the energies assigned to each bin. Bin 17 still contains $N$ counts but is now twice as wide on the same energy scale. This is useless for models that take only the intensities, as the peak has not moved position in the channel array.

In this case, the counts of bin(17) need to be moved to around bin(34) for a 2× gain shift. Shifts are therefore tackled with a rebin and interpolate strategy much like that of Kamuda *et al.* [16].

Re-binning commonly refers to downsampling histograms in factors of two. For example, transforming a 2048 channel spectrum to 512 by summing adjacent channels in sets of four. This is a simple operation is part of ROOT's core functionality. Factors for gain shifts can take any value however, so the intensity of any particular bin must be inferred from interpolation. Through this, the original bin and energy structure is conserved, while only intensities are reassigned. The result of doing so is shown in Figure 4.5.

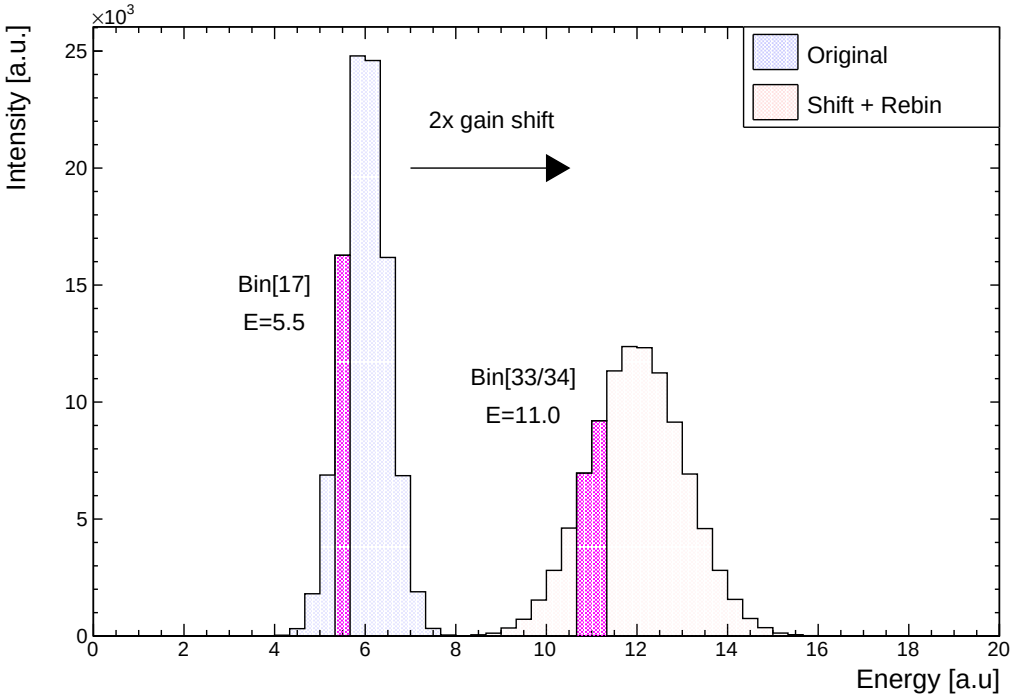



Figure 4.5: Gain shift, maintaining bin structure and moving counts around. The $N$ counts of Bin 17 are now split across Bins 33 and 34 of the same array. Useful for models that take only the intensities, as the peak has now moved position in the array. This also has the advantage of inherently accommodating arbitrary and inconsistent bin structures.

Now, the bin structure remains consistent with the original after applying a gain shift, while the intensities are reassigned within the channel array. The new intensities are scaled by 0.5 in this case since the original counts of bin(17) are being split over bin(33) and bin(34). This is not necessary since normalising will do the same thing, but as a stand-alone operation the rescaling of absolute counts is included for completeness. Of course, these are simple examples, but by interpolating intensities the original bin structure and gain factors may be any arbitrary value, not just factors of two.

Note that it is easier to interpolate from a factor >1.0 since data are pre-existing. A shift <1.0 requires knowledge of features beyond the upper bound of the histogram. This is fine if data are available, when only a 0-3 MeV range is taken from a more comprehensive spectrum, but could otherwise present an issue. For example, there may be no knowledge of the 4.4 MeV from excited $^{12}$C nuclei. For the isotopes used in these datasets it is known that there are no significant transitions beyond the ∼2.6 MeV of $^{208}$Pb, with any summations being minor enough not to create distinguishable features over the experimental timescales used. However, this is something to be taken into consideration for future RIID conditions.

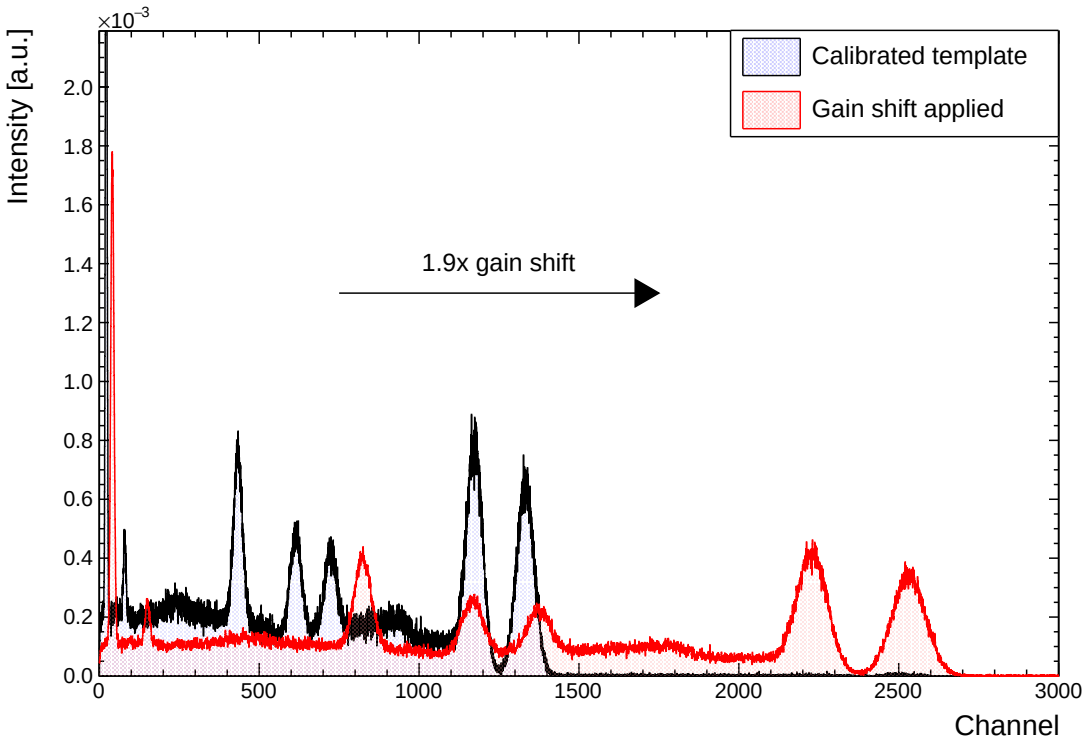



Figure 4.6: An example 1.9× gain shift applied to a multi-isotope spectrum containing $^{108m}$Ag and $^{60}$Co. The 2.0× upper limit for gain shifts was chosen to accommodate the dominant $^{60}$Co peaks, though for spectra such as $^{207}$Bi some more minor high-energy peaks will be lost.

For the 0-3 MeV range chosen for these data, the shift factors are randomly sampled from a uniform distribution in a 0.5-2.0 range. The lower limit allows the low energy $^{44}$Ti and $^{207}$Bi peak pairs to look extremely similar once aggressively re-binned in the final stages, removing any chance of the $^{44}$Ti pair being separable in NaI spectra. The upper limit was chosen to ensure the primary $^{60}$Co peaks remain part of the spectrum, though the summation is of course lost as seen in the example of Figure 4.6. Taking gain shifts to these extremes allows for a true test of a model's ability to make the most what features it can distinguish, learning not to rely too heavily on less consistent features in the overall profile.

### Standardisation

Standardising the input is one of the key tenets of machine learning. While there are ways around this with special network architectures such as the fully convolutional network [85], it is far simpler to enforce a standardised input for both the model and user. The initial step is to aggressively downsample the original 8192 channel spectra with the typical re-

binning operation. The reasoning for this is twofold: computational expense, and feature enhancement.

A 8192 channel structure is often beyond what is necessary for distinguishing peaks over the chosen energy range, especially for NaI spectra. This was chosen to provide excellent template spectra from which the augmented data would be sampled. By performing operations on tensors that have had a dimension reduced by a factor of eight, computational expense is significantly reduced over many epochs that process thousands of spectra. To give an example, the reference model of Chapter 5 was typically trained on a 'small' dataset containing $10^4$ spectra over 150 epochs. Secondly, spectra containing relatively few counts render many features virtually indistinguishable from noise. To help these spectra it is beneficial to cluster sets of bin intensities into single values that enhance prominent photopeaks, and smooth out the surrounding noise. An example of this is shown in Figure 4.7.

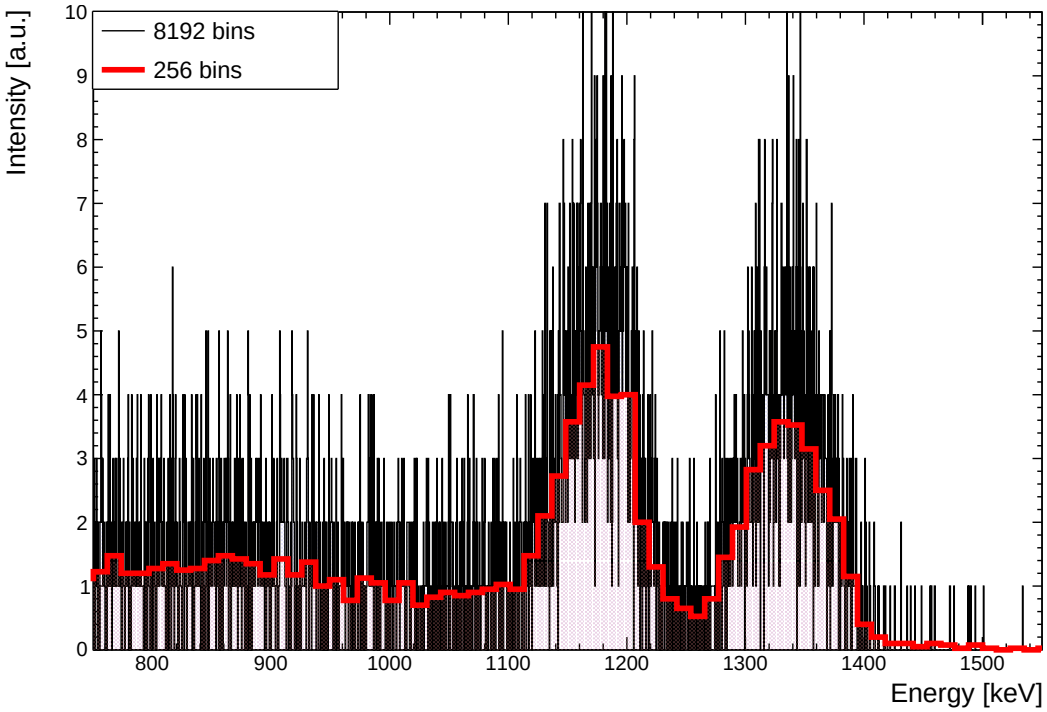



Figure 4.7: Fairly extreme example of re-binning $^{60}$Co spectrum taken with a NaI detector. With only 8000 counts in total, the original structure is noisy, even in such a simple spectrum.

The final step is to normalise the intensities by total counts. It becomes far easier to compare profiles on the same scale, and the vast majority of machine learning models perform better when dealing with data close to zero [14]. These are stored as floating point values to marginally reduce storage requirements.

### 4.2.2 Handling large datasets

Originally the training spectra were all loaded into memory for use in the model. Of course, even with small datasets the $\sim 10^3$ intensities in each sample quickly produces an extremely large array. On top of this, another dimension is required for storing the target outcomes (i.e. which sources are present in the spectrum). This is, after all, a supervised learning problem. A solution was therefore required that would load training data in manageable batches, and keep track of associated labels.

#### Label preparation

The most convenient method to maintain knowledge of the labels and cut down on memory usage is to change how data are provided to the model. As the next section describes, spectra may be loaded as-needed while the model trains. If spectra are stored as individual files, then only the name of the spectrum and a list of the sources present need to be readily available in memory. A unique number is assigned to every sample of unique source combination when compiling datasets. Spectra are stored not only collectively as ROOT objects, but also individually in ASCII files. The names of any ROOT object or file are all of the format where a unique identifier is followed by a delimited list of the sources present. Everything the model needs to know about target outcomes can therefore be derived from just the file names.

In preparation for training the model, directories containing data are queried and two python lists created: one for the path to each spectrum, and one for the associated sources present. This translates to having a list of file paths and a list of target labels. For example, the filename `'100_Ag-108m_Co-60_Ti-44'` allows simple string operations to make a label list of `['Ag-108m', 'Co-60', 'Ti-44']`. Consistency is ensured when modifying either list. Both are then randomly shuffled to ensure each batch is not just a collection of similar spectra, and to reduce over-fitting problems.

The file and label lists are kept separate because they have very different purposes and it is conceptually easy to do so. The file path list is used in a generator to load spectra in small batches. The label list is used as an array of target outcomes for the model.

In this work, the problem may be framed as that of a multi-class, multi-label classification task. For such a task, the model output is generally a 0-1 value for each label that indicates the probability of a source being present in a spectrum. The model therefore needs to compare to a binary target of 1 (present) or 0 (not present) for each source, hence the use of a binary cross-entropy loss function (Eq. 2.8).

Every entry in the target label list is 'one-hot' encoded with a multi-label binarizer. Take the following example set of five possible classes:

```
['Ag-108m', 'Am-241', 'Cs-137', 'Co-60', 'Ti-44']
```

An array of five binary indicators is created for each label list entry, with '1' indicating the presence of a particular label. A spectrum containing only $^{241}$Am is therefore `[0 1 0 0 0]`. If the sources present were $^{241}$Am, $^{60}$Co, and $^{44}$Ti, the label becomes `[0 1 0 1 1]`. After the binarizer object is created it becomes trivial to switch between human readable text and the model comparable binarized forms. Once prepared, both the file paths and binary labels are passed to a custom batch generator.

### Batch generator

A python generator allows for a function to be declared that will behave like an iterator. In the context of this work, this means that training spectra may be fed to the model in real time as needed. Tensorflow keras has a similar object called `Sequence` that does much the same, with the benefit of being far safer for multiprocessing. This is due to additional backend that guarantees the model will only train once on each spectrum in the training set over the course of an epoch.

A custom 'batch generator' class was written using the sequence object. The shuffled file path list is provided in batches of typically 32 spectra. The training spectra and associated target labels are stored in multidimensional arrays, with spectra read from the file paths provided. This array is reshaped to a `(32, 1, 1024, 1)` tensor for compatibility with model input conventions, the sizes corresponding to `(batch size, height, width, depth)`. This changes the loss function slightly, so more rigorously a cost function ($C$) is now being used over the batch rather than the loss of a single sample. This is simply the average loss over the batch of $n$ samples,

$$C = \frac{1}{n} \sum_{i=1}^{n} L(y_i, p_i). \tag{4.2}$$

For efficiency the batch generator is run in parallel with model training. To ensure there is no delay while batches are processed, a buffer of several batches is loaded in a small queue. Returned data are a set of spectra to train on, and the associated target labels for the batch.

### Extended Usage

With everything now only dependent on lists of files names which are to be included, there is no reason why the training spectra must come from a single dataset. This makes it very simple to selectively choose subsets of data so they can be mixed and matched as needed, as shown in Figure 4.8.

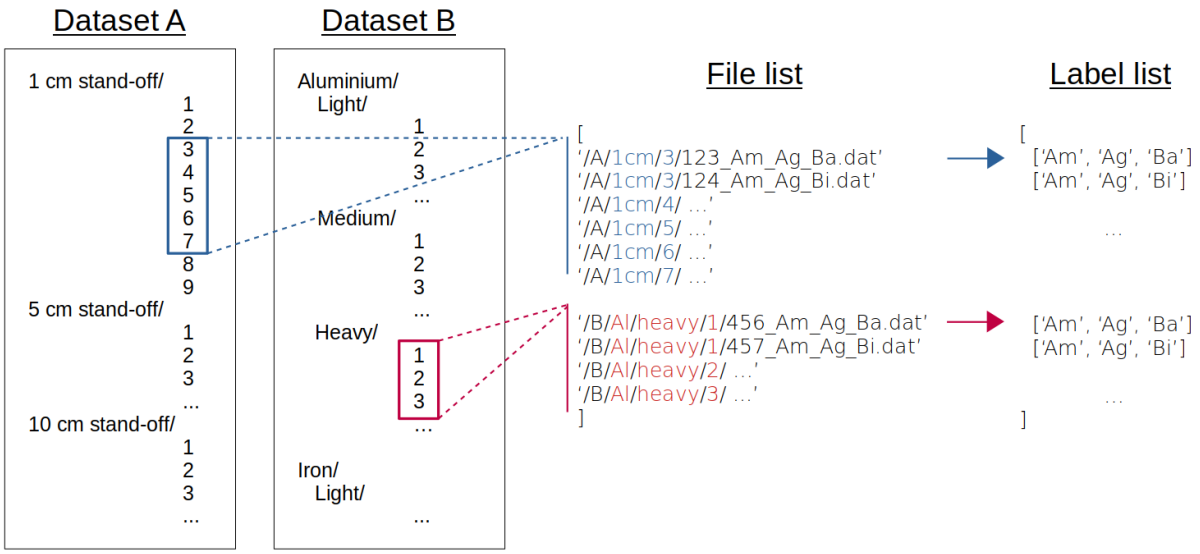



Figure 4.8: Example selection of arbitrary datasets to build up customised training and test sets. Samples are stored individually in the core datasets, with subsets chosen by specifying paths in the *files paths* list. This is then used to generate the associated *labels* list.

In the example here, subsets of distinct datasets may be brought together. The example takes spectra from Set A at 1 cm containing 3-7 isotopes, and combines them with samples of spectra from Set B of heavily shielded conditions that contain 1-3 isotopes. Since all string handling and subsequent binarization occurs after a file path list is generated, the only manual input required is to arbitrarily choose the subsets through common interfaces. Funnelling everything through the list system scales well to being able to choose any collection of subsets many dataset libraries that may be available, and allows datasets to remain modular. This is especially beneficial when there are new data to consider, or if calibrations vary between datasets.

The simulation environment has been developed and to address the forward problem, creating accurate spectra for various RIID conditions. In this chapter, techniques were developed to scale up datasets from a template library, preparing them for machine learning models. The next chapter discusses the final stage of this research, where the reverse problem of identifying unknown sources from gamma-ray spectra is addressed.

# Chapter 5

# Radio-isotope identification models

It is evident that to get to this stage from nothing, a significant amount of preparation was required. By reviewing historical approaches to RIID (Sec. 2.2.4), it was clear the challenges were primarily in handling transient effects that would alter the profile of a spectrum or change the calibration. This led to machine learning approaches based on ANNs. These rapidly grow in complexity and consider all channels independent, so it was recently suggested that CNNs may provide a better alternative [54]. Promising results were found after independently testing very basic CNNs, so they became a focus of this work.

A generalised gamma simulator was developed using GEANT4 so that a diverse range of RIID conditions could be tested in a rapid development environment (Chap. 3). This included the flexibility to easily swap out detector types and change simulation geometries, while having access to a wealth of information not easily obtained by the user in more rigid frameworks such as MCNP and GADRAS. Timing information became crucial to accurately reproducing true coincidence summations, and enabled the simulation of coincidence measurements with multi-detector arrays.

It was apparent that the large data sets required for effective deep learning were impractical to obtain experimentally or even simulate, so data sets were built up from a library of template data that represent perfect examples of various conditions (Chap. 4). By understanding how the problem must be framed for CNNs, data sets were converted into formats that the models can understand, while providing the binarized target outcomes needed for supervised learning. All that remained was to develop these networks for RIID, and the initial approach was to use a sequential CNN. For most of these investigations, the focus was on spectra from NaI detectors that have relatively low energy resolution, due to their prevalence across many RIID applications.

#### Sequential CNN

A sequential CNN is structured as a stack of layers, where each layer has one input tensor that maps to one output tensor. There are more advanced models such as the inception model [86], where an output is shared between multiple operations and recombined later. For exploratory investigations, a reference model with the basic sequential architecture provided a baseline. This became a foundation for further development, making model prototyping much quicker.

The sequential reference model used is slightly modified from that of Figure 2.14. The model is broken up into modules for performing convolutions, where the features are learnt, and a final classification module that makes the predictions.

The TensorFlow [55] libraries were used throughout this work due to the excellent higher level functionality that made prototyping and analysis far more accessible. The functional Application Programming Interface (API) for TensorFlow allows everything to be written in Python. This further streamlines the process, making available the excellent functions of the `scikit-learn` [87] modules for performance analysis. Unfortunately, these functions are tailored to images, so analysis tools had to be modified because of the special 1D case and extended to accommodate this multi-class, multi-label solution. These codes were moved to the BlueBEAR High-Performance Computing (HPC) servers at the University of Birmingham to train multiple models simultaneously.

## 5.1 Reference data sets

It is important to understand the data, so the relevant information on the data sets used for training and testing is outlined in Table 5.1. Here, the training data are entirely simulated to accommodate an often overlooked challenge that may occur during deployment. A new system may not have enough, or indeed any, real spectra to learn from. While a few samples can be collected quickly, it would be of great advantage to be able to reliably train on purely simulated data. This would provide a working system from the outset, with easy incorporation of new or complex conditions. Further training would be carried out to refine performance once the system was in place, once on-site experimental data becomes available.

Section 4.2.1 details the implementation of the various operations listed in Table 5.1. The reference set used contains spectra from NaI detectors. These have relatively low energy resolution, but are by far the most widespread detector used for RIID. Most importantly, all spectra are provided to models with no calibration, passed simply as a vector of intensities.

Table 5.1: Scope of the training and test sets.

| Parameter | Training set | Test set |
|---|---|---|
| Type | simulated | experimental |
| Total counts sampled | $10^3$ - $10^6$ | $10^3$ - $10^6$ |
| Total sampling method | logarithmic | logarithmic |
| Background cap | 25% of total counts | 25% of total counts |
| Gain shift | 0.5 - 2.0 | 0.5 - 2.0 |
| Low energy discrimination | <40 keV | 15 - 40 keV |
| Sources in multi-nuclide spectra | 1 - 9 | 1 - 9 |
| Samples per unique combination | 20 | 20 |
| Number of channels | 1024 | 1024 |
| Training/validation split | 50/50 | N/A |

It should be noted that the background included in both sets is experimentally obtained. The reasons for this are discussed in Section 3.2.3, detailing compromises made for the sake of generalisation. The background contribution is capped at a fraction of the total counts so as not to unnaturally dominate spectra, though 25% is still an exaggerated contribution. It is ensured that every unique combination of possible sources is present in the full set. The training set is randomly shuffled and split into training and validation sets. With smaller sets it is therefore possible to not train on certain combinations. The 50/50 split means that there is an average of just 10 samples for every unique multi-isotope combination.

The test sets are entirely generated from experimental spectra, and have much the same properties as the training sets, including the extreme gain shifts. There is a natural threshold at ∼15 keV due to a low level discrimination set for the ADC. One important point is that the simulated spectra are imperfect. While some backscatter from the detector itself is present, these spectra contain no additional backscatter from surrounding materials that will often occur in the experimental spectra. Again, this is a choice made to emulate the challenges of practical deployment, with simulations favouring generalisation over the kind of accuracy that is possible with bespoke solutions.

Every model variation to be discussed is trained from scratch on four separate data sets (A-D), each generated with a different random seed. In smaller sets especially, this could end up having the effect of providing a poor representation of the associated test sets. In general, the more comprehensive a training set, the better the performance seen. Six test sets are evaluated for similar reasons, with the average taken to represent all performance indicators. For brevity, it should be assumed this is the case unless stated otherwise.

## 5.2 Hyperparameter trends

'Hyperparameters' are all of the higher level variables that relate to the structure and training of a model. The number of convolution layers, filter sizes, and initialisers used are all examples of hyperparameters. Assessing how performance varies as these are changed is a fundamental part of building a model from scratch. Manually investigating these allowed for a deeper understanding of how the model interprets the data. Once trends are known, it becomes more viable to define sensible limits and automate the optimisation.

Note that the following investigations were carried out with the 10 cm stand-off point source data for a NaI detector. This distance could be considered the data set with fewest complications since there is no shielding, and any true coincidence summations are very minor.

### 5.2.1 Set-up and structure

For reference, a basic convolution module is repeated here in Figure 2.12. The relevant hyperparameters are: the number of convolution layers, number of filters, filter size, dropout rate, activation function, and the pooling type, length, and stride.

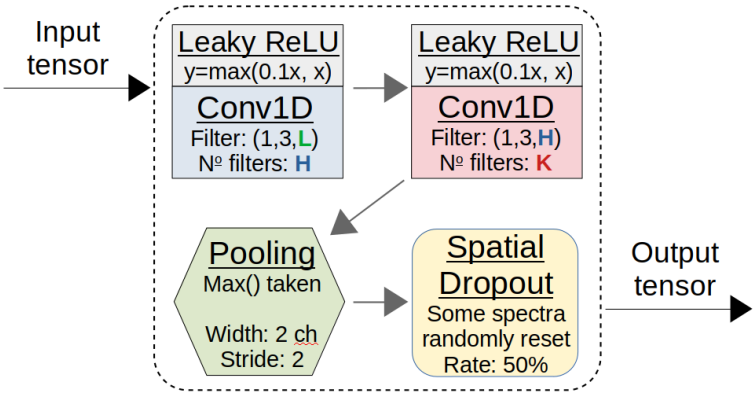



Figure 5.1: Representation of the convolution module. This takes the $L$ entries of the previous feature map, and trains first $H$ filters, then $K$ filters in these two convolution layers. Pooling and dropout reduce the dimensionality and combat over-fitting. For demonstration purposes, the shapes exclude a dimension for batch size.

These provide many degrees of freedom for changing the structure of a model, making the possibilities seem endless. This only expands further with the extra hyperparameters of the classification module shown in Figure 5.2. The module contains possibilities for: the number of dense layers, dense layer sizes, activation functions used, and the dropout rates.

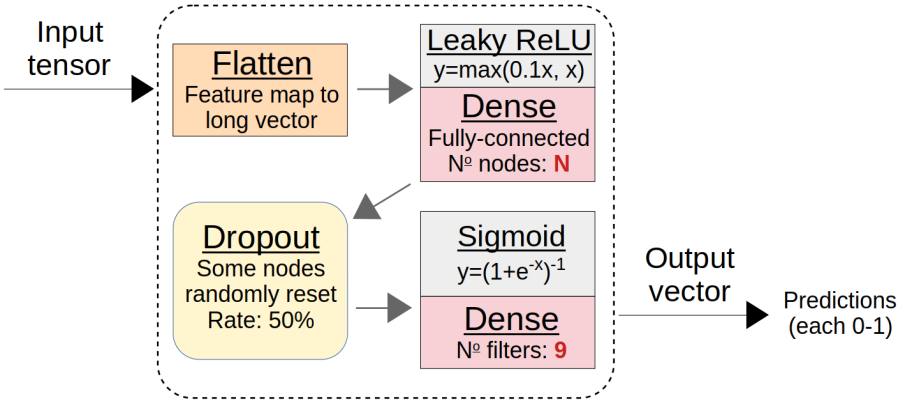



Figure 5.2: Representation of the classification module. The convolution module feature map is flattened into a vector. A small 'fully-connected' ANN makes final predictions of present sources, converting outputs to a 0-1 probability via the Sigmoid activation.

The standard performance metrics discussed in Section 2.3.3 are excellent indicators for how models perform on a data set as a whole. These have all been extended to apply to the multi-label case of this research. However, these take the rates across all possible labels into consideration. Given the RIID applications, the performance on individual spectra is also extremely important. The 'perfect' prediction rate is therefore also used in assessing a model's RIID effectiveness. This became a very useful metric since many models would often predict just one of the labels incorrectly, which has little effect on the big picture. An example of this is the model of Table 5.2.

Table 5.2: Example of overall classification outcomes for a model with a perfect prediction rate of 31.7(5)%. The vast majority of labels are still correctly predicted, making it difficult to use these more traditional performance measures when comparing models.

| | | Actual label | |
|---|---|---|---|
| | | 1 | 0 |
| Prediction | 1 | 94.5(1)% | 18.1(1)% |
| | 0 | 5.5(1)% | 81.9(1)% |

The model here has an average precision of 95.6(1)%, and as the TP and TN rates demonstrate, it is fairly effective at correctly predicting the presence of any one source. By all accounts, this is a good model. Unfortunately, a perfect prediction rate of 31.7(5)% means that only a third of the spectra have the full set of nine sources predicted correctly. This alternate perspective on performance for RIID provides a metric that is much more sensitive to hyperparameter changes. Other metrics typically remain within uncertainties to make model comparisons difficult.

### 5.2.2 Training hyperparameters

Preliminary testing found that a good sequential CNN for initial prototypes contained three convolution modules, followed by a classification module with two dense layers. The 50/50 split of training and validation data meant that each model only trained on an average of 10 samples per unique combination of sources. As a small set, this helped reduce training times for quicker prototyping. Interestingly, the split of these sets had little effect on performance until it became more extreme. A validation size between 50% and 30% of the full training set had a perfect classification rate of ∼73%, but dropped significantly to 46.8(9)% when just 20% was used. This illustrates the need for having enough validation samples to make a meaningful contribution to the learning process, hence the 50/50 split used throughout.

On the subject of learning parameters, the batch size is another consideration. As discussed, to accommodate arbitrarily large data sets, the spectra are loaded in batches. The size of these batches is known to sometimes have an effect on the performance of a model, and there is a poor practice known as 'batch hacking' where only the optimal size is chosen. This is a problem for generalisation since it becomes very specific to a particular data set. This is largely mitigated by using the average of many data sets for both training and testing, as in this analysis, but the rates were investigated anyway.

Table 5.3: Classification rates for a range of batch sizes.

| Batch size | Perfect rate |
|---|---|
| 32 | 72.0(7)% |
| 64 | 73.1(8)% |
| 128 | 74.7(7)% |
| 256 | 74.6(6)% |
| 512 | 70.7(7)% |

Table 5.3 shows that for this model architecture and data, there is fortunately very little variation over a large range of the usual batch sizes. A batch size of 32 spectra is queued 10 batches at a time while the model learns, keeping the memory requirements to a reasonable minimum.

When training brand new models, it is the user's prerogative to choose starting conditions. The kernel initialiser sets the initial values of all convolution filters and weights, randomly sampling from various distributions that have developed over the years. In all cases, the values are small and close to zero, and some are better matched to different activation functions.

Table 5.4 shows the result of investigating those readily available, the details of which may be found at Ref. [88].

Table 5.4: Classification rates for a selection of initialisers. Those marked with an asterisk (*) failed to converge for at least one of the sets.

| Initialiser | Perfect rate |
|---|---|
| Glorot normal* | 68.9(8)% |
| Glorot uniform* | 71.1(5)% |
| He normal | 73.8(9)% |
| He uniform | 74.3(11)% |
| Lecun normal | 72.8(9)% |
| Lecun uniform | 70.3(7)% |
| normal* | 65.9(3)% |
| ones | 73.7(7)% |
| orthogonal | 72.2(7)% |

All initialisers were used, with some untabulated alternatives failing to converge for all four test sets. These results were equivalent to a coin toss, with each label chosen at random. Those marked with an asterisk (*) converged for most of the test sets, but are unreliable for this model structure as they fail to do so on at least one of the sets. The He initialisers [89] are generally considered to be well matched to ReLU-based activation functions. He normal initialiser weights are sampled from a truncated normal distribution that is centred on zero. The variance factor is double that of the very similar Glorot initialisers, taken as $var(w_i) = 2/n$, where $n$ is the number of inputs in the weight tensor.

Optimisers have developed to the point where there are two clear frontrunners that are typically the go-to algorithms. These are Adam [90] and RMSProp [91], both of which were tested when compiling these models. Perhaps as expected, they perform similarly with average perfect classification rates of 70.3(7)% and 67.1(7)% respectively, across the four test sets.

It is worth noting that for all of these investigations the number of epochs was kept at a consistent 150 epochs as far as possible. It is possible to set up 'early stopping', which attempts to regulate overfitting by ending the learning process before the predefined number of epochs. For example, terminate the training early if the loss has not improved for several epochs. This addresses the problem of training for far too long, which leads to overfitting, but it was found that some runs would prematurely end when investigating extremes due to any large fluctuation in the early iterations. Since these investigations are very similar in their optimum, it was initially more productive to fix the number of epochs.

### 5.2.3 Core hyperparameters

The convolution filter should be matched to the data, with $3 \times 3$ filters being very common for images to find smaller details. For gamma spectra from NaI detectors, the resolution is such that the main features are spread over many channels. For example, a 3-channel edge detector looking at a peak spread over 50 channels begins to find patterns in the noise. This is why input samples are best standardised, so parameters like this can be better matched to the data. In this case, all spectra are rebinned to 1024-channel spectra to consolidate intensities and smooth out the noise. For NaI spectra, even the low energy peaks are still several channels wide. Table 5.5 demonstrates the result of using various filter widths. Only at the low end is there a significant impact on performance. Since computational expense increases with size, smaller filters are still better to have where possible.

Table 5.5: Classification rates for a range of convolution kernels and numbers of filters. For both, it is only at the lower extremes that performance suffers, quickly stabilising for increased computational expense with no additional benefit.

| Convolution filter size | Perfect rate | № of filters | Perfect rate |
|---|---|---|---|
| 1 | 15.3(6) % | 5 | 41.7(8)% |
| 3 | 65.6(7) % | 10 | 58.6(7)% |
| 5 | 73.4(8) % | 15 | 63.2(8)% |
| 7 | 72.6(9) % | 20 | 64.7(7)% |
| 9 | 72.4(8) % | 30 | 71.2(9)% |
| 11 | 75.0(10)% | 40 | 71.0(7)% |
| 13 | 70.1(9) % | 50 | 73.3(7)% |
| 15 | 73.5(8) % | 75 | 72.7(7)% |
| 17 | 70.0(8) % | 100 | 72.6(8)% |
| 19 | 70.9(8) % | 150 | 71.5(7)% |

Intermediate activations exaggerate weights to make features more expressive. These activation functions introduce non-linearity into the model, with Rectified Linear Unit (ReLU) based functions ($y = max(0, x)$) chosen over legacy functions such as $tanh(x)$ due to their robust nature. With the ReLU function, the zeroing of negative weights makes it unlikely for any given node to activate. This is desirable because only the meaningful aspects are being processed, leading to a concise model that is less prone to overfitting. To provide an analogue, not every neuron in a biological network is constantly firing for every stimulus.

The downside to this is that nodes can become stuck on the negative side and are constantly being set to zero. This is known as the 'dying ReLU' problem and can lead to a lot of the network not doing anything. To counteract this, the 'Leaky' variant may be used instead.

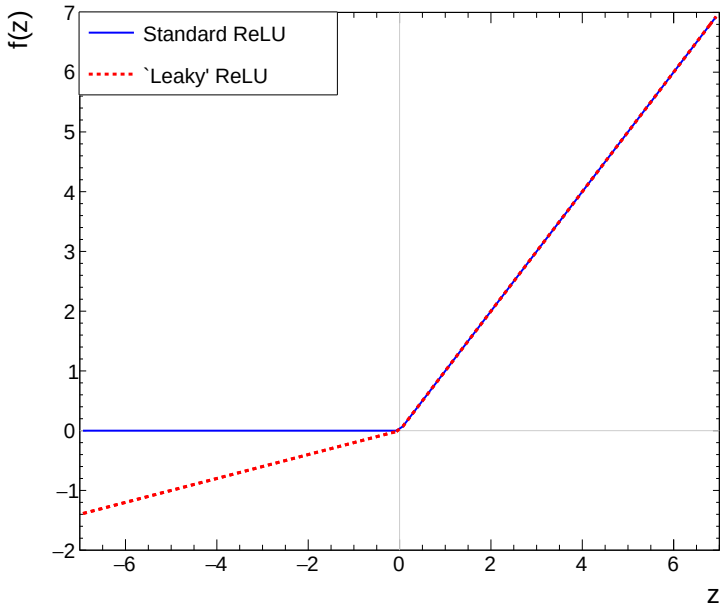



Figure 5.3: Comparison of the standard Rectified Linear Unit (ReLU) activation function with its 'leaky' variant. This adds a small gardient in the negative region to counter the 'dying ReLU' problem. The gradient ($\alpha$) is 0.2 in this case.

Figure 5.3 demonstrates the small difference from the standard ReLU activation function. This introduces a small gradient ($\alpha$) as $y = max(\alpha x, x)$ for negative values. Both variants were tested and showed consistent perfect rates of 72.0(7)% and 72.5(5)% for leaky and standard ReLU respectively. The leaky variant was eventually chosen to ensure all nodes are potentially active while minimising the number of filters required. Of course, this is not always the case, and there are other variants such as Exponential Linear Unit (ELU) [92] and Concatenated ReLU (CReLU) [93] that modify this further for addressing more minor issues, but this was the most efficient solution for RIID. It was found that the range of 0.01 to 0.5 for $\alpha$ had little effect on the overall classification rates.

The 'max pooling' operation down-samples convolved spectra by propagating only the maxima of non-overlapping channel groups. This reduces the size of output tensors while retaining important features. Given the smaller scale of 1-dimensional inputs that had already been rebinned down to 1024 channels in pre-processing, there was no reason not to use the minimum. A pooling operation with a size and stride of just two channels halves the size of output tensors, greatly reducing dimensionality to improve performance.

Since pooling already quite aggressively downsamples the tensors, padding was left as 'same' to include information from the spectrum edges. This means that when a three-channel

filter is at the edge of the spectrum, an extra intensity of zero is included in a dummy channel for the filter to work with. Addressing features at the extremes becomes especially important when there are gain shifts, where it is desirable not to lose data at the periphery of the spectrum.

Spatial dropout follows the pooling operation, removing spectra from feature maps at random to prevent over-dependence on any particular path for a final classification. As the dropout factor increases, more features are dropped from the feature maps, making it far more difficult to train.

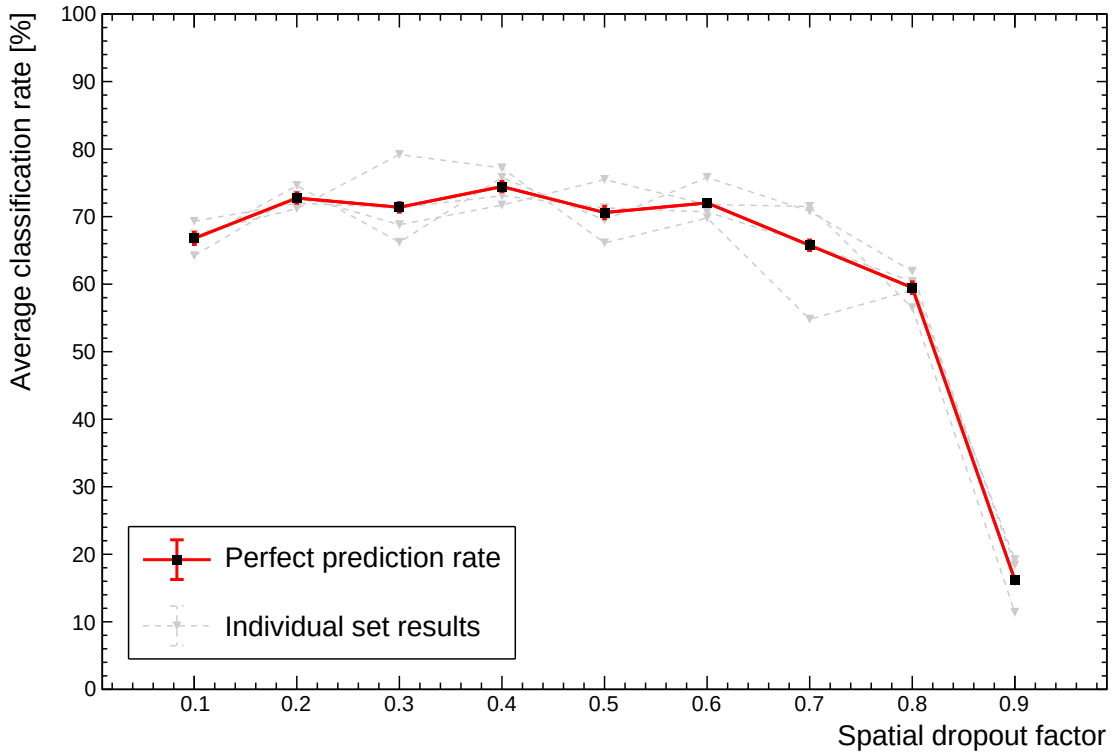



Figure 5.4: Perfect classification rate performance as the dropout rates are varied in the convolution modules.

This is exactly the trend seen in Figure 5.4, where the rates suffer greatly from randomly dropping too much information. Below 0.7 appears consistent but an issue is seen with smaller factors, where the learning curves indicate overtraining. The result was a choice to use the 0.5 middle ground as a reasonable compromise.

With the convolution modules investigated, only the hyperparameters of the classification module remain. It is here that the most dramatic effects can occur. After the tensor of the final convolution module is flattened into a vector, classifications are decided through a small

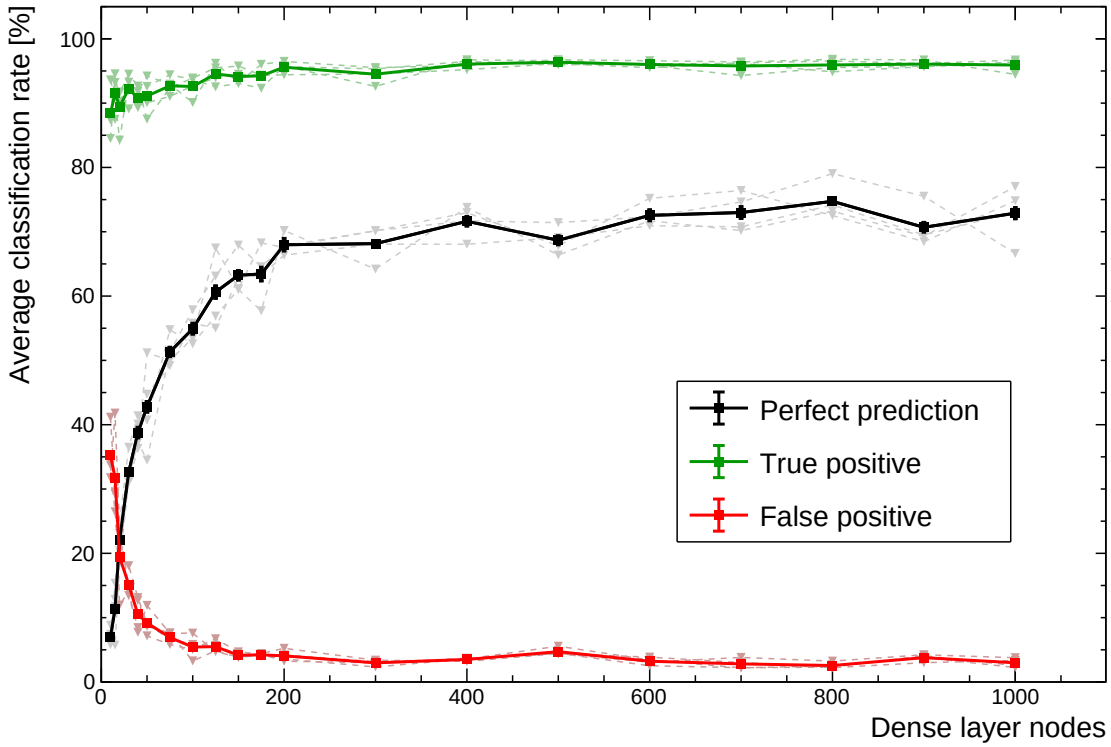



Figure 5.5: Representation of the classification module.

ANN. While the second dense layer of Figure 5.2 must have nine nodes (one for each source), the first can vary and will have significant impact in comparison to the other hyperparameters. Given the computational expense of fully-connected layers, the classification module is limited to just two layers. Figure 5.5 demonstrates how important the small ANN is to making good use of the features extracted by the preceding convolution layers.

There is a clear degradation in performance when using an extremely small ANN. However, diminishing returns with size encouraged a compromise, since these dense layers rapidly increase in complexity with scale and become computationally expensive. One interesting note is that the drop in perfect performance rates are primarily due to false positives. While not ideal, it is far better that this is the case for RIID applications. False positives will be inspected, whereas false negatives may permit important material to remain undetected.

Finally, the dropout rate between the two dense layers follows the same trend as those seen for spatial dropout rates in the convolution module. Averaging $\sim$72% for perfect rates, this falls off after 0.7 to reach 44.9(8)% for a 0.9 dropout rate.

## 5.3 Model performance

Table 5.6 shows the important hyperparameters of the reference model. With these parameters fixed, a final evaluation of performance may be carried out. Of course, there is a huge range of potential architectures due to the many hyperparameters available, but a model is always needed as a baseline for performance.

Table 5.6: Relevant reference model hyperparameters.

| Parameter | Value |
|---|---|
| Optimiser | Adam |
| Initialiser (kernel, bias) | He normal, zeros |
| Activation function | Leaky ReLU ($\alpha = 0.1$) |
| Batch size | 32 |
| Dense layer nodes | 700 |
| Dropout (spatial, dense) | 0.5, 0.5 |
| Filters per Conv layer | 40 |
| Filter size | 5 |
| Pooling type | Max pooling (length 2, stride 2) |
| Padding | Same |
| Loss function | Binary cross-entropy |

Using these hyperparameters on the NaI detector spectra at a 10 cm stand-off distance, the overall performance of the reference model was assessed. All metrics are again the average of four separate simulated training sets, each evaluated against six test sets that contain only experimental spectra. All test sets have 20 examples for every possible unique combination of sources since there is no splitting of the data.

The average perfect prediction rate is 74.4(9)%. A precision of 99.61(3)% reflects very few of the individual labels being predicted incorrectly in the remaining spectra, and indicates that the correct predictions were made with a high degree of confidence. The confusion matrix of Table 5.7 sums up the overall binary rates across all labels.

Table 5.7: Classification outcomes for the reference model. These values are averaged across test sets of experimental data taken using a 3"×3" NaI detector at a stand-off distance of 10 cm.

| | | Actual label | |
|---|---|---|---|
| | | 1 | 0 |
| Prediction | 1 | 96.5(1)% | 3.1(2)% |
| | 0 | 3.5(1)% | 96.9(2)% |

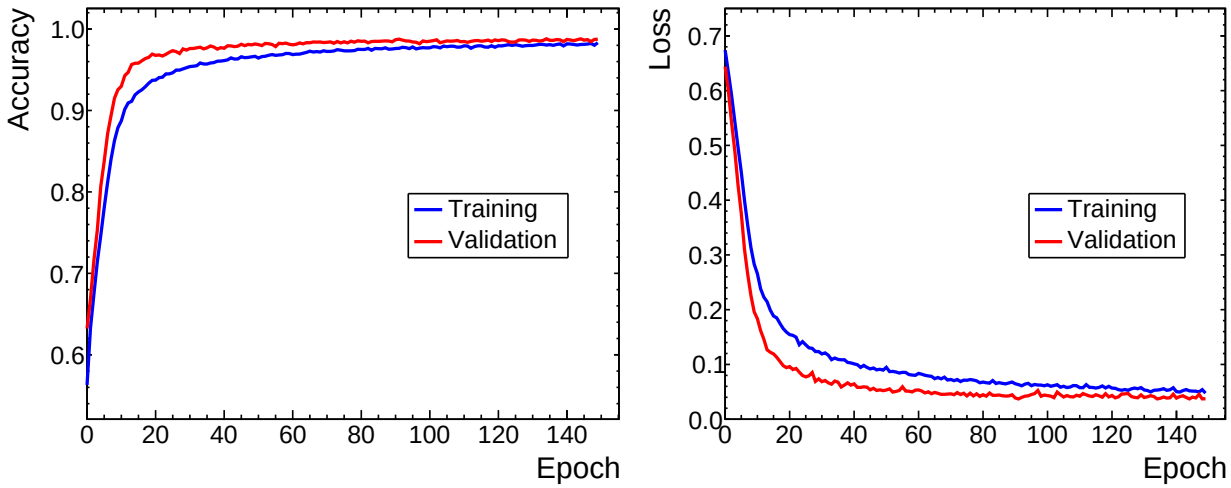


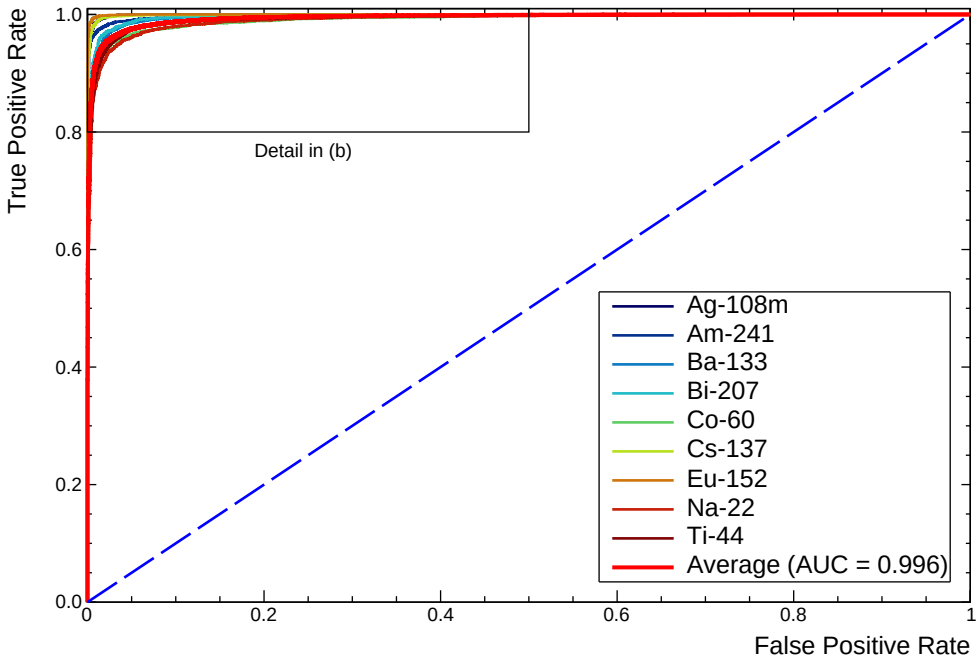

Figure 5.6: Learning metrics from training the reference model over 150 epochs, showing good conversion of the training and validation sets.

The training of the reference model over the relatively short 150 epochs shows good convergence of the training and validation sets (Fig. 5.6). There is no overfitting, characterised by the validation loss reaching a minimum before increasing while the training set continues to improve. It may be said that the validation set is slightly too easy to predict as the losses are not perfectly converged, but it is enough to see no evidence of overfitting, and that the validation and training sets are well matched.

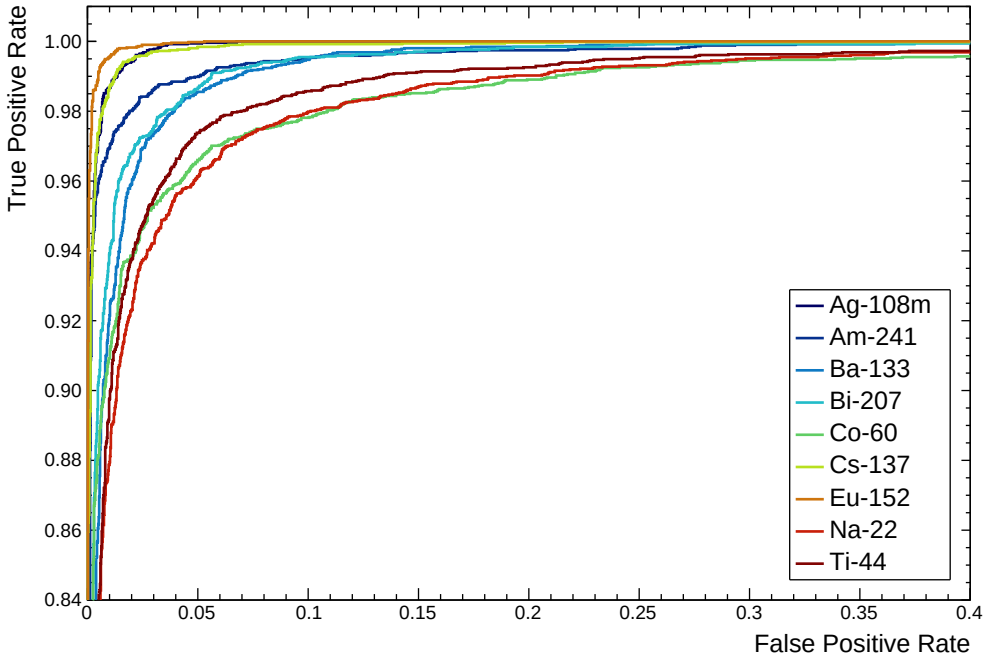
The average time to load spectra, reformat for the model, and make predictions was $<5$ ms per test spectrum, making real-time updated classifications readily available to a user if necessary. With efficient data structures where spectra are not read in from a text file, this can certainly be improved for a commercial system.

Individual binary confusion matrices may be created for each source, but the clearest and most concise visual aid for observing the performance of the nine classes is the Receiver Operating Characteristic (ROC) (see Section 2.3.3). For RIID, the multi-label ROC curves have proven an extremely useful tool for quickly recognising the sources that different models find to be a challenge, and offer insight into what the model has learnt to look for.

Figure 5.7 (a) shows a full ROC extended to the multi-label case. Each curve tends toward (0,1) as performance improves, with the dashed line representing random assignment of positive or negative classifications.




(a) Full ROC with averaged curve.




(b) Detail of ROC from *(a)* showing individual sources.

Figure 5.7: ROC curves extended to the multi-label case for the reference model. Provides comparisons of performance for each source as the decision threshold is varied. The average (red) is included in 5.7a. Figure 5.7b shows $^{22}$Na, $^{60}$Co, and $^{44}$Ti as marginally responsible for more false classifications. Dashed blue in (a) represents random label assignment.

Inspecting the detail in Figure 5.7 (b) identifies $^{44}$Ti, $^{22}$Na, and $^{60}$Co as being marginally responsible for more false classifications. Slight differences in AUC confirm this quantitatively. A $^{22}$Na spectrum in particular contains very few features, all of which closely overlap with those of many other sources to make distinction difficult (Figure 5.8). Finding distinctions between $^{207}$Bi and $^{44}$Ti is also challenging as they are extremely similar in profile, especially with the potential for large gain shifts and very few counts to work with.

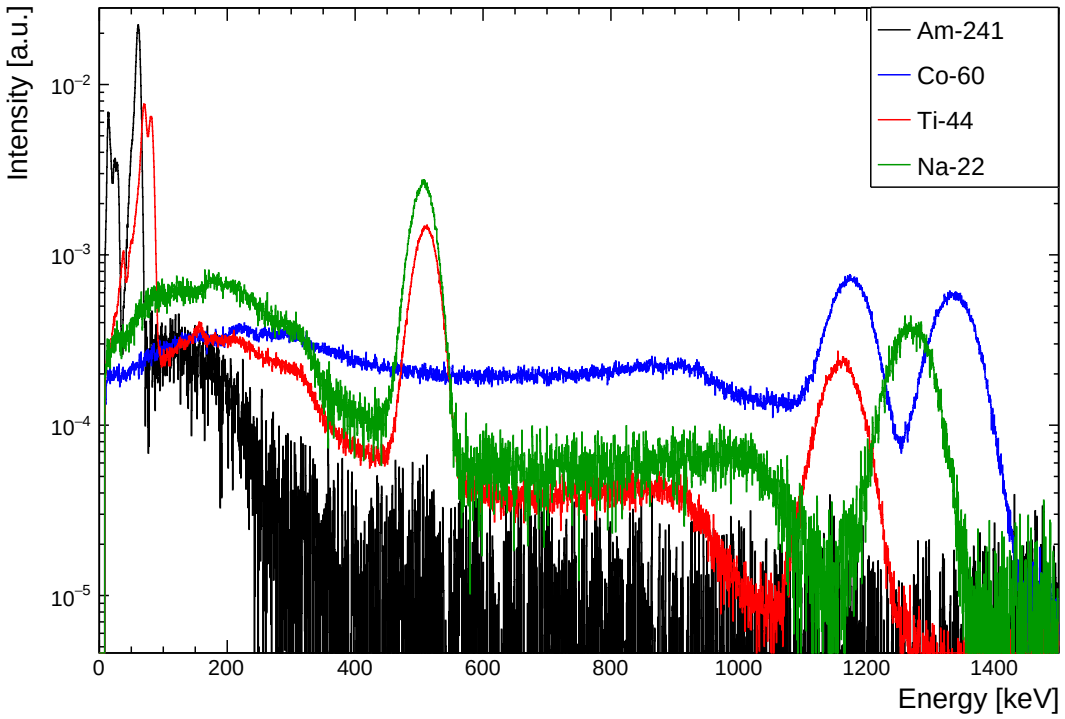



Figure 5.8: Experimental $^{44}$Ti, $^{22}$Na, and $^{60}$Co spectra, showing the difficulty in resolving features. The $^{241}$Am feature is large enough to stand out from close peaks of other sources.

Much like $^{22}$Na, the gamma emissions from $^{60}$Co are such that the two greatest features overlap with those of multiple other sources, as Figure 5.8 demonstrates. This makes it extremely difficult to identify their presence, especially when $^{60}$Co has a lower activity compared to others in a multi-isotope spectrum.

The $^{241}$Am predictions are surprisingly accurate given the that the source contributes just one photopeak of significance. Because all of the counts go toward a single peak at 59.5 keV, it remains a distinct feature even with a relatively low activity. This is only aided by being spread over fewer channels as a low energy peak, and that it does not overlap with any significant features of other sources.

## 5.4 Challenging conditions

The previous sections have all used the 3"×3" NaI detector at a stand-off distace of 10 cm. The most challenging conditions identified in the aims of RIID, and where other models often fail, is with shielding and very close geometries.

### 5.4.1 Heavily shielded conditions

Attenuation of $\gamma$ rays through shielding is exponential with thickness. Lower energy photons (<100 keV) become increasingly more likely to interact, affecting them more severely. Backscatter, where a photon Compton scatters in the shielding material and then into the detector, also changes the profile significantly. The combination of these is what makes shielded conditions one of the biggest challenges in RIID.

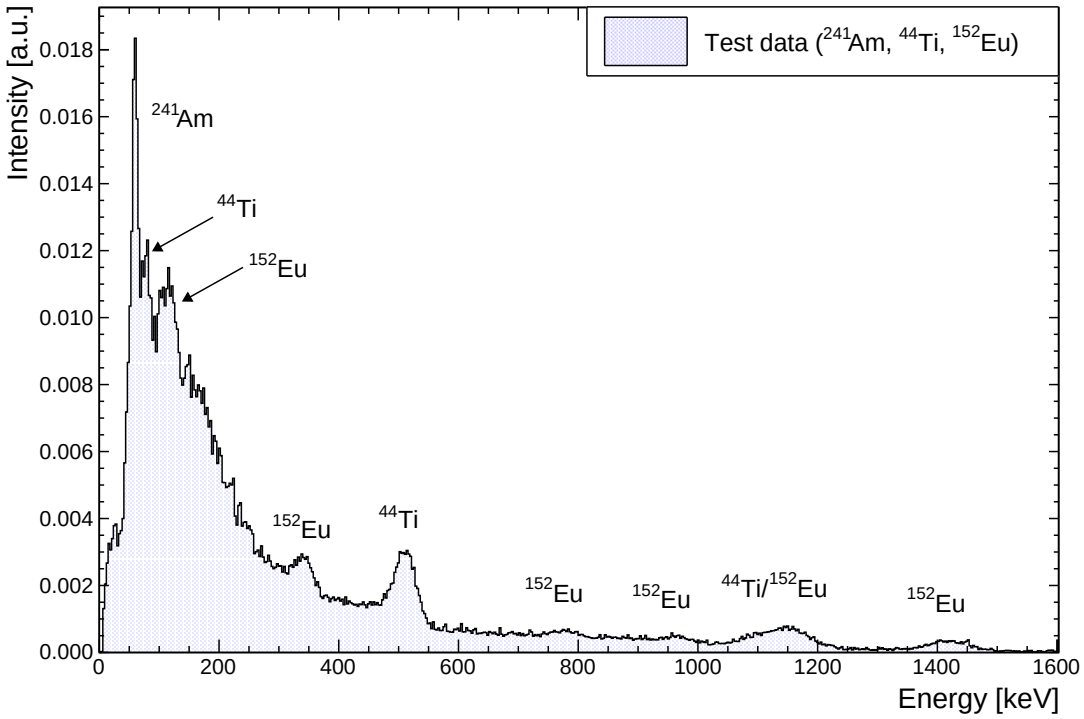



Figure 5.9: Example test spectrum containing $1\times10^6$ total counts from equal activity $^{241}$Am, $^{152}$Eu, and $^{44}$Ti. There is 6 cm of aluminium shielding between the sources and the detector. Note the loss of low energy information due to attenuation and a large backscatter feature.

A test case of heavily shielded conditions were simulated to reproduce the experimental test data. The sources were placed 10 cm from the 3" NaI detector, with varying amounts

of shielding material positioned between the two. Up to three plates of $10 \times 10 \times 2$ cm$^3$ aluminium provided a maximum of 6 cm of shielding that caused significant disruptions to the spectra. It was ensured that the detector was completely obscured by the shielding, so that no source radiation could enter the crystal unimpeded. Figure 5.9 shows the large disruptions that make these conditions are such a challenge.

True coincidence summations are a very minor concern here because the detector cannot physically get close enough to cause significant features due to the shielding. The same procedure of averaging four simulated training sets, each evaluated against six experimental test sets, is followed for consistency. All data sets have the same large gain shifts and other parameters specified in Table 5.1. The result of training the reference model on these data is summarised in Table 5.8.

Table 5.8: All performance metrics for a range of shielding conditions. Data were collected using a 3" NaI detector at 10 cm, with shielding placed between the source and detector. Each aluminium plate is 2 cm thick.

| Performance Metric | 1 Al plate (%) | 2 Al plates (%) | 3 Al plates (%) |
|---|---|---|---|
| Precision | 99.4(1) | 98.6(1) | 95.0(2) |
| Perfect rate | **73.9(9)** | **59.9(7)** | **34.9(8)** |
| TPR | 96.7(2) | 94.2(2) | 87.7(3) |
| FPR | 3.8(1) | 6.5(3) | 12.9(3) |
| TNR | 96.2(1) | 93.5(2) | 87.1(3) |
| FNR | 3.3(2) | 5.8(2) | 12.3(3) |

By all accounts, the model is performing well overall, with good precision and classification rates for individual labels. The difference in perfect rates best highlights how performance suffers as more shielding is used. To be explicit, the probability of correctly predicting the presence of each of the nine possible sources for any one spectrum decreases as more shielding is added. Of course, this is almost a foregone conclusion. The interesting details are in why this happens in terms of the model.

The ROC curve of a 6 cm thick shielding data set is shown in Figure 5.10. Unlike those of the point source data (see Figure 5.7), these rates indicate $^{241}$Am, $^{152}$Eu, and $^{44}$Ti to be the cause of more incorrect predictions. The $^{44}$Ti case has always been difficult to distinguish from $^{207}$Bi, but performs similarly to previous results since there are large features at energies beyond the main effects of shielding. Similarly, $^{60}$Co and $^{22}$Na are no longer the cause of the most incorrect predictions because they too have all of their most prominent features at the higher energies.

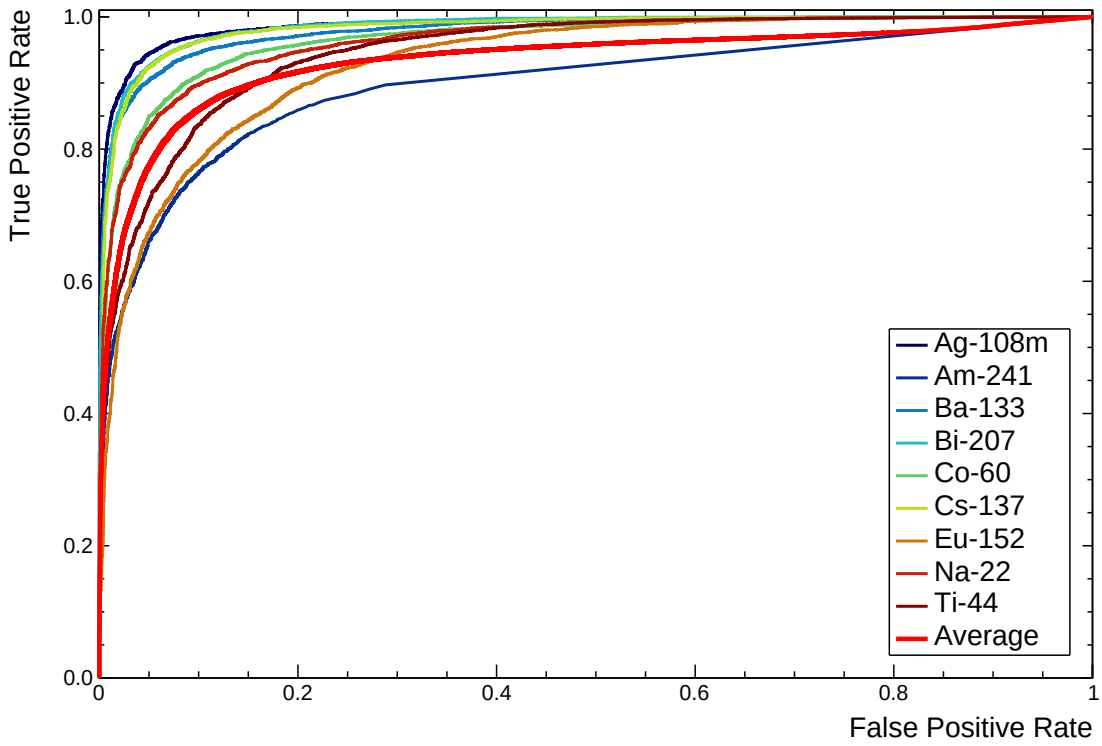



Figure 5.10: Receiver Operating Characteristic (ROC) for a heavily shielded test data set, showing $^{241}$Am, $^{152}$Eu, and $^{44}$Ti to be responsible for more incorrect predictions. Americuim-241 in particular contributes to the majority of false negative classifications.

Below ∼300 keV the effects of shielding are clear from Figure 5.9. Photopeaks sit atop a large backscatter feature, and contain significantly fewer counts than would be expected form an unattenuated source. The ROC curves reveal $^{241}$Am to be the most difficult. The only real feature, a 59.5 keV peak, is expected to be 55%, 31%, and 17% of the equivalent unshielded intensity for the three plate thicknesses respectively [94]. This, coupled with a low relative activity in multi-isotope spectra, mean that it can often leave very little to indicate its presence. Indeed, checking the confusion matrix for the source shows a heavy skew toward false negative results.

In a similar vein, the most prominent features of an $^{152}$Eu spectrum are also at low energies and quickly become heavily attenuated. However, unlike $^{241}$Am, this source does present features at higher energies that could be used instead by a spectroscopist. It may be inferred that the unshielded data assign more weight to these larger, low-energy features that are no longer available in the shielded data.

Overall these results are very promising, and it is encouraging that model performance reflects the challenges of the physical phenomena involved. This allows for a more intuitive

understanding of model limitations. A 2 cm thick aluminium shield performs almost as well as the unshielded conditions. Classification rates reduce as more shielding is added, dropping to a perfect rate of 35% for 6 cm. The precision shows that of those source predictions that were correct, they were made with confidence. It is worth noting that while perfect rates drop below 50%, this is certainly better than just randomly assigning a label. In this multi-label case the perfect rate for doing so would be 0.2%.

### 5.4.2 Close geometries

True coincidence summations become more likely in close geometries. The solid angle of a source at 1 cm means that almost all spectra contain additional photopeaks. This can drastically increase the complexity of multi-isotope spectra. This is an important challenge because having the detector as close as possible to the source reduces collection times. For low activity samples this can be a crucial factor for pragmatic RIID systems.

The reference model was able to perform relatively well even with the additional complexity. This is, in part, due to accurate handling of coincidence summations in the simulations, which produce good training data as a result.

Table 5.9: All performance metrics for a range of stand-off conditions.

| Performance Metric | 1 cm (%) | 5 cm (%) | 10 cm (%) |
|---|---|---|---|
| Precision | 99.24(3) | 99.51(3) | 99.61(3) |
| Perfect rate | **66.5(5)** | **73.9(6)** | **74.4(9)** |
| TPR | 94.5(1) | 95.4(1) | 96.5(1) |
| FPR | 3.4(1) | 2.0(1) | 3.1(2) |
| TNR | 96.6(1) | 98.0(1) | 96.9(2) |
| FNR | 5.5(1) | 4.6(1) | 3.5(1) |

The results of varying the stand-off distance are shown in Table 5.9. Since the probability of summation quickly reduces with an inverse square law, it is unsurprising that the 5 cm and 10 cm stand-off distances performed similarly. Investigating the ROC curves reveals the same sources ($^{44}$Ti, $^{22}$Na, and $^{60}$Co) to cause more false classifications as with the 10 cm reference case.

The false negative rates are the most concerning for security. While similar, these are larger than the false positive rates. This would ideally be reversed as seen in the shielding results. This highlights another consideration, which is the threshold used to make a prediction.

The ROC curves show the result at every decision threshold, so it is important to understand that this threshold can play an important role in RIID. Most models demonstrate good

stability in this regard and have little variation in prediction rates around the 0.5 training threshold. In these evaluations, the threshold was taken that maximises perfect prediction rates. However, with the application in mind, it may prove appropriate to sacrifice performance on individual spectra if the FNR can be reduced. Better yet, the threshold can be tuned to reduce false negatives for certain sources of particular interest. There are, therefore, several ways to optimise the predictive behaviour of these models that will need to be considered for the application at hand.

There is a much greater degree of difficulty for these sets when compared to an unshielded point source at a greater stand-off distance. While performance may suffer as a result of using a model not optimised for them specifically, it shows that even a basic model structure is generalised enough to work well for a diverse range of challenging conditions.

To this point, the focus has been on the model itself. In machine learning applications there are many other things that can be done to improve performance. While emphasis has been placed on the need to understand the data sets used, they have so far been neglected in these investigations. The following section details how important these data sets can be to a model's ability to generalise.

## 5.5 Data sets

The 74.4(9)% perfect rate for the 10 cm stand-off NaI spectra containing a mixture of up to nine different sources is already an excellent result. This is also true of the model performance under close geometries and heavily shielded conditions. However, as previously mentioned, the data sets themselves are incredibly important for machine learning. Revisiting the data sets used to train the models can provide additional gains in performance. Since the 10 cm NaI data are well understood for the reference model, they will be carried through for comparison. The common theme here is representation. Simply put, the ideal learning conditions for any deep learning application are those that provide a comprehensive set that contains as many examples as possible. With more examples included for training, the more representative of real conditions the set becomes.

### 5.5.1 Data set size

One of the simplest things to change is the size of the training data set. For rapid prototyping, a very small set averaging only ∼10 samples per unique source combination was previously used. With the extremes in gain shifts, counts, and relative activities of the mixed sources, this makes for a set that can not possibly represent the diverse range of spectra that may be present in experimental test sets.

The parameters of Table 5.1 were used to generate new simulated training sets. The same random seed was kept consistent between sets, with the only variable being the number of samples for each of the 511 unique source combinations. Three repeat models were trained for each set size, testing against the six original test sets.

Figure 5.11 shows how there is a sharp performance drop when using too few training samples. While there is an improvement seen as more spectra are added, the performance plateaus for these particular data. For more complex data sets the additional representation is expected to become far more important.

By doing nothing more than adding more training data, the plateau seen reaches ∼81% for the perfect prediction rate. This is one reason why it is often worth augmenting particularly small training sets with simulated data. However, doing so is limited because here the training data are entirely simulated, perhaps missing the subtleties of real spectra. This is why it is always better to include as many real samples as possible, whatever the application.

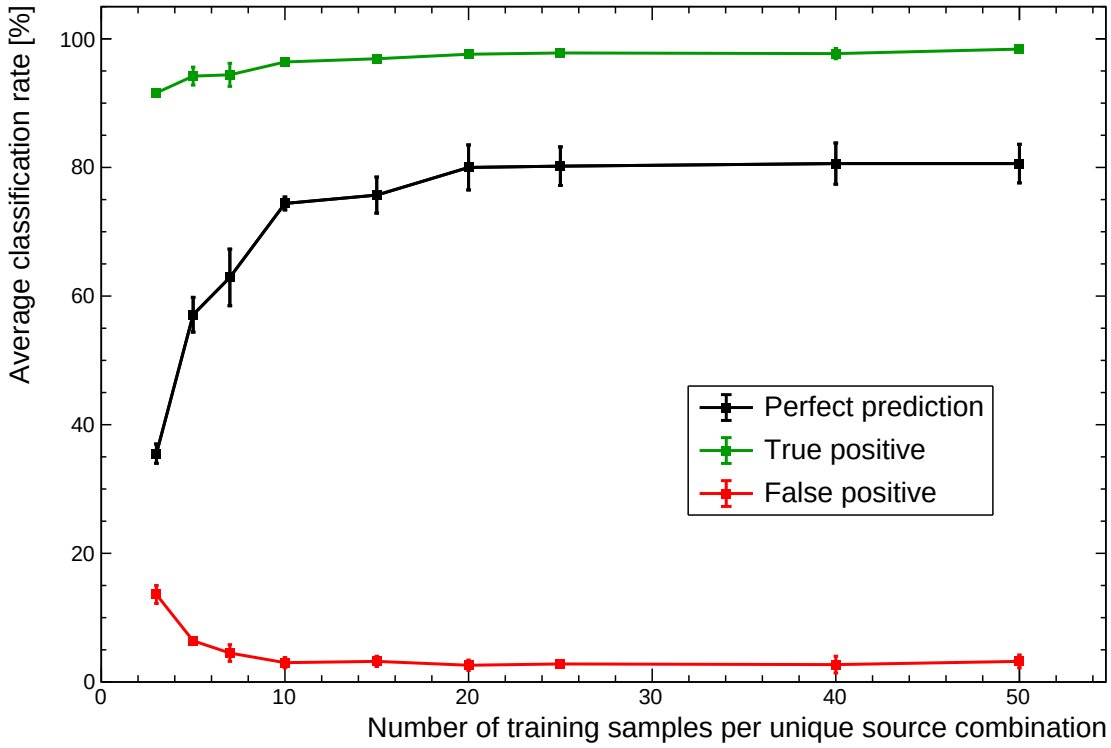



Figure 5.11: Example of the variation in classification rates as the number of training samples per unique source combination is varied.

### 5.5.2 Including real data

These investigations started by including yet another degree of difficulty to address a very specific problem by using *only* simulated spectra for training, while testing on experimental data. Unfortunately, simulated data are only ever an approximation of reality no matter how good they may be. Taking the generalised approach in this work has been fantastic for providing a rapid development environment, but excludes the subtleties of backscatter from materials surrounding the detector that are present in real spectra. In an ideal world there would be plenty of these experimental data, but a compromise often made is to work with what is available while supplementing with simulation.

To make new training sets that contain an even mix of real and simulated data, four new half-sets of training data were generated from experimental templates using unique random seeds, with a total of 10 samples per unique source combination. This ensured that there would be no bias toward any particular test set. Except for the size, these data were generated using the same parameters specified in Table 5.1. These sets were each combined

using the methods described in Section 4.2.2 with one of four simulated half-sets, also 10 samples per source combination, to make a full set of training data.

With file and label lists combined, a random 50/50 split of training and validation data were taken for consistency. Training on the reference model, the perfect prediction rate averaged 89.8(9)% over the 16 possible combinations of the half-sets. The ROC curves show that the $^{44}$Ti, $^{22}$Na, and $^{60}$Co sources were, again, the source of more incorrect predictions.

Such a good prediction rate for individual spectra provides a great example of desirable performance stability as the decision threshold varies. As Figure 5.12 shows, there is a large range around the 0.5 training threshold where the overall performance is relatively unaffected. This allows the model to be tuned to the needs of the system. For example, security applications need fewer false negatives, so the threshold can be reduced. This will increase the false positive rate, but to an extent will leave overall performance for individual spectra relatively unaffected.

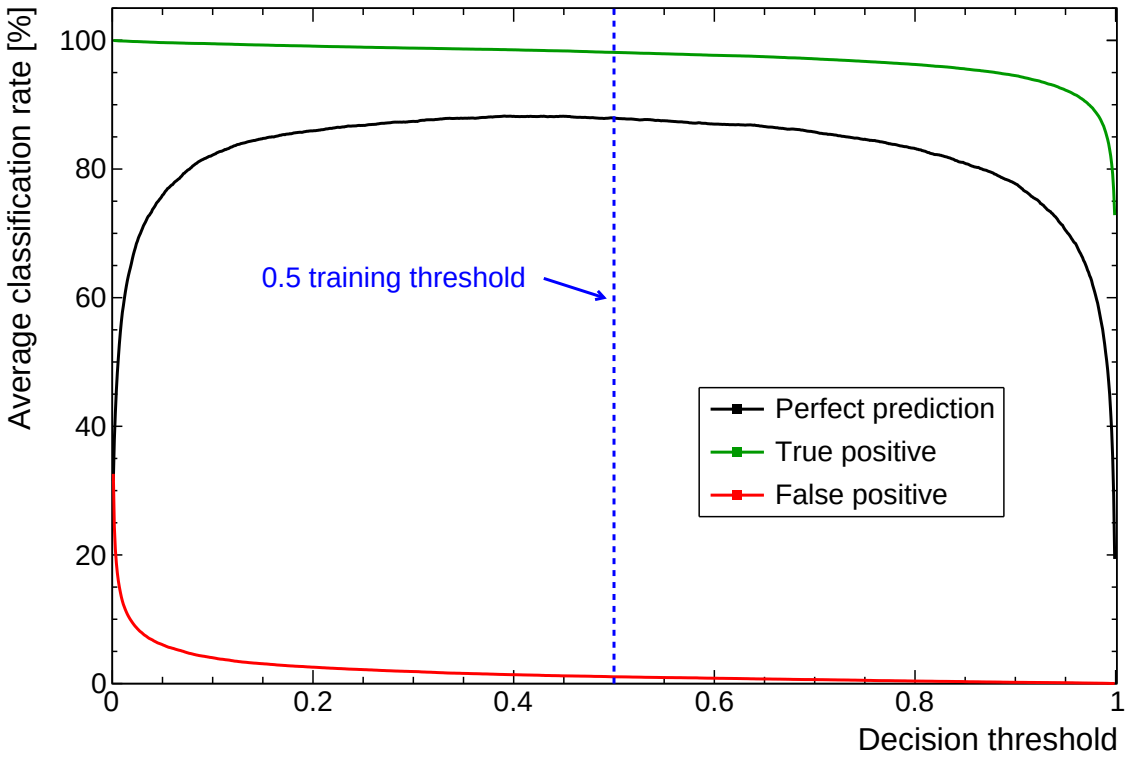



Figure 5.12: Example of the variation in classification rates as the decision threshold is varied.

Including as many real data as possible is highly recommended, whatever the application, and these results clearly demonstrate the advantages of doing so. As good as the simulations are, the subtleties of backscatter from the surrounding environment and other effects left

unsimulated have clear impact on a model's ability to learn. As suggested by the shielded investigations, the low energy region may have more importance, especially as this is where some of the most prominent features can be. Having a better representation of the distortions in this region was certainly proven desirable for these sets.

### 5.5.3 Generalising the model

It has already been shown that these models perform relatively well at different stand-off distances and levels of shielding. In every case, a model is trained on a specific simulated data set, and tested on the experimental version. Section 4.2.2 detailed how many data sets can be combined to make a more generalised model.

A large data set was made by combining every individual data set, each with 20 training samples per source combination. The reference model was then trained on this all inclusive set. The single, generalised model made predictions on all of the test sets separately for comparison with the performance of previous sections. The results are summarised in Table 5.10.

Table 5.10: Comparison of performance on test data sets. The models trained individually are compared against the generalised model.

| Test set | Individual (Perfect rate %) | Generalised (Perfect rate %) |
|---|---|---|
| 1 cm stand-off | 66.5(5) | 72.6(9) |
| 5 cm stand-off | 73.9(6) | 90.5(4) |
| 10 cm stand-off | 74.4(9) | 90.9(4) |
| 1 Al plate | 73.9(9) | 88.5(7) |
| 2 Al plates | 59.9(7) | 81.5(6) |
| 3 Al plates | 34.9(8) | 60.4(7) |

The results are surprisingly good, rivalling the addition of real data for the 10 cm stand-off data. This follows expectations, given that the training data are a 6× larger, and far more comprehensive. Challenging conditions still make heavy shielding and close geometries more difficult, mirroring the trends seen with the more focused models trained in individual data sets.

# Chapter 6

# Conclusions

The scope of this research covers many goals and spans a range of disciplines. The original proposal has the open-ended aim of developing techniques for both forward and reverse problems in nuclide identification.

As Section 2.2.1 discusses, there are many challenges that Radio-Isotope IDentification (RIID) algorithms must face. A robust, affordable, and pragmatic system is one that can handle adverse conditions and any transient effects that change the profile of ideal spectra. Shielding and gain shifts are pertinent examples. This is where many current solutions fail, and why research aims quickly moved toward applying machine learning methods to the RIID problem space.

The relatively unexplored deep learning approach used here requires large datasets filled with example spectra to learn from. The use of simulated spectra is well matched to such techniques, with a good simulator rapidly generating spectra that bypasses the need for experimental data in prototyping phases. To this end, a simulator was built that could produce varied and realistic spectra for a rapid development cycle. The development, testing, and integration of various tools into this simulator enabled high-fidelity modelling of radioactive sources, shielding, and detection systems.

Chapter 3 discusses the details of the simulator. All important features are accurately reproduced in simulated spectra, including Gaussian Energy Broadening (GEB) and multi-isotope sources. This extended to even the more subtle effects, such as X-ray fluorescence, true coincidence summations, and extra broadening of annihilation peaks.

While there is a tendency to ignore the result of true coincidence summations, it was decided that the effect of close source-detector geometries was important for practical RIID. Significant work on the handling of timing data made this information readily available, while working within the GEANT4 framework. This naturally led to a simulator with extended functionality towards more in-depth investigations, such as coincidence and angular

correlation measurements with multiple detectors in an array. For specific cases, this is not too difficult to achieve, but the generalised techniques implemented here allow for far more versatility with minimal user input.

As noted in the summary of the simulator (see Section 3.3), no simulator is ever perfect, especially those intended to be a generalised solution. The angular correlations showcased demonstrate there to be no general implementation in the current GEANT source. Fortunately this is a negligible effect for the purposes of this research, but a solution would need to be manually coded for applications requiring it. The greatest source of discrepancy in the overall profile of a spectrum is always in the low (<100 keV) energy region. The contributions of Naturally Occurring Radioactive Material (NORM), extraneous background sources, and backscatter are all so specific to the surrounding environment that a generalised profile becomes impossible to define.

Even with these current limitations the simulator was certainly able to closely reproduce even HPGe spectra with minimal effort. For the lower resolution detectors, much of the backscatter and NORM effects are somewhat mitigated due to most of the subtle features being smeared out. Chapter 4 detailed how these spectra, along with their experimentally obtained counterparts, were prepared for the machine learning models to train on.

There are many effects that will change the features seen in a spectrum and its general profile (see Section 2.1). The ambitious challenge of including variations in statistics, gain shifts, multiple sources with different relative activities, background contributions, and low-energy discrimination makes it an impossibly slow task to collect or simulate so many possible combinations. It is well known that a key part of deep learning lies with providing a representative set of training data, so this was a problem.

The template method recently established by Kamuda *et. al.* [16] was adopted to keep simulations and experimental collection times to a minimum. Examples of idealised spectra serve as templates that can be combined and manipulated in the pre-processing of training and test sets. For example, there is no need to change the ADC threshold in a series of experimental data if it is possible to simple reset the first few bins to have zero counts in the histogram. This is the most basic example, but certainly illustrates the simple principle.

By normalising the spectra, they become a probability distribution from which to randomly sample. This allowed counts from many templates to be combined, in varying amounts, to build up multi-isotope spectra and emulate the statistics of low count rates. Gain shifts were also included in this preprocessing stage, though proved to be more complex than simply changing the energy calibration. The machine learning models take only the count data across a fixed range of channels, with no information on the energy assigned to each bin.

While it is an additional degree of difficulty to make predictions on uncalibrated data, this meant that the intensities had to shift bins. This was achieved by applying the shift before interpolating the result to conform to the original channel structure.

The library of samples generated from templates, all containing random perturbations of the aforementioned effects, was labelled with the target outcomes for a supervised learning problem. Storing spectra individually with the target outcomes in the file name meant that all information could be inferred from lists of the file paths. This allowed for a modular and scaleable approach to handling datasets, with any combination of datasets easily provided to the models for training and testing. This format also allowed training data to be loaded in batches to minimise memory usage and accommodate large datasets.

The 30 years of deep learning for RIID discussed in Section 2.2.4 showed the Artificial Neural Network (ANN) to be the exclusive choice for deep learning techniques. These models are able to handle unexpected and changing conditions because they learn generalised rules from experience, rather than being explicitly programmed to find certain identifying features. However, an ANN rapidly grows in complexity as more layers and nodes are added because it is typically fully-connected. Chapter 5 demonstrates how a Convolutional Neural Network (CNN) may be applied to RIID with excellent results, often with less computational expense.

The CNN variant of neural networks works by training sets of convolution filters that perform feature extraction for an input spectrum, before using a small ANN for the final classification. Responsibility is abstracted away from the user and placed with the model so that there can be minimal pre-processing. In this case, spectra were given to the model uncalibrated, simply passed as an array of normalised intensities. No peak finding, smoothing, or transforms required.

A reference model was created that used three convolution modules (Fig. 5.1) followed by a classification module (Fig. 5.2). Even this basic sequential structure has many hyper-parameters to investigate, the permutations of which can create many drastically different models. It was found that the dropout rates and number of nodes in the classifying ANN were perhaps the most important, with compromises to be made between computational expense and concerns of overfitting. Testing on the 10 cm stand-off data for NaI spectra, a perfect prediction rate of 74.4(9)% showed excellent predictive capabilities for individual spectra. For perspective, a model trained on only simulated data was very often able to correctly predict the presence of all nine possible sources in experimentally obtained, multi-isotope spectra with extreme gain shifts, low energy cuts, and $10^3 - 10^6$ counts. Overall true positive and true negative rates of 96.5(1)% and 96.9(2)% respectively show that few of the labels were incorrect in the remaining 25% of test spectra with a set of imperfect predictions. A

99.61(3)% average precision also indicates that all correct predictions were made with a high degree of confidence.

Of course, the other major challenges of RIID beyond gain shifts and multi-isotope spectra are when there are true coincidence summations or heavy shielding. Training and test sets at near stand-off distances and with aluminium shielding were used to test the reference model.

For shielding, there was the expected drop in performance as more shielding was added. Predictions on data with up to 4 cm of aluminium performed surprisingly well, only beginning to have significant difficulty when 6 cm started to attenuate low energy features that are relied upon for predictions, such as $^{241}$Am. This increased false negative rates to an uncomfortable 12.3(3)%, which is certainly an undesirable trait for many RIID applications.

The presence of true coincidence summation peaks also showed a small reduction in performance as the sources came closer to the detector. However, even with the added complexity of these additional features, the perfect rate of 66.5(5)% for 1 cm is not dissimilar to the more ideal 10 cm case. Intuitively it is perhaps better to have these extra features to aid in the identification, but the limitations of NaI detectors make resolving peaks difficult, with many source emissions already overlapping to form distorted features. This is only exasperated by the addition of summation peaks.

The datasets used to train models can influence the performance greatly. The same reference model was trained while datasets varied. By simply including more simulated spectra in the training sets, performance improved by $\sim$5%. The extra examples of different spectra helped improve performance, but another approach is to make the training data more representative by using experimental data directly. A new set of the same 20 sample per source size was generated that contained 50% experimental data. Performance increased dramatically, from 74.4(9)% to 89.8(9)%. While it is good to be able to train on purely simulated data, mixing in as many available real spectra as possible is the obvious choice.

By combining all six sets of various stand-off distances and shielding conditions, a large and more generalised model may be trained. Using the same generalised model to make predictions on individual test sets showed performance that rivalled the inclusion of real data. Again, this is due to having far more examples to train on, including spectra that are distorted in ways that may not even be present in every test set.

This research spent a lot of time building the simulator, writing supporting code around the models, and developing techniques specific to this application. However, with this in place and trends known, any model optimisation can simply be automated using sensible limits. Having a reference model and datasets to act as a baseline also encourages a lot of future work to continue and expand the project in new directions.

# Chapter 7

# Future work

The simulation and machine learning environments developed over the course of this research provide a strong foundation on which to build. Of course, no simulator is ever perfect, so one option is to further expand its capabilities for new RIID conditions. However, it is the author's opinion that the immediate work should not be to eliminate the current limitations or expand functionality. Instead, the current framework should be used to further investigate machine learning techniques. The methodology described in this thesis took time to develop, and while development of RIID techniques is now rapid, many avenues remain unexplored.

The first and most obvious objective should be to collect template data for other detectors, such as NaI detectors of varying size and configuration, HPGe with its superb resolution, or $LaBr_3$ scintillators for their excellent efficiency. With the current tools in place, it would simply be a matter of defining suitable parameters with which to generate new datasets, using them to retrain current models. This is all done through the common preprocessing steps and interfaces described in Chapter 4. Additional coding is, therefore, kept to a minimum.

On the subject of new models, this work provides the opportunity to investigate many more advanced models and techniques. As mentioned, an inception model [86] is just one of many convolution-based models that have shown promise in exploratory testing. The VGG-16 [95] inspired sequential CNN presented is meant only to establish a baseline against which others may be tested.

There are also options for more varied and advanced techniques. For example, 'transfer learning' takes pre-trained models as a starting point. For example, there are models already trained for identifying certain animals in images, but many of the important features will apply to other animals. For applications with new animals it can be beneficial to use the

filters and weights already trained on a much larger, more comprehensive set. This is how commercial systems could work, shipping with a model trained on a large central dataset that can either be expanded upon or used as a starting point for more specialised RIID conditions. The fundamentals of photopeaks, Compton continua, and Gaussian Energy Broadening will always be features of gamma spectra.

In a similar vein it is possible to freeze the training of layers so that only certain parts of the model continue training. One could, for example, fix the early convolution layers that often extract the same low level features, and train only the later stages that look at more abstract, higher level features. In this case the expectation is that filters are already trained for the identification of edges, peaks, and shapes that apply to any spectrum. Later layers could then tailor the model to recognise features specific to target sources, such as the number of expected peaks and the distance between them. While the early stages were confirmed to perform this way by visualising filters, this was not explored in enough depth to investigate the later layers. The visualisation of filters and their effects is not overly complex, but remains another opportunity for future work.

By defining the RIID aims to be that of a multi-class, multi-label problem, the presence of all nine test sources could be predicted with a single model. However, one of the major problems with this is scalability. While this may be fine for up to the ANSI list of $\sim$20 target nuclides [34], the computational expense is still a factor due to the fully-connected layers of the small ANN. Since the labels are treated independently anyway, it will perhaps be better to move toward 'ensemble'-style networks to carry forward the modular ideas suggested in [22].

Ensemble methods combine multiple smaller models into a single predictive model to obtain better classification performance. The commonly used random forest [96] model is a good example, splitting the data across many decision trees and aggregating results to make predictions. Smaller and more modular CNNs could therefore be focused solely on the binary classification of single sources. This would make it easier to provide pre-trained models for any combination of applicable sources. Of course, this may perform worse than a single multi-label model, but grants an interesting opportunity for further work nonetheless.

Machine learning is a rapidly evolving field with a constantly shifting landscape of ideas and best practices. The accessibility of open-access software such as TensorFlow provides researchers with the tools to explore learning techniques in their own areas, as this work demonstrates. It is the author's hope that this work be developed further by future projects, using many exciting new model types and techniques to come.

# Glossary

**ANN** Artificial Neural Network. 3, 115

**CNN** Convolutional Neural Network. i, 3, 115

**GEANT** GEometry ANd Tracking. 3, 42, 43, 45

**HPGe** High Purity Germanium. 1, 7, 46

**LANL** Los Alamos National Laboratory. 42

**MCNP** Monte Carlo N-Particle. 42

**NaI** Sodium Iodide. 1, 7, 46

**PMT** Photomultiplier Tube. i, 11, 12, 46

**RIID** Radio-Isotope IDentification. i, 1, 2, 17, 113

**ROC** Receiver Operating Characteristic. vii, 101, 106

# Bibliography


1. Turner, A., Wheldon, C., Gilbert, M., Packer, L., Burns, J., Kokalova, T. & Freer, M. Generalised gamma spectrometry simulator for problems in nuclide identification. *J. Phys.: Conf. Ser.* **1643.** 27th International Nuclear Physics Conference (INPC2019). doi:10.1088/1742-6596/1643/1/012211 (2019) (cit. on p. i).
2. Turner, A., Wheldon, C., Kokalova, T., Gilbert, M., Packer, L., Burns, J. & Freer, M. Convolutional neural networks for challenges in nuclide identification (2020, submitted) (cit. on p. i).
3. Stromswold, D. *et al.* Field tests of a NaI(Tl)-based vehicle portal monitor at border crossings. *Nuclear Science Symposium Conference Record* **IEEE.** doi:10.1109/NSSMIC.2004.1462180 (2004) (cit. on pp. 2, 17).
4. Blackadar, J. Automatic isotope identifiers and their features. *IEEE Sensors Journal* **5,** 589–592. doi:10.1109/JSEN.2005.846374 (2005) (cit. on p. 2).
5. Sullivan, C. *et al.* Evaluation of key detector parameters for isotope identification. *IEEE Nuclear Science Symposium Conference Record* **2,** 1181–1184. doi:10.1109/NSSMIC.2007.4437217 (2007) (cit. on pp. 2, 17).
6. Burr, T. & Hamada, M. Radio-Isotope Identification Algorithms for NaI $\gamma$ Spectra. *Algorithms* **2,** 339–360. doi:10.3390/a2010339 (2009) (cit. on pp. 2, 20).
7. Burr, T. *et al.* Augmenting real data with synthetic data: an application in assessing Radio-Isotope identification algorithms. *Quality and Reliability Engineering International* **25,** 899–911. doi:10.1002/qre.1003 (2009) (cit. on pp. 2, 17).
8. Stinnett, J. Automated isotope identification algorithms for low-resolution gamma spectrometers. *PhD Thesis, University of Illinois at Urbana-Champaign* (2017) (cit. on pp. 2, 17).

9. Swoboda, M. *et al.* Spectral gamma detectors for hand-held radioisotope identification devices (RIDs) for nuclear security applications. *IEEE Transactions on Nuclear Science* **52,** 3111–3118. doi:10.1109/NSSMIC.2004.1466839 (2005) (cit. on p. 2).

10. Caffrey, A. *et al.* OSIRIS: Gamma-ray spectroscopy software for on-site inspections under the Comprehensive Nuclear-Test-Ban Treaty. *Nuclear Instruments and Methods in Physics Research Section A* **784,** 405–411. doi:10.1016/j.nima.2014.10.066 (2015) (cit. on p. 2).

11. Jarman, K. *et al.* A comparison of simple algorithms for gamma-ray spectrometers in radioactive source search applications. *Applied Radiation and Isotopes* **66,** 362–371. doi:10.1016/j.apradiso.2007.09.010 (2008) (cit. on p. 2).

12. Mariscotti, M. A method for automatic identification of peaks in the presence of background and its application to spectrum analysis. *Nuclear Instruments and Methods* **50,** 309–320. doi:10.1016/0029-554X(67)90058-4 (1967) (cit. on pp. 2, 21).

13. Hofstadter, R. Alkali Halide Scintillation Counters. *Phys. Rev.* **74,** 100–101. doi:10.1103/PhysRev.74.100 (1 1948) (cit. on pp. 2, 11).

14. Mehta, P. *et al.* A high-bias, low-variance introduction to Machine Learning for physicists. *Physics Reports* **810,** 1–124. doi:10.1016/j.physrep.2019.03.001 (2019) (cit. on pp. 2, 28, 85).

15. Agostinelli, S. Geant4 - a simulation toolkit. *Nuclear Instruments and Methods in Physics Research Section A* **506,** 250–303. doi:10.1016/S0168-9002(03)01368-8 (2003) (cit. on pp. 3, 42).

16. Kamuda, M., Stinnett, J. & Sullivan, C. Automated Isotope Identification Algorithm Using Artificial Neural Networks. *IEEE Transactions on Nuclear Science* **64,** 1858–1864. doi:10.1109/TNS.2017.2693152 (2017) (cit. on pp. 3, 24, 26, 27, 33, 41, 75, 77, 82, 114).

17. Kim, J., Park, K. & Cho, G. Multi-radioisotope identification algorithm using an artificial neural network for plastic gamma spectra. *Applied Radiation and Isotopes* **147,** 83–90. doi:10.1016/j.apradiso.2019.01.005 (2019) (cit. on pp. 3, 24, 25).

18. Medhat, M. Artificial intelligence methods applied for quantitative analysis of natural radioactive sources. *Annals of Nuclear Energy* **45,** 73–79. doi:10.1016/j.anucene.2012.02.013 (2012) (cit. on pp. 3, 23, 29).

19. Olmos, P. *et al.* A new approach to automatic radiation spectrum analysis. *IEEE Transactions on Nuclear Science* **38,** 971–975. doi:10.1109/23.83860 (1991) (cit. on pp. 3, 21, 24, 27).

20. Kamuda, M. & Sullivan, C. An automated isotope identification and quantification algorithm for isotope mixtures in low-resolution gamma-ray spectra. *Radiation Physics and Chemistry* **155,** 281–286. doi:10.1016/j.radphyschem.2018.06.017 (2019) (cit. on pp. 3, 24, 25, 27).

21. Liang, D. *et al.* Rapid nuclide identification algorithm based on convolutional neural network. *Seminars in nuclear medicine* **133,** 483–490. doi:10.1016/j.anucene.2019.05.051 (2019) (cit. on pp. 3, 21, 24, 26).

22. Daniel, G. *et al.* Automatic and Real-Time Identification of Radionuclides in Gamma-Ray Spectra: A New Method Based on Convolutional Neural Network Trained With Synthetic Data Set. *IEEE Transactions on Nuclear Science* **67,** 644–653. doi:10.1167/17.9.10 (2020) (cit. on pp. 3, 26, 27, 118).

23. Browne, E. & Tuli, J. Nuclear Data Sheets for A = 60. *Nuclear Data Sheets* **114,** 1849–2022. doi:10.1016/j.nds.2013.11.002 (2013) (cit. on p. 4).

24. Browne, E. & Tuli, J. Nuclear Data Sheets for A = 137. *Nuclear Data Sheets* **108,** 2173–2318. doi:10.1016/j.nds.2007.09.002 (2007) (cit. on pp. 5, 16).

25. Knoll, G. *Radiation Detection and Measurement, 4th Edition* ISBN: 978-0470131480 (John Wiley and Sons, University of Michigan, 2010) (cit. on pp. 5, 6, 8, 10, 12, 13, 15, 50, 63).

26. Leroy, C. & Rancoita, P. *Principles of Radiation Interaction in Matter and Detection, 2nd Edition* ISBN: 978-981-281-829-4 (World Scientific, Université de Montréal, Canada, 2009) (cit. on pp. 6, 12).

27. Gilmore, G. *Practical Gamma-ray Spectroscopy, 2nd Edition* ISBN: 9781119964698 (John Wiley and Sons, 2011) (cit. on pp. 10, 13, 50, 51, 57).

28. Iltis, A. *et al.* Lanthanum Halide Scintillators: Properties and Applications. *Nuclear Instruments and Methods in Physics Research A* **563,** 359–363. doi:10.1016/j.nima.2006.02.192 (2006) (cit. on p. 11).

29. Van Loef, E. *et al.* Scintillation Properties of $LaCl_3$:$Ce^{3+}$ Crystals: Fast, Efficient, and High-Energy Resolution Scintillators. *IEEE Transactions on Nuclear Science* **48,** 341–345. doi:10.1109/23.940077 (2001) (cit. on p. 11).

30. Higgins, W. *et al.* Crystal growth of large diameter $LaBr_3$:Ce and $CeBr_3$. *Journal of Crystal Growth* **310,** 2085–2089. doi:10.1016/j.jcrysgro.2007.12.041 (2008) (cit. on p. 11).

31. Ter Weele, D., Schaart, D. & Dorenbos, P. The Effect of Self-Absorption on the Scintillation Properties of $Ce^{3+}$ Activated $LaBr_3$ and $CeBr_3$. *IEEE Transactions on Nuclear Science* **61,** 683–688. doi:10.1109/TNS.2013.2296101 (2014) (cit. on p. 11).

32. Van Loef, E. *et al.* Scintillation Properties of $LaBr_3$:$Ce^{3+}$ Crystals: Fast, Efficient and High-energy-resolution Scintillators. *Nuclear Instruments and Methods in Physics Research A* **486,** 254–258. doi:10.1016/S0168-9002(02)00712-X (2002) (cit. on p. 11).

33. H.G.J. Moseley. High-frequency spectra of the elements. *The London, Edinburgh and Dublin philosophical magazine and journal of science* **6,** 1024–1034 (1840) (cit. on p. 14).

34. Institute, A. N. S. N42.34-2015 - American National Standard Performance Criteria for Handheld Instruments for the Detection and Identification of Radionuclides. *IEEE Standards Association.* doi:10.1109/IEEESTD.2016.7551091 (2016) (cit. on pp. 17, 26, 118).

35. Moss, C. E. *et al.* Comparison of active interrogation techniques. **1,** 329–332. doi:10.1109/NSSMIC.2005.1596264 (2005) (cit. on p. 19).

36. Olmos, P. *et al.* Application of neural network techniques in gamma spectroscopy. *Nuclear Instruments and Methods in Physics Research Section A* **312,** 167–173. doi:10.1016/0168-9002(92)90148-W (1992) (cit. on pp. 22, 24, 25).

37. Olmos, P. *et al.* Drift problems in the automatic analysis of gamma-ray spectra using associative memory algorithms. *IEEE Transactions on Nuclear Science* **41,** 637–641. doi:10.1109/23.299814 (1994) (cit. on pp. 22, 24).

38. Keller, P., Kouzes, R. & Kangas, L. Applications of Neural Networks to Real-Time Data Processing at the Environmental and Molecular Sciences Laboratory. *Conference Record of the Eighth Conference on Real-Time Computer Applications in Nuclear* **(PNL-SA–22026)** (1993) (cit. on p. 22).

39. Keller, P. *et al.* Nuclear spectral analysis via artificial neural networks for waste handling. *IEEE Transactions on Nuclear Science* **42,** 709–715. doi:10.1109/23.467888 (1995) (cit. on p. 22).

40. Vigneron, V. *et al.* Statistical modelling of neural networks in $\gamma$-spectrometry. *Nuclear Instruments and Methods in Physics Research Section A* **369,** 642–647. doi:10.1016/S0168-9002(96)80068-4 (1996) (cit. on pp. 22, 23, 27).

41. Abdel-Aal, R. & Al-Haddad, M. Determination of radioisotopes in gamma-ray spectroscopy using abductive machine learning. *Nuclear Instruments and Methods in Physics Research Section A* **391,** 275–288. doi:10.1016/S0168-9002(97)00391-4 (1997) (cit. on p. 22).

42. Pilato, V. *et al.* Application of neural networks to quantitative spectrometry analysis. *Nuclear Instruments and Methods in Physics Research Section A* **422,** 423–427. doi:10.1016/S0168-9002(98)01110-3 (1999) (cit. on pp. 23, 27, 29).

43. Yoshida, E. *et al.* Application of neural networks for the analysis of gamma-ray spectra measured with a Ge spectrometer. *Nuclear Instruments and Methods in Physics Research A* **484,** 557–563. doi:10.1016/S0168-9002(01)01962-3 (2002) (cit. on pp. 23, 27).

44. Kangas, L. *et al.* The use of artificial neural networks in PVT-based radiation portal monitors. *Nuclear Instruments and Methods in Physics Research Section A* **587,** 398–412. doi:10.1016/j.nima.2008.01.065 (2008) (cit. on p. 23).

45. Goorley, T. *et al.* Initial MCNP6 Release Overview. *Nuclear Technology* **180,** 298–315. doi:10.13182/NT11-135 (2012) (cit. on pp. 23, 25, 42).

46. Horne, S. *et al.* GADRAS-DRF 18.5 User's Manual. *Sandia National Lab, Technical Report.* doi:10.2172/1166695 (2014) (cit. on pp. 23, 25, 42).

47. Chen, L. & Wei, Y. Nuclide identification algorithm based on K–L transform and neural networks. *Nuclear Instruments and Methods A* **598,** 450–453. doi:10.1016/j.nima.2008.09.035 (2009) (cit. on pp. 23, 27).

48. Maccone, C. *Deep Space Flight and Communications* ISBN: 978-3-540-72942-6. doi:10.1007/978-3-540-72943-3 (Springer, Berlin, Heidelberg, 2009) (cit. on p. 23).

49. Vargas, A. *et al.* Development of a neural network approach to characterise $^{2}$26Ra contamination at legacy sites using gamma-ray spectra taken from boreholes. *Journal of Environmental Radioactivity* **140,** 130–140. doi:10.1016/j.jenvrad.2014.11.011 (2015) (cit. on pp. 24, 27).

50. Pelowitz, D. MCNPX Users Manual Version 2.7.0. *Los Alamos National Laboratory,* LA–CP–11–00438 (2011) (cit. on pp. 24, 42).

51. Felipe-Blàzquez, L. *et al.* Classification of Radionuclides on Polyvinyl Toluene Radiation Portal Monitors by a Neural Network Based System. *IFAC-PapersOnLine* **48,** 852–857. doi:10.1016/j.ifacol.2015.09.633 (2015) (cit. on p. 24).

52. Bobin, C. *et al.* Charge carrier and exciton dynamics in LaBr3:Ce3+ scintillators: Experiment and model. *Applied Radiation and Isotopes* **109,** 405–409. doi:10.1016/j.apradiso.2015.12.029 (2016) (cit. on p. 24).

53. Bayes, T. An essay towards solving a problem in the doctrine of chances (1763). *MD Comput.* **8,** 157–171 (1991) (cit. on p. 24).

54. Kamuda, M., Zhao, J. & Huff, K. A comparison of machine learning methods for automated gamma-ray spectroscopy. *Nuclear Instruments and Methods in Physics Research Section A* **64,** 1858–1864. doi:10.1016/j.nima.2018.10.063 (2018) (cit. on pp. 24, 26–28, 41, 89).

55. Abadi, M. *et al.* TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. *Online resource, arXiv* (2015) (cit. on pp. 24, 90).

56. Devroye, L. Sample-based non-uniform random variate generation. *WSC '86 Proceedings of the 18th conference on Winter simulation,* 260–265. doi:10.1145/318242.318443 (1986) (cit. on p. 25).

57. Bridle, J. Probabilistic Interpretation of Feedforward Classification Network Outputs, with Relationships to Statistical Pattern Recognition. *NATO ASI Series (Series F: Computer and Systems Sciences)* **68.** doi:10.1007/978-3-642-76153-9_28 (1990) (cit. on p. 32).

58. Krizhevsky, A., Sutskever, I. & Hinton, G. ImageNet Classification with Deep Convolutional Neural Networks. *Advances in Neural Information Processing Systems* **25,** 1097–1105 (2012) (cit. on p. 33).

59. Bishop, C. *Pattern recognition and machine learning* p.209. ISBN: 978-0387-31073-2 (Springer, Cambridge, UK, 2006) (cit. on p. 35).

60. Srivastava, N. *et al.* Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research* **15.** http://jmlr.org/papers/v15/srivastava14a.html, 1929–1958 (2014) (cit. on p. 36).

61. Chollet, F. Keras. *GitHub repository.* https://github.com/fchollet/keras (2015) (cit. on p. 36).

62. Romano, P. *et al.* OpenMC: A state-of-the-art Monte Carlo code for research and development. *Annals of Nuclear Energy* **82,** 90–97. doi:10.1016/j.anucene.2014.07.048 (2015) (cit. on p. 42).

63. Leppänen, J. *et al.* The Serpent Monte Carlo code: Status, development and applications in 2013. *Annals of Nuclear Energy* **82,** 142–150. doi:10.1016/j.anucene.2014.08.024 (2015) (cit. on p. 42).

64. Collaboration, R. GEANT4: An object-oriented toolkit for simulation in HEP. *CERN Report, CERN/LHCC* **98** (1998) (cit. on p. 43).

65. Sempau, J. *et al.* Experimental benchmarks of the Monte Carlo code penelope. *Nuclear Instruments and Methods in Physics Research Section B* **207,** 107–123. doi:10.1016/S0168-583X(03)00453-1 (2003) (cit. on p. 45).

66. Hurtado, S., García-León, M. & García-Tenorioc, R. A revision of energy and resolution calibration method of Ge detectors. *Nuclear Instruments and Methods in Physics Research A* **564,** 295–299. doi:10.1016/j.nima.2006.03.018 (2006) (cit. on p. 52).

67. Debertin, K. & Helmer, R. *Gamma-and X-ray spectrometry with semiconductor detectors* (Elsevier Science and Technology, 1988) (cit. on p. 52).

68. Helmer, R. & McCullagh, C. GAUSS VII, a computer program for the analysis of $\gamma$-ray spectra from Ge semiconductor spectrometers. *Nuclear Instruments and Methods in Physics Research* **206,** 477–488. doi:10.1016/0167-5087(83)90385-X (1983) (cit. on p. 52).

69. Canberra. *Genie 2000 Spectroscopy Software, Operations Manual* 9230846K (Canberra Industries, Inc, 2006) (cit. on p. 52).

70. Bossew, P. A very long-term HPGe-background gamma spectrum. *Applied Radiation and Isotopes* **62,** 635–644. doi:10.1016/j.apradiso.2004.09.006 (2005) (cit. on p. 57).

71. Chen, J. Nuclear Data Sheets for A=40. *Nuclear Data Sheets* **140,** 1–376. doi:10.1016/j.nds.2017.02.001 (2017) (cit. on p. 57).

72. Hendriks, P., Maučec, M. & de Meijer, R. MCNP modelling of scintillation-detector $\gamma$-ray spectra from natural radionuclides. *Applied Radiation and Isotopes* **57,** 449–457. doi:10.1016/S0969-8043(02)00118-5 (2002) (cit. on p. 57).

73. Wallace, J. Monte Carlo modelling of large scale NORM sources using MCNP. *Journal of Environmental Radioactivity* **126,** 55–60. doi:10.1016/j.jenvrad.2013.06.009 (2013) (cit. on pp. 57, 59).

74. Novikov, E., Phlips, B. & Wulf, E. A $\gamma$-ray background model for Monte Carlo simulations. *Nuclear Instruments and Methods in Physics Research Section A* **579,** 279–283. doi:10.1016/j.nima.2007.04.056 (2007) (cit. on pp. 57, 59).

75. R. Merk and H. Kröger and L. Edelhäuser-Hornung and B. Hoffmann. PENELOPE-2008 Monte Carlo simulation of gamma exposure induced by $^{60}$Co and NORM-radionuclides in closed geometries. *Applied Radiation and Isotope* **82,** 20–27. doi:10.1016/j.apradiso.2013.07.006 (2013) (cit. on pp. 57, 59).

76. Sublet, J.-C. *et al.* FISPACT-II: An Advanced Simulation System for Activation, Transmutation and Material Modelling. *Nuclear Data Sheets* **139,** 77–137. doi:10.1016/j.nds.2017.01.002 (2017) (cit. on p. 58).

77. Britton, R., Jackson, M. & Davies, A. Quantifying radionuclide signatures from a $\gamma-\gamma$ coincidence system. *Journal of Environmental Radioactivity* **149,** 158–163. doi:10.1016/j.jenvrad.2015.07.025 (2015) (cit. on p. 60).

78. Pirrie, S. *et al.* Undergraduate Experiments with a $^{44}$Ti source. *European Journal of Physics (submitted)* (2020) (cit. on pp. 66, 71).

79. Hampel, D. M. Improving the Identification of Unstable Isotopes Via Measurement and Simulation. *University of Birmingham, Master's thesis* (2018) (cit. on p. 69).

80. Chen, J., Singh, B. & Cameron, J. Nuclear Data Sheets for A = 44. *Nuclear Data Sheets* **112,** 2357–2495. doi:10.1016/j.nds.2011.08.005 (2011) (cit. on p. 70).

81. Martin, M. Nuclear Data Sheets for A = 208. *Nuclear Data Sheets* **108,** 1583–1806. doi:10.1016/j.nds.2007.07.001 (2007) (cit. on p. 70).

82. Buuck, M. *et al. Angular Correlations in Gamma Deexcitation* (Geant4 RDM Mini Workshop 2017, 2017) (cit. on p. 72).

83. Brun, R. & Rademakers, F. ROOT - An object oriented data analysis framework. *Nuclear Instruments and Methods in Physics Research Section A* **389,** 81–86. doi:10.1016/S0168-9002(97)00048-X (1997) (cit. on p. 75).

84. Pattabiraman, N., Chintalapudi, S. & Ghugre, S. An automatic gain matching method for $\gamma$-ray spectra obtained with a multi-detector array. *Nuclear Instruments and Methods in Physics Research Section A* **526,** 432–438. doi:10.1016/j.nima.2004.02.032 (2004) (cit. on p. 81).

85. Deepak, P. *et al.* Fully convolutional multi-class multiple instance learning. *arXiv preprint,* arXiv:1412.7144 (2014) (cit. on p. 84).

86. Szegedy, C. *et al.* Rethinking the Inception Architecture for Computer Vision. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* 2818–2826. doi:10.1109/CVPR.2016.308 (2016) (cit. on pp. 90, 117).

87. Pedregosa, F. *et al.* Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* **12,** 2825–2830 (2011) (cit. on p. 90).

88. Tensorflow Initialisers. https://www.tensorflow.org/api_docs/python/tf/keras/initializers (cit. on p. 95).

89. He, K. *et al.* Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. *ICCV,* 1026–1034 (2015) (cit. on p. 95).

90. Kingma, D. & Ba, J. Adam: A method for stochastic optimization. *arXiv: 1412.6980,* Preprint (2014) (cit. on p. 95).

91. Hinton, G, Srivastava, N & Swersky, K. Lecture 6d- a separate, adaptive learning rate for each connection. *Slides of Lecture Neural Networks for Machine Learning* (2012) (cit. on p. 95).

92. Clevert, D., Unterthiner, T. & Hochreiter, S. Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289* (2015) (cit. on p. 97).

93. Shang, W. *et al.* Understanding and Improving Convolutional Neural Networks via Concatenated Rectified Linear Units. *Proceedings of the 33rd International Conference on Machine Learning,* 2217–2225 (2016) (cit. on p. 97).

94. Hubbell, J. & Seltzer, S. Tables of X-Ray Mass Attenuation Coefficients and Mass Energy-Absorption Coefficients from 1 keV to 20 MeV for Elements Z = 1 to 92 and 48 Additional Substances of Dosimetric Interest. *NIST Standard Reference Database 126.* doi:10.18434/T4D01F (2004) (cit. on p. 106).

95. Simonyan, K & Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014) (cit. on p. 117).

96. Breiman, L. Random Forests. *Machine learning* **45,** 5–32. doi:10.1023/A:1010933404324 (2001) (cit. on p. 118).

97. Pirrie, S., Kokalova, T., Wheldon, C., Bailey, S., Bishop, J., Curtis, N., Smith, R., Torresi1, D., Turner, A., Hertenberger, R., Wirth, H.-F., Faestermann, T., Mengoni, D. & Dell'Aquila, D. A method to determine $\gamma$ branching ratios using charged particle detectors for states in $^{18}$O. *AIP Conference Proceedings* **1.** doi:10.1063/1.5078856 (2018) (cit. on p. 2).

98. Trinder, R., Kokalova, T., Parker, D., Wheldon, C., Phoenix, B., Ivanov, P., Russel, B., Webster, B., Regan, P., Robinson, A., Cullen, D., Pells, S., Allen, R., Pirrie, S., Turner, A. & Santa-Rita, P. Theragnostics - Alternative production of terbium isotopes at the University of Birmingham using an MC40 cyclotron. *J. Phys.: Conf. Ser.* **1643.** 27th International Nuclear Physics Conference (INPC2019) (2019) (cit. on p. 2).

99. Griffiths, M., Wheldon, C., Kokalova, T., Turner, A., Pirrie, S., Ziman, V., Ashwood, N., Malcolm, J., Barr, M., Freer, M., Faestermann, T., Wirth, H.-F., Hertenberger, R., Gernhäuser, R. & Krücken, R. Mass measurement of Re-190. *Journal of Physics G: Nuclear and Particle Physics* **47,** 85–104. doi:10.1088/1361-6471/ab98e3 (2020) (cit. on p. 2).

100. Pirrie, S., Wheldon, C., Kokalova, T., Bishop, J., Hertenberger, R., Wirth, H., Bailey, S., Curtis, N., Dell'Aquila, D., Faestermann, T., Mengoni, D., Smith, R., D.Torresi & Turner, A. Clustering in $^{18}$O - absolute determination of branching ratios via high-resolution particle spectroscopy. *SciPost Phys. Proc.* **9.** doi:10.21468/SciPostPhysProc.3.009 (2020) (cit. on p. 2).

101. Pirrie, S., Tungate, G., Kokalova, T., Wheldon, C., Nikolopoulos, K., Rita, P. S. & Turner, A. Undergraduate Experiments with a $^{44}$Ti source. *European Journal of Physics (submitted)* (2020) (cit. on p. 3).

# Appendices

# Appendix A

# Publications

## A.1 List of other publications

Pirrie, S., Kokalova, T., Wheldon, C., Bailey, S., Bishop, J., Curtis, N., Smith, R., Torresi1, D., Turner, A., Hertenberger, R., Wirth, H.-F., Faestermann, T., Mengoni, D. & Dell'Aquila, D. A method to determine $\gamma$ branching ratios using charged particle detectors for states in $^{18}$O. *AIP Conference Proceedings* **1.** doi:10.1063/1.5078856 (2018)

Trinder, R., Kokalova, T., Parker, D., Wheldon, C., Phoenix, B., Ivanov, P., Russel, B., Webster, B., Regan, P., Robinson, A., Cullen, D., Pells, S., Allen, R., Pirrie, S., Turner, A. & Santa-Rita, P. Theragnostics - Alternative production of terbium isotopes at the University of Birmingham using an MC40 cyclotron. *J. Phys.: Conf. Ser.* **1643.** 27th International Nuclear Physics Conference (INPC2019) (2019)

Griffiths, M., Wheldon, C., Kokalova, T., Turner, A., Pirrie, S., Ziman, V., Ashwood, N., Malcolm, J., Barr, M., Freer, M., Faestermann, T., Wirth, H.-F., Hertenberger, R., Gernhäuser, R. & Krücken, R. Mass measurement of Re-190. *Journal of Physics G: Nuclear and Particle Physics* **47,** 85–104. doi:10.1088/1361-6471/ab98e3 (2020)

Pirrie, S., Wheldon, C., Kokalova, T., Bishop, J., Hertenberger, R., Wirth, H., Bailey, S., Curtis, N., Dell'Aquila, D., Faestermann, T., Mengoni, D., Smith, R., D.Torresi & Turner, A. Clustering in $^{18}$O - absolute determination of branching ratios via high-resolution particle spectroscopy. *SciPost Phys. Proc.* **9.** doi:10.21468/SciPostPhysProc.3.009 (2020)

Pirrie, S., Tungate, G., Kokalova, T., Wheldon, C., Nikolopoulos, K., Rita, P. S. & Turner, A. Undergraduate Experiments with a $^{44}$Ti source. *European Journal of Physics (submitted)* (2020)

## A.2 Copy of thesis publications

See below for copies of the listed thesis publications.

Journal of Physics: Conference Series **1643** (2020) 012211 doi:10.1088/1742-6596/1643/1/012211

# Generalised gamma spectrometry simulator for problems in nuclide identification

**A. Turner[a], C. Wheldon[a], M.R. Gilbert[b], L.W. Packer[b], J. Burns[c], Tz. Kokalova Wheldon[a], M. Freer[a]**

[a] University of Birmingham, Edgbaston, Birmingham, B15 2TT, UK
[b] UKAEA, Culham Science Centre, Abingdon, Oxfordshire, OX14 3DB, UK
[c] AWE plc., Aldermaston, Reading, RG7 4PR, UK

**Abstract.** Improvements in Radio-Isotope IDentification (RIID) algorithms have always been a continuous research focus. However, significant developments in machine learning have recently sparked renewed interest. To provide a rapid development environment for this, a generalised gamma simulator has been built using the GEANT4 toolkit. This enables consideration of a diverse range of radiation sources and shielding scenarios. The simulator currently provides training data for the development of neural network based RIID models.

## 1. Introduction

Radio-Isotope IDentification (RIID) algorithms have broad application across alarms and detection, identification, and mapping [1]. Often these are developed in the context of security, but are readily applied to problems in decommissioning and survey [2–6]. The increased accessibility of machine learning techniques has recently produced many exciting new directions for this problem space [7, 8].

Success in these applications come down to the detector, electronics, and the RIID algorithm. Since the widely adopted detectors rarely change, it is the algorithms that often develop. The goal is to infer the identity of all isotopes present in a source and to quantify as many properties as possible [6]. Research is continuous because it can be extremely difficult to automate. There are a myriad of competing effects that can impact gamma ray spectra such as shielding, gain shifts, and changing environmental conditions. Even the system itself is bound by practical considerations of robustness, availability, and affordability [9].

To develop algorithms it has become common practice to use simulated spectra that augment the real data, which are often limited. This requires a simulator that can generate accurate gamma spectra for a diverse range of radiation sources, detectors, and scenarios. Such an environment is conducive to rapid development in RIID.

GEANT4 [10] is chosen to fulfil the role of simulator due an unmatched flexibility when compared with MCNP [11] or GADRAS [12]. Timing information is readily accessible with a GEANT4 based simulator, as are the energy depositions needed to generate a spectrum. The fundamental focus is that of per-decay accuracy rather than a statistical approach, allowing for more subtle effects to be well represented in the resulting spectra.

## 2. Project scope

A significant part of generalising a simulator is the ability to model multiple detector types. This becomes as simple as swapping out models and redefining a few parameters in the simulator. For these proceedings, evaluations using the HPGe model are chosen for demonstrative purposes. Of course the NaI, $LaBr_3$, and PVT models are also of particular interest to RIID, but many features of the spectrum are clearer with the superior resolution of HPGe.

Bespoke solutions to timing or true coincidence summations are undesirable. MCNP, for example, requires the emission probabilities of every possible particle. Summations must therefore be pre-calculated based on the simulation geometry and detector properties, and be appended to the source definition. This requires additional preprocessing steps liable to user error. True coincidence summations are of particular interest since it is common practice to simulate larger stand-off distances and avoid the issue. By accurately reproducing true coincidence summation peaks, extra information becomes available for RIID algorithms in low activity, close geometry scenarios. Finally, heavily shielded scenarios will be considered as another difficulty RIID typically faces.

## 3. Producing energy spectra

### *3.1. Gaussian energy broadening*

Fundamental to simulating any gamma-ray spectrum, each energy deposit is sampled from a Gaussian distribution with parameters derived from experimental data. This emulates the expected energy broadening, with the width ($\sigma = fwhm/2\sqrt{\ln(2)}$) from the empirical [13],

$$fwhm = a + b\sqrt{E + cE^2}. \tag{1}$$

Note that this describes full energy peaks that do not involve annihilation photons. In real data, these are broader than expected, leading to a distinctly separate dataset shown in Figure 1. The low energy data have broader FWHM due to contributions from background and other very close transitions, but the empirical fit reproduces the low energy region well when simulated.

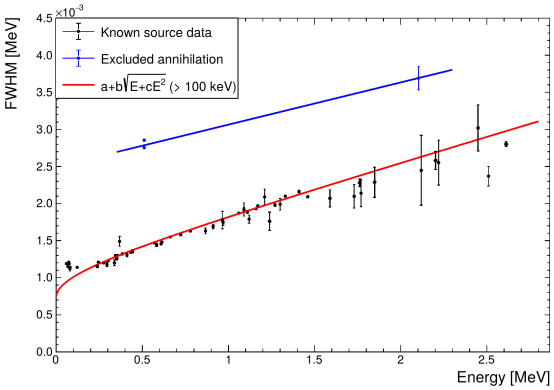



**Figure 1.** FWHM calibration curve for the HPGe model. Broader annihilation peak related data excluded from core set.

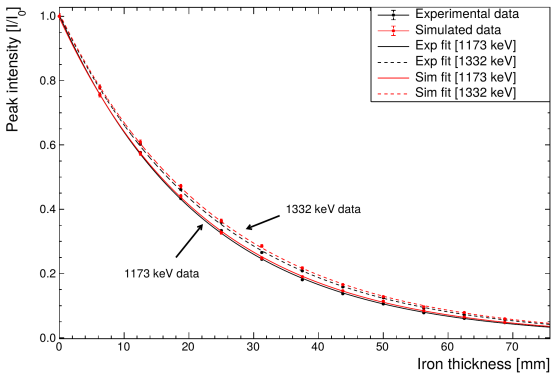



**Figure 2.** Comparison of gamma attenuation with real data for $^{60}$Co, with up to 70 mm of iron plates.

### *3.2. Annihilation*

A net residual momentum of the $e^-/\beta^+$ results in additional broadening of these annihilation related events. Below ~2.0 MeV special consideration may be given to the 511 keV. More rigorously, the relevant tracks may be flagged during tracking to force sampling form an alternate parameter set. This may be taken from the annihilation curve shown in Figure 1.

### *3.3. Decay chain limiting*

GEANT4 will follow a decay chain for as long as no stable isotope is reached. This can result in unrealistic spectra over experimental time scales. A perfect example is $^{241}$Am alpha decaying to $^{237}$Np, which has a $2\times10^{6}$ yr half-life. A cut-off based on lifetime for any secondary nuclei is implemented, killing tracks above an acceptable threshold. Without it, simple spectra like $^{241}$Am will end up with contributions from the entire neptunium decay series.

### *3.4. X-ray fluorescence*

While often very minor, the effects of X-ray fluorescence can be represented at the lower energies of gamma spectra. Characteristic X-rays are often emitted following a photoelectric absorption as electrons move to fill the inner-shell vacancy. Since low-energy gamma interactions are more likely to occur close to the crystal surface, these X-rays can escape the crystal without depositing any energy. The resulting feature is known as an X-ray escape peak.

Production cuts are tailored to the detector material. A threshold is set as high as possible while being low enough to allow X-rays to be produced by atomic relaxation models. Only crystal volumes are assigned to a special 'region'. Custom production cuts are specified for these alone, leaving other volumes a compromise between speed and fidelity. This is based on consideration of dead layer attenuation expectations, distance/composition of surrounding materials, and most importantly the impact on the final energy spectrum.

### *3.5. Multi-isotope sources*

The preferred method of simulating a multi-isotope source is to sample from a list of nuclides based on activity, then decay that nuclide for the current event. This is true of most MCNP implementations. Here, all sources are defined at the start of each run, with one nuclide randomly chosen for each event based on relative activity.

For very active sources, where random summations can occur with other sources or even themselves, the probability of two nuclides decaying at the same time for a given mass and activity can be calculated. Sampling based on this probability, multiple nuclides may be defined in the same event and forced to 'decay' at the same time. This is currently unnecessary for RIID purposes. Anything of high enough activity for it to be an issue will be flagged by gross counts alone and can be manually inspected.

### *3.6. Shielding*

Including shielding in the geometry is as simple as modelling it. All elements and their properties are predefined in GEANT4, as are a selection of common materials. New compounds and materials may also be user defined with relative abundances and densities. Taking the simple example of several iron plates between a $^{60}$Co source and the detector, the photopeak intensities may be found as a function of thickness (Figure 2). Attenuation coefficients may then be calculated from the exponent.

## 4. Timing and summation peaks

In GEANT4 every particle propagation object contains a huge amount of information, including tracked times. One of the driving factors for the switch from MCNP to GEANT4 was this access to timing information. Having these times is excellent for setting up multi-detector coincidences, but also crucial for accurate summation peaks. Precise timing and accurate summation peaks are inextricably linked; arrival times are required to decide whether two energy deposits should be summed, and long radioactive time scales make this a challenge to do in a generalised manner.

### *4.1. Radioactive decay time scales*

A 'global time' tracks the time passed since an event began. Its nanosecond base unit is stored in C++ double precision. Rounding inevitably becomes detrimental to timing precision, especially with longer lifetimes. Calculating small time differences after million year half-lives becomes unreliable and will often round to zero. If particles of interest are known it is easy to create bespoke solutions, but for a generalised simulator this must be taken care of automatically.

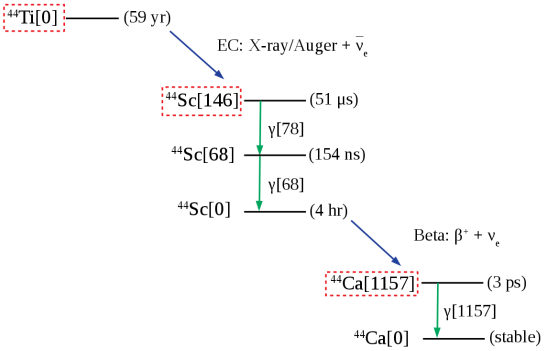



**Figure 3.** Simplified $^{44}$Ti decay scheme.

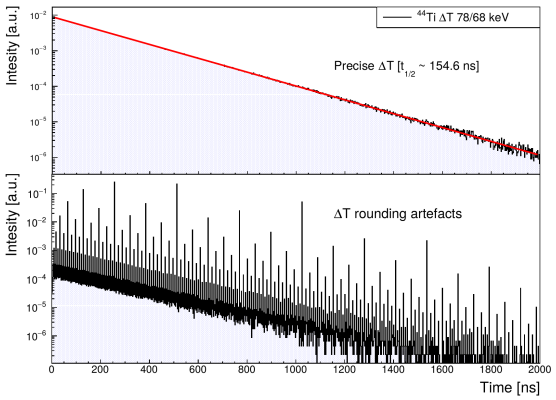



**Figure 4.** Rounding artefacts in timing.

Taking the $^{44}$Ti decay chain as an example (Figure 3), the difference in creation times of the 68 keV and 78 keV photons are collected. Experimentally this investigates the half-life of the interim $^{44}$Sc excited state. Simply subtracting global times results in the artefacts seen in Figure 4, even after a relatively short 59 yr initial half-life.

Resetting the global time to zero at key points is the simplest solution, noting when a reset happens. Time differences are then calculated relative to the nearest common reset point rather than the start of an event. In doing so, magnitudes are kept small enough that rounding is no longer an issue.

For example, a reset point at $^{44}$Sc[146] (shown in Figure 3) can be forced. The global times being subtracted for the previous calculation are now on the order of microseconds rather than years. By implementing code to check time differences between any two given secondary particles, a 'trigger' system analogous to experimental methods can be set up based on energy. With global time magnitudes now under control, the same subtraction produces an exponential without rounding errors, as seen in the top half of Figure 4. This works well for even a 17 ps state in $^{208}$Tl after the full $^{232}$Th decay chain (in which it is populated) is followed within a single event.

### *4.2. True coincidence summing*

Following a radioactive decay in GEANT4, the entire series of secondary photons and particles becomes part of the same event. True summations therefore naturally occur in these simulations. Taking $^{60}$Co as a standard example, the intensities of two primary photo-peaks and their corresponding sum peak may be compared to experimental data (Figure 5).

The trends closely follow a solid angle ($\Omega$) dependence as expected, becoming $\Omega^2$ for the summation. A naïve approach to energy deposition, where everything entering the crystal is summed, works fine for $^{60}$Co. However, for more complex decay schemes it can result in spurious summations (Figure 6). Any $^{44}$Sc emissions occur *hours* before any annihilation photons in the $^{44}$Ti example of Figure 3.

To solve this, time differences between tracks in the crystal must be found and handled accordingly. For gamma detectors all energy deposition comes from secondary particles produced

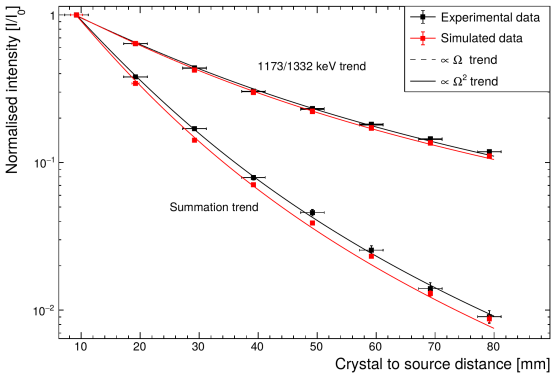



**Figure 5.** $^{60}$Co photo-peak intensities as a function of source distance.

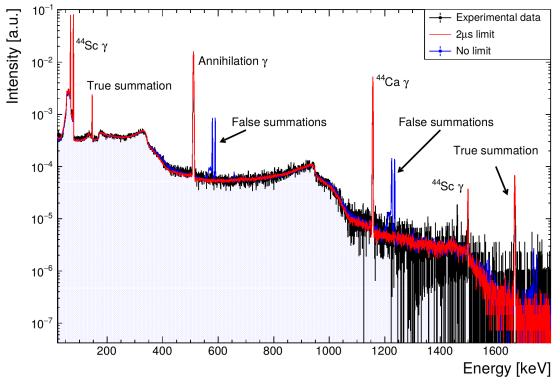



**Figure 6.** Demonstration of spurious $^{44}$Ti summations with no timing information used.

within the crystal. Reconstruction of full, independent tracks is therefore carried out by following parent-child chains. Once reconstructed, the total energy deposited for each track may either be summed or considered separate based on an appropriate time window.

A 2 $\mu$s window for the ADC conversion time seen in the red spectrum of Figure 6 demonstrates this method removing spurious peaks while keeping accurate true summations. This generalised use for any source is only possible by maintaining good timing precision over radioactive decay time scales.

### *4.3. Multi-detector coincidences*

With precision maintained in emission time differences, only travel times are further required to allow for multi-detector coincidence measurements. Summation peaks also simply need an extra condition to check that two candidate tracks enter the same detector.

A coincidence measurement is a combination of emission and travel time differences, with broadening from intrinsic physical properties like scintillation lifetimes. Just as energy resolution is sampled from a Gaussian based on experimental data, so too is a time resolution. Figure 7 shows the inclusion of an arbitrary 35 ns time resolution. The fit in this figure is a convolution of a Gaussian and exponential [14], confirming accurate representation of the simulation parameters.

With multiple detectors and good timing information, angular correlations may also be investigated. At the time of writing, the GEANT4 source only has a demonstrative implementation for select isotopes; the simple $4^+ \rightarrow 2^+ \rightarrow 0^+$ cascade in $^{60}$Co. Checking with $^{108m}$Ag it is clear that any angular correlations must be manually implemented by the user.

Of course, the $4^+ \rightarrow 2^+ \rightarrow 0^+$ cascade in $^{108m}$Ag mirrors that of $^{60}$Co and works well (Figure 8). However, no correlation is seen for the $6^+ \rightarrow 4^+ \rightarrow 2^+$ cascade in the same decay. While the amplitude should be slightly suppressed, the phase should be exactly the same as the transitions that follow.

## 5. Current limitations

As with any simulator there are inherent limitations in fidelity achievable due to environmental effects. The Naturally Occurring Radioactive Background (NORM) is primarily from the $^{235}$U and $^{232}$Th decay series, along with a significant $^{40}$K photopeak. This corresponds to contributions from only ∼25 distinct radioactive nuclides. However, local contaminations, activations, and relative intensities all make it difficult to define a representative template. The same is true of backscatter contributions to spectra. Any photons scattering back into the detector from surrounding materials are dependent on accurate reproduction of the nearby environment.

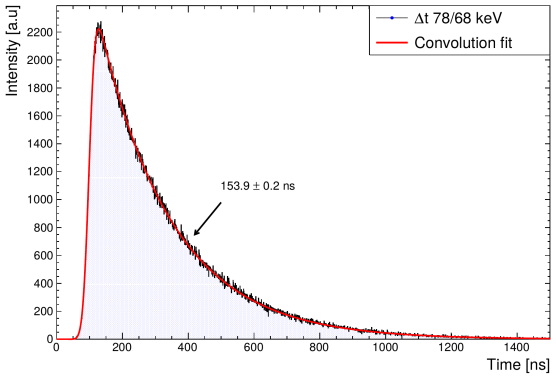



**Figure 7.** Example time spectrum for half-life measurement of $^{44}$Sc.

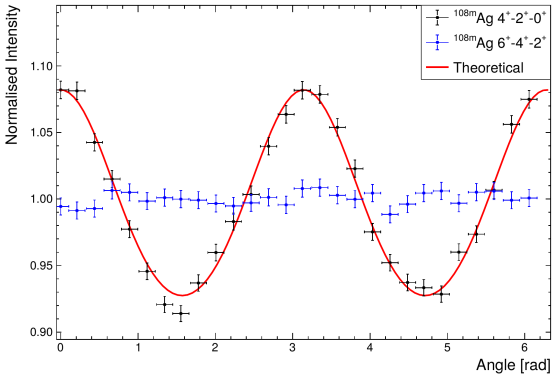



**Figure 8.** Demonstration of limited angular correlation implementation using $^{108m}$Ag.

## 6. Simulator usage

This simulator is currently serving the crucial role of providing machine learning models with training data. These data are generated as needed for rapid testing of diverse scenarios. Of course, corresponding test data are experimentally collected for final evaluations to present better representations of real world performance. Accurate coincidence summations allow investigations of closer geometries, and whether the additional peaks are useful for improved RIID. Setting up multiple detectors also provides coincidence data that may give the machine learning models extra relevant information.

## References

[1] C.J. Sullivan and J. Lu. Automated photopeak detection and analysis in low resolution gamma-ray spectra for isotope identification. *Nuclear Science Symposium and Medical Imaging Conference*, IEEE, 2013. DOI: 10.1109/NSSMIC.2013.6829437.

[2] J.M. Blackadar. Automatic isotope identifiers and their features. *IEEE Sensors Journal*, 5(4):589–592, 2005. DOI: 10.1109/JSEN.2005.846374.

[3] T. Burr, M.S. Hamada, T.L. Graves, and S. Myers. Augmenting real data with synthetic data: an application in assessing Radio-Isotope identification algorithms. *Quality and Reliability Engineering International*, 25(8):899–911, 2009. DOI: 10.1002/qre.1003.

[4] C.J. Sullivan, S.E. Garner, M. Lombardi, K.B. Butterfield, and M.A. Smith-Nelson. Evaluation of key detector parameters for isotope identification. *IEEE Nuclear Science Symposium Conference Record*, 2:1181–1184, 2007. DOI: 10.1109/NSSMIC.2007.4437217.

[5] D.C. Stromswold, J.W. Darkoch, J.H. Ely, R.R. Hansen, R.T. Kouzes, B.D. Milbrath, R.C. Runkle, W.A. Sliger, J.E. Smart, D.L. Stephens, L.C. Todd, and M.L. Woodring. Field tests of a NaI(Tl)-based vehicle portal monitor at border crossings. *Nuclear Science Symposium Conference Record*, IEEE, 2004. DOI: 10.1109/NSSMIC.2004.1462180.

[6] T. Burr and M. Hamada. Radio-Isotope Identification Algorithms for NaI $\gamma$ Spectra. *Algorithms*, 2(1):339–360, 2009. DOI: 10.3390/a2010339.

[7] M. Kamuda, J. Stinnett, and C.J. Sullivan. Automated Isotope Identification Algorithm Using Artificial Neural Networks. *IEEE Transactions on Nuclear Science*, 64(7):1858–1864, 2017. DOI: 10.1109/TNS.2017.2693152.

[8] D. Liang, P. Gong, X. Tang, P. Wang, L. Gao, Z. Wang, and R. Zhang. Rapid nuclide identification algorithm based on convolutional neural network. *Seminars in nuclear medicine*, 133:483–490, 2019. DOI: 10.1016/j.anucene.2019.05.051.

[9] M. Swoboda, R. Arlt, V. Gostilo, A. Lupilov, M. Majorov, M. Moszynski, and A. Syntfeld. Spectral gamma detectors for hand-held radioisotope identification devices (RIDs) for nuclear security applications. *IEEE Transactions on Nuclear Science*, 52(6):3111–3118, 2005. DOI: 10.1109/NSSMIC.2004.1466839.

[10] S. Agostinelli et al. Geant4 - a simulation toolkit. *Nuclear Instruments and Methods in Physics Research Section A*, 506(3):250–303, 2003. DOI: 10.1016/S0168-9002(03)01368-8.

[11] T. Goorley et al. Initial MCNP6 Release Overview. *Nuclear Technology*, 180(3):298–315, 2012. DOI: 10.13182/NT11-135.

[12] S.M. Horne, G.G. Thoreson, L.A. Theisen, D.J. Mitchell, L. Harding, and W.A. Amai. GADRAS-DRF 18.5 User's Manual. *Sandia National Lab, Technical Report*, 2014. DOI: 10.2172/1166695.

[13] G. Gilmore. *Practical Gamma-ray Spectroscopy, 2nd Edition*. John Wiley and Sons, 2011. ISBN 9781119964698.

[14] C. Wheldon. *Convolution of a Gaussian with an exponential (and its application in programs halflife.c/nanofit.f)*. University of Birmingham, 2014.

Definitions/sensors-logo-eps-converted-to.pdf

*Article*

# Convolutional neural networks for challenges in automated nuclide identification

**Anthony N. Turner [1], Carl Wheldon [1], Tzanka Kokalova Wheldon [1], Mark R. Gilbert [2], Lee W. Packer [2], Jonathan Burns [3], Martin Freer [1]**

[1] School of Physics and Astronomy, University of Birmingham, Edgbaston, Birmingham, B15 2TT, UK
[2] UKAEA, Culham Science Centre, Abingdon, Oxfordshire, OX14 3DB, UK
[3] AWE plc., Aldermaston, Reading, RG7 4PR, UK
* Correspondence: a.turner@pgr.bham.ac.uk

Received: date; Accepted: date; Published: date

**Abstract:** Improvements in Radio-Isotope IDentification (RIID) algorithms have seen a resurgence in interest with the increased accessibility of machine learning models. Convolutional Neural Network (CNN) based models have been developed to identify arbitrary mixtures of unstable nuclides from gamma spectra. In service of this, methods for the simulation and pre-processing of training data were also developed. The implementation of 1D multi-class, multi-label CNNs demonstrated good generalisation to real spectra with poor statistics and significant gain shifts. It is also shown that even basic CNN architectures prove reliable for RIID under the challenging conditions of heavy shielding and close source geometries, and may be extended to generalised solutions for pragmatic RIID.

**Keywords:** radio-isotope identification; convolutional neural network; gamma spectrometry; simulations; GEANT; nuclear applications

## 0. Introduction

Radio-Isotope IDentification (RIID) finds a broad scope of applications in security, decommissioning, and public health & safety. Each come with their own challenges, but the effectiveness of any detection system stems from physical limitations and algorithm performance. The ubiquity of HPGe and NaI detectors across the nuclear industry establishes a consistent landscape that is ideal for RIID development, keeping physical constraints relatively predictable.

In contrast to advances in detector materials, identification algorithms are constantly developed and updated. Persistent challenges in the explicit handling of transient effects (such as gain drift) have led to new approaches with machine learning [1–6]. By definition, these models operate without being explicitly programmed, instead making decisions based on generalised rules born of experience.

Artificial Neural Networks (ANN) have dominated the deep learning RIID approaches since their inception [1]. These typically 'fully-connected' models have a huge number of trainable parameters for even simple networks. Efforts must consistently focus on reducing the number of input features given to the model, often discarding information. Several interesting approaches for extracting underlying signals from background noise include Principle Component Analysis [2,7] and the closely related K-L transform [5]. Others rephrase the problem, performing feature extraction manually [1,3,4,6] and providing these as input data instead of raw spectra.

A variant known as the Convolutional Neural Network (CNN) may provide new directions [8–10] that inherently mitigate many of the issues ANNs suffer from, while possessing unique advantages beneficial to the RIID problem space. This work explores the application of sequential CNN models as pragmatic RIID solutions, able to address the challenges of transient effects and shielded sources.

## 1. Spectra for training and testing

Any machine learning technique requires training, validation, and test data to work with. A generalised gamma spectrometry simulator was developed for high-fidelity modelling of a diverse range of data sets. The details of this GEANT4-based [11] simulator are available in Ref. [12]. The synthetic spectra generated are used to train CNN models, monitoring for undesirable effects such as over-fitting.

A 3 MeV range was simulated for nine sources that are readily available to this work: $^{108m}$Ag, $^{241}$Am, $^{133}$Ba, $^{207}$Bi, $^{60}$Co, $^{137}$Cs, $^{152}$Eu, $^{22}$Na, and $^{44}$Ti. To demonstrate that these models may be used even where experimental spectra are absent, the reference model (Section 4.2) was trained entirely on simulated data, then tested on unseen experimental data.

Unfortunately, even when simulated, any comprehensive training set would take months to produce if all variations in physical effects, geometries, source combinations, and shielding scenarios were to be included. By simulating 'template' spectra instead, physically accurate variations in gain shift, count rate, and relative activity of multiple isotopes can be randomly varied to make new, equally valid training spectra. The analogy to image data is changing colour balances or rotating the original; a common practice in data science.

### *1.1. Template spectra*

Normalised template spectra were combined into multi-isotope spectra with experimental background for every unique combination of sources. Each template contributes a share of the total counts based on randomised factors (0.2-1.0). To provide significantly more training spectra with poorer statistics in the $\sim 10^3$-$10^6$ counts range, this total sampled from a logarithmic scale. A reasonable minimum of 500 counts per source was ensured. Background radiation was capped at 25% of the total counts from all other sources.

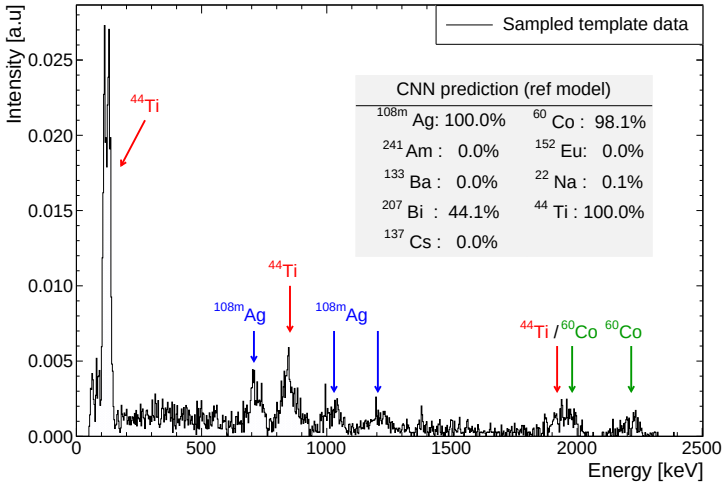



**Figure 1.** Example $4\times10^3$ count spectrum containing natural background, $^{108m}$Ag, $^{44}$Ti and $^{60}$Co. A 31 keV discrimination energy and gain shift of $\times 1.7$ are applied. Note the ambiguity in predictions made by the reference model (See Sec. 4.2) for $^{207}$Bi, which is extremely similar in profile to $^{44}$Ti.

Lower energy thresholds were applied with a 40 keV maximum for the sake of $^{241}$Am, which predominantly emits a 59.5 keV $\gamma$-ray. A small lower threshold is often used in experimental measurements to exclude low level noise, but 40 keV is large enough to remove significant features from several spectra. Gain shifts were implemented for a large 0.5-2.0 range of factors, emulating extreme transients in temperature or signal processing electronics to test robustness. Resulting spectra were re-binned using interpolated counts to ensure a consistent channel structure, with photopeaks now shifted from the calibrated energy. Figure 1 shows an example of these randomised perturbations.

Finally, spectra were compressed from 8192 to 1024 channels to clarify features in those with poor statistics, normalising the intensities to unity for consistency. The same process was applied to templates of experimentally obtained data, though the background must be removed before sampling. All data pre-processing used the ROOT [13] framework. Since training data load into memory, a data generator was used for scalability. With this, a buffer of several batches is loaded in parallel while the model is training.

### *1.2. Test spectra*

For testing, data sets were prepared for a 3″×3″ NaI detector at 1, 5, and 10 cm from the source to investigate close geometry effects. For shielded scenarios, up to 6 cm of aluminium shielding was placed between the source and detector, with the source fully obscured. All simulated scenarios have four training sets and six test sets of experimental data for performance testing, each using unique random seeds for including further perturbations. Quoted classification rates and results are taken as the average across all test sets.

## 2. Convolutional Neural Networks

Originally designed for image classification, the core of a CNN optimises successive 'filter' sets that are convolved with an input image. ANNs consider every pixel independent, with each pixel connected to every part of a hidden layer. Complexity therefore rapidly increases with scale. By comparison, CNN convolution layers train only a set of small (typically 3×3) filters instead. Computational expense is significantly reduced, scaling almost linearly with the number of input samples. Context is also considered by including information from neighbouring pixels. Applying this architecture to gamma spectra, where many important 'features' (such as photopeaks) are localised over channels, is very beneficial.

By convolving a gamma spectrum with filters, new abstractions of features are created. The output tensor of each convolution becomes a 'feature map'. More complex features are learnt by building upon multiple convolution layers. The first layers typically end up performing basic operations such as edge detection.

Since filters are optimised during training, the model learns to extract features and decide which are important for the final identification. This places the responsibility of feature extraction on the model with minimal manual pre-processing. A small, fully-connected structure following the convolution operations provides the probability of each source being present in the spectrum. An overview of the model architecture is shown in Figure 2. Presented in this work is an approach to RIID using 'multi-class, multi-label' classification based on the special 1D case of a CNN. An overview of this network type is provided in the next section, with key choices concerning certain parameters discussed in the Results (Section 4).

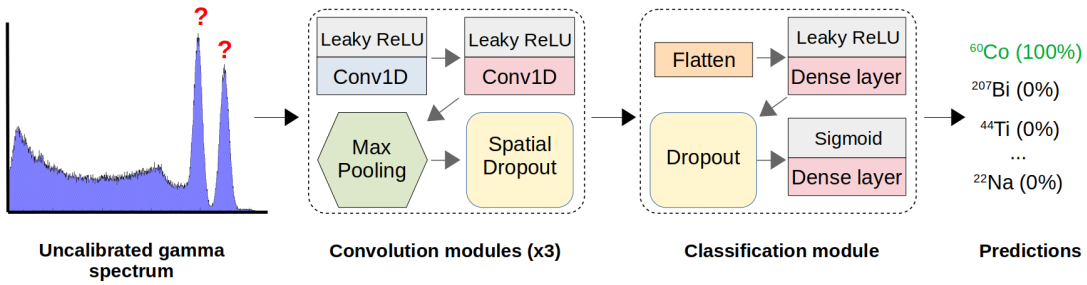



**Figure 2.** Simplified CNN structure for radio-isotope identification. The sequential model consists of several convolution modules, each performing 1D convolution operations to extract relevant features. For classifications, the resulting tensor is flattened into a vector to interface with a small, fully-connected ANN. The final output layer provides the probability of each source being present in a gamma spectrum.

## 3. Methods

### *3.1. Model definition and learning*

Rapid uptake of machine learning methods in the last five years has been driven by accessibility. This work utilises the open-source TensorFlow [14] libraries. Such powerful tools would otherwise require a great deal of effort to implement; another limiting factor in the complexity of early models following Olmos *et al.* [1].

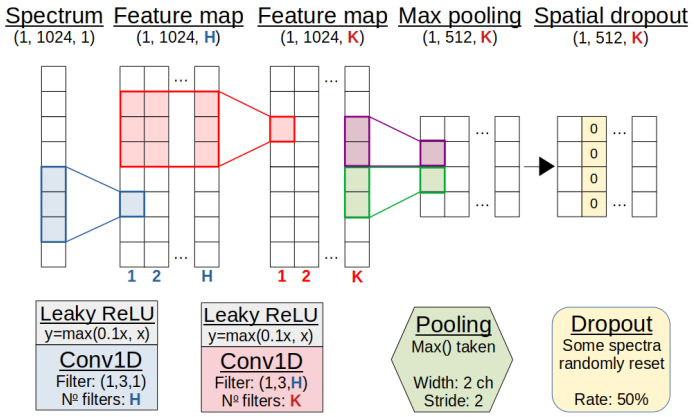



**Figure 3.** Representation of the convolution module. An array of intensities represents the spectrum. This trains first $H$ filters, then $K$ filters in two convolution layers to emphasise key features in the spectra. Pooling and dropout then reduce the dimensionality and combat over-fitting. For demonstration purposes, the shapes exclude a dimension for batch size. See text for details.

This CNN model consists of groups of 1D convolution layers in three modules (Fig. 3), each with many filters. By stacking two convolution operations in each module, a larger channel range is seen with fewer parameters, while intermediate activations exaggerate weights to make features more expressive. These activation functions introduce non-linearity into the model, with Rectified Linear Unit (ReLU) based functions ($y = max(0, x)$) chosen over legacy functions due to their robust nature.

The 'max pooling' operation down-samples convolved spectra by propagating only the maxima of non-overlapping channel groups. This reduces the size of output tensors while retaining important features. 'Spacial dropout' is also employed, removing spectra from feature maps at random to prevent over-dependence on any particular path for a final classification.

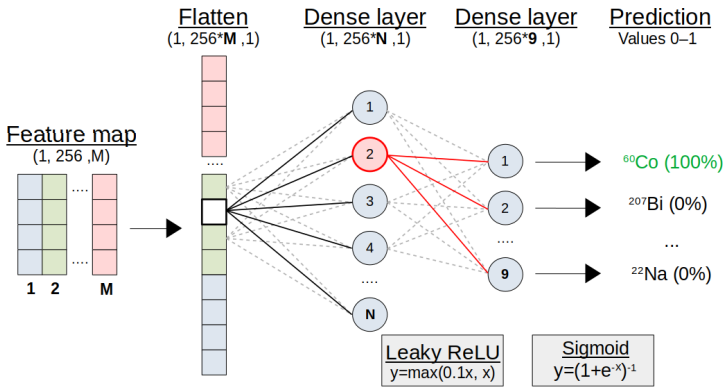



**Figure 4.** Representation of the classification module. The convolution module output is flattened into a vector. A small 'fully-connected' ANN makes final predictions of present sources, with a 0-1 probability through a Sigmoid activation. See text for details.

With the tensor of the final convolution module flattened into a vector, classifications are decided through a small ANN (Fig. 4). This may be only two fully-connected or 'dense' layers, again randomly resetting individual nodes with dropout to discourage over-fitting. The final layer and its activation function decide the type of classification.

For practical RIID with a single model (Fig. 2), this is a multi-class, multi-label problem. Spectra contain arbitrary combinations of multiple labels, each corresponding to one of the nine unique classes (sources). Arrays of labels are encoded to designate sources as either present (1) or not present (0) for each spectrum. The model is compiled with the Adam optimiser [15] and a binary cross-entropy loss function [16]. All parameters relating to the model structure and compilation are known as 'hyperparameters', with specific examples listed in the results of Table 2.

### *3.2. Performance evaluation*

Evaluation of this multi-label classification problem is best described by building upon binary concepts. The final Sigmoid activation ($y = (1 + e^{-x})^{-1}$) encodes the probability of each source being present in a test spectrum. Models are trained on a decision threshold of 0.5, with probabilities $> 0.5$ defining a 'positive' (1) classification. If the source is actually present, it is correctly identified as a 'True Positive (TP)'. Correspondingly the other possible outcomes are shown in Table 1.

**Table 1.** Simple binary classification outcomes.

| | | Actual label | |
|---|---|---|---|
| | | 1 | 0 |
| Prediction | 1 | True Positive (TP) | False Positive (FP) |
| | 0 | False Negative (FN) | True Negative (TN) |

Performance may vary as this decision threshold changes. For the balanced training sets used, a Receiver Operator Characteristic (ROC) provides a good indication of performance with threshold. To produce an ROC plot, True Positive Rate (TPR) and False Positive Rate (FPR) pairs are plotted for each threshold (t), defined as,

$$TPR = \frac{TP}{TP + FN}, \quad and \quad FPR = \frac{FP}{FP + TN}. \tag{1}$$

For RIID applications, the TPR is a measure of ability to find all sources present, while FPR is the probability of a false alarm. A closely related metric is 'precision', or how relevant the predictions are.

$$precision = \frac{TP}{TP + FP} \tag{2}$$

The ROC curve is extended to the multi-label case by considering each label individually, and together as an average. This information may be summarised in a single value by the Area Under Curve (AUC) as,

$$AUC = \int_{-\infty}^{\infty} TPR(t) FPR'(t) \; dt. \tag{3}$$

The 'Perfect' Prediction Rate (PPR) is also used in this work as a complementary measure of RIID performance given the real world applications. For example, averaging rates over the many sources may fail to reflect a consistently poor performance when making predictions for one in particular. Predictions are only considered perfect when every possible source is correctly identified as either TP or TN, acting as a much harsher performance metric. The PPR is simply the fraction of test spectra that had such perfect overall predictions.

## 4. Results and discussion

Performance of any RIID model based on machine learning is dependent on two primary aspects. First, the model's architecture and hyperparameters must be chosen with desirable properties in mind. One example is the dropout rate of a dense layer, where a balance must be struck between dropping so few nodes that over-fitting is a risk, or so many that performance suffers. The second consideration must be the data sets used to train the model. For example, larger data sets are generally better for deep learning because the model has more representative samples to learn from. An investigation of both aspects was carried out. The same model architecture dubbed the 'reference model' was used to evaluate all data sets, including the challenges of heavy shielding and close geometries.

### *4.1. Reference model*

For every unique combination of the nine sources, only 20 spectra were produced to build a small, entirely simulated training set for the 3" NaI detector at 10 cm. Relatively few spectra ($\sim 1 \times 10^4$) means that if the model does not generalise well, performance will suffer. Significant over-fitting will also make poor predictions on the unseen real data. Several equally small test sets were produced from experimental spectra, provided to models as uncalibrated spectra with no pre-processing beyond rebinning to 1024 channels and count normalisation.

Combinations of the two most common optimisers, Adam [15] and RMSProp [17], were assessed with a variety of initialisers. While consistent across initialisers, a difference between the two optimisers saw PPR change by 3.2(1)% on average. The marginally better He normal [18] initialiser was chosen to randomly set initial weights from a truncated normal.

While the second dense layer (Fig. 4) must have nine nodes (one for each class), the first can vary and have significant impact in comparison to the other hyperparameters. Figure 5 demonstrates how important the size of this small ANN is for making good use of the features extracted by the convolution layers. There is a clear degradation in performance when using a smaller ANN, but the computational expense rapidly increases with the number of nodes, thereby encouraging a compromise.

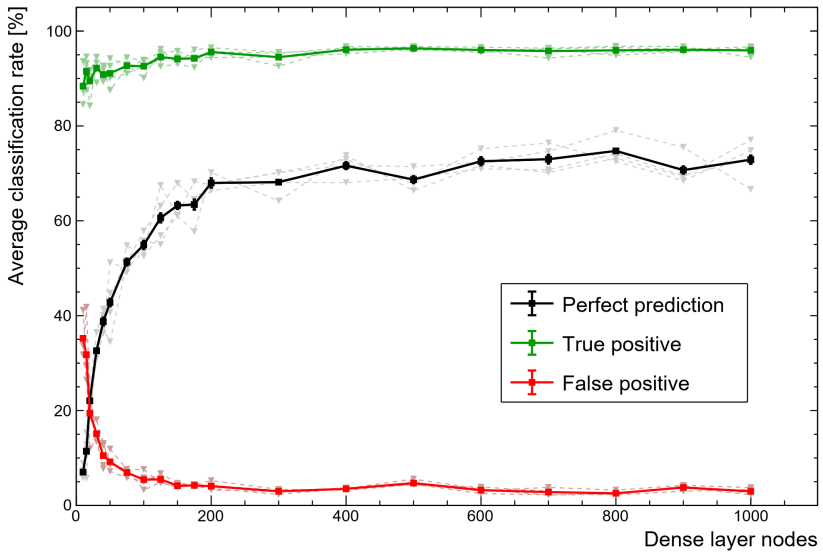



**Figure 5.** Variation in performance rates as the dense layer size of the classification module is varied. Bold lines represent the average across all training and test sets. Diminishing returns quickly encourage a compromise as dense layers rapidly increase in complexity with scale.

The 'leaky' ReLU activation function variant was implemented throughout the model to combat an issue where negative weights may a gradient of zero, leading to the well known 'dying node'

problem. A small gradient in the negative weights is therefore included as $y = max(\alpha x, x)$, where $\alpha = 0.1$, rectifying the problem of vanishing gradients.

**Table 2.** Relevant reference model hyperparameters.

| Parameter | Value |
|---|---|
| Optimiser | Adam |
| Initialisers (kernel, bias) | He normal, zeros |
| Activation function | Leaky ReLU ($\alpha = 0.1$) |
| Batch size | 32 |
| Dense layer nodes | 700 |
| Dropout (spatial, dense) | 0.5, 0.5 |
| Filters per Conv layer | 40 |
| Filter size | 7 |
| Pooling type | Max pooling (length 2, stride 2) |
| Padding | Same |
| Loss function | Binary cross-entropy |

A collection of all important hyperparameters used in the final reference model is shown in Table 2. Of course, there are many combinations and drastically different CNN structures possible. However, with the exception of dense layer size, most appropriate hyperparameter ranges showed only minor effects on classification performance during the prototyping phase of this work.

### *4.2. Reference model performance*

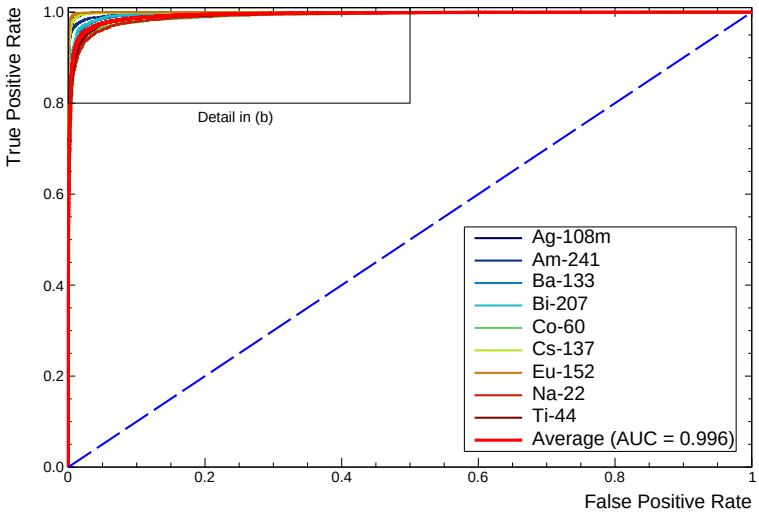
Performing evaluation on the test sets of unseen experimental data, the overall classification rates are shown in Table 3. The average perfect prediction rate across test sets was 74.4(9)%. A precision of 99.61(3)% reflects that few of the possible labels were predicted incorrectly in the remaining test spectra, indicating that correct label predictions were made with a high degree of confidence. The average time to load spectra and make predictions using the pre-trained model was <5 ms per spectrum on an i5-5200U CPU, making real-time updated classifications readily available to an end user.

**Table 3.** Overall classification rates for the reference model.

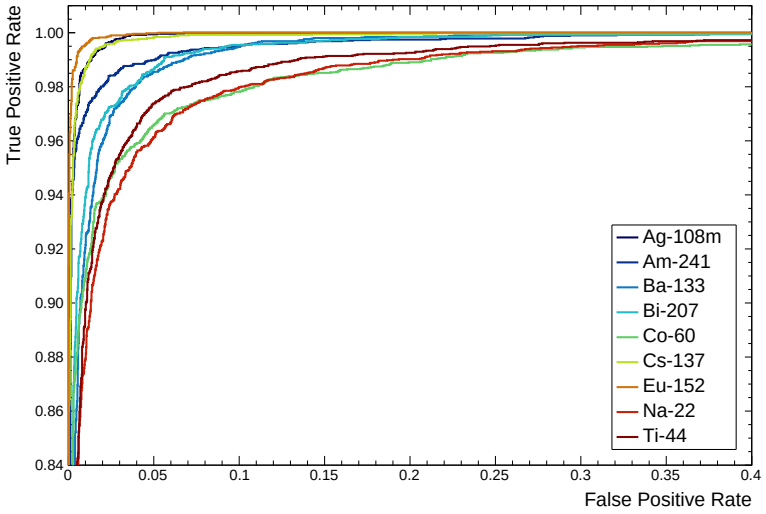
| | | Actual | |
|---|---|---|---|
| | | 1 | 0 |
| Predicted | 1 | 96.5(1) % | 3.1(2) % |
| | 0 | 3.5(1) % | 96.9(2) % |

As discussed in Section 3.2, the rates shown in Table 3 take performance as a whole. Individual binary confusion matrices can be extracted for each class, but the clearest visual aid for observing class performance is the ROC curve.

Figure 6 (a) shows a full ROC for this model after extension to the multi-label case. Each curve tends toward (0,1) as performance improves, with the dashed line representing random assignment of positive or negative classifications. Inspecting the detail in Figure 6 (b) identifies $^{22}$Na, $^{60}$Co, and $^{44}$Ti as being marginally responsible for more false classifications. Slight differences in AUC confirm this quantitatively. A $^{22}$Na spectrum in particular contains very few features, all of which closely overlap with those of many other sources to make distinction difficult. The shallow architecture found $^{207}$Bi or $^{44}$Ti slightly more challenging, without enough higher level feature extraction making it difficult to separate the two extremely similar spectra. Figure 1 previously demonstrated such a result. The $^{60}$Co source, while distinctive, has both major peaks at energies that overlap with several other sources. Again, this makes separating the sources in the overall profile much more difficult for even a trained spectroscopist. For RIID, these multi-label ROC curves prove an extremely useful tool for quickly recognising which sources different models have difficulty with.




**(a)** Full ROC with averaged curve




**(b)** Detail of ROC from *(a)* showing individual sources.

**Figure 6.** ROC curves extended to the multi-label case for the reference model. Provides comparisons of performance with each source as the decision threshold is varied. The average (red) is included in 6a as a reference for model performance across all classes. Figure 6b shows $^{22}$Na, $^{60}$Co, and $^{44}$Ti as marginally responsible for more false classifications.

### *4.3. Challenging conditions*

With knowledge of the reference model performance on unshielded sources at a 10 cm stand-off distance, new models were trained from scratch on challenging data sets using the same architecture. Results for the challenging conditions of close geometries and shielded sources are summarised in Table 4.

**Table 4.** Average perfect classification rates across data sets. All models were trained using the reference model architecture. Reference model developed on marked data set (*)

| Stand-off data | AUC | Perfect rate [%] | Shielded data | AUC | Perfect rate [%] |
|---|---|---|---|---|---|
| Point, 10 cm* | 0.996(1) | 74.4(9)* | 2 cm Al | 0.994(1) | 73.9(9) |
| Point, 5 cm | 0.995(1) | 73.9(6) | 4 cm Al | 0.986(2) | 59.9(7) |
| Point, 1 cm | 0.991(2) | 66.5(5) | 6 cm Al | 0.951(4) | 34.9(8) |

True coincidence summations become more likely in close geometries. Indeed, the solid angle of the 1 cm distance means that almost all spectra contain additional photopeaks, drastically increasing the complexity of multi-isotope spectra. Despite this, the reference model was able to perform relatively well even with the additional complexity. Note that this is in part due to the accurate handling of coincidence summations of the simulator, which produces good training data as a result. The probability for summation reduces quickly with distance, following an inverse square law, so it is unsurprising that the 5 cm and 10 cm stand-off distances performed similarly.

Attenuation of $\gamma$ rays through shielding is exponential with thickness. Lower energy photons ($<$100 keV) become increasingly more likely to interact, affecting them more severely. Backscatter, where a photon Compton scatters in material prior to entering the detector, also changes the profile of spectra significantly. A combination of these effects makes shielded scenarios one of the greatest challenges of RIID.

Heavily shielded conditions were simulated to reproduce experimental test data. Sources were placed 10 cm from the detector, with up to 6 cm of aluminium positioned between the two. Classification rates suffer as shielding is increased. In contrast to other data sets, the ROC curves reveal $^{44}$Ti, $^{241}$Am and $^{152}$Eu to be the cause of more incorrect predictions. Interestingly it can be inferred that the $^{152}$Eu predictions rely quite strongly on the more prominent, low-energy peaks for distinguishing the source.

The $^{60}$Co and $^{22}$Na spectra, each with prominent features at higher energies, have been replaced by those that rely on low-energy features. Of course, these are now attenuated, often to the point of being missing. At 6 cm there is simply not enough of the solitary $^{241}$Am peak left to stand out among backscatter and other sources, with the average false negative rate of 12.3(3)% almost entirely due to this source.

While perfect rates drop below 50%, it is important to note that the perfect rate for randomly assigning a label as present is just 0.2% for this set. It is very encouraging that model performance reflects the challenges of the physical phenomena involved, enabling a more intuitive understanding of model limitations and how pre-processing may be adapted to highlight certain features.

### *4.4. Optimising for deployment*

The CNN approach has demonstrated excellent performance for the challenges addressed, but the focus consistently lay with changes in the model architecture. Pragmatic solutions for real-world deployment can not ignore the data sets themselves. The key theme is always one of representation. Application of common tactics for improving performance beyond hyperparameters were investigated.

#### 4.4.1. Data set size

One of the simplest changes to make is in the size of the training data set. For rapid prototyping, a small set of $\sim$20 samples per unique source combination was used to optimise workflow. This reduces to an average of just 10 samples after splitting spectra into training and validation sets. With the extremes in gain shifts, total counts, and relative activity of the mixed sources, this makes for a data set that can not possibly represent the diverse range of spectra that may be present in experimental test sets. As a brief experiment, the same parameters used to generate the unshielded data set of the reference model were applied here.

As Figure 7 shows, changing nothing but the size of the training set sees PPR performance increase to $\sim$81%. The advantages of a larger set are seen to plateau here because the training data variations are all well represented. Of course, the limitation here is that training data are entirely simulated, missing some of the the subtleties of real spectra. However, this general trend does demonstrate why augmenting particularly small training sets with simulated data is very often a worthwhile endeavour. The next section explores the inclusion of real spectra as a more realistic use of the data likely to be available when deploying systems.

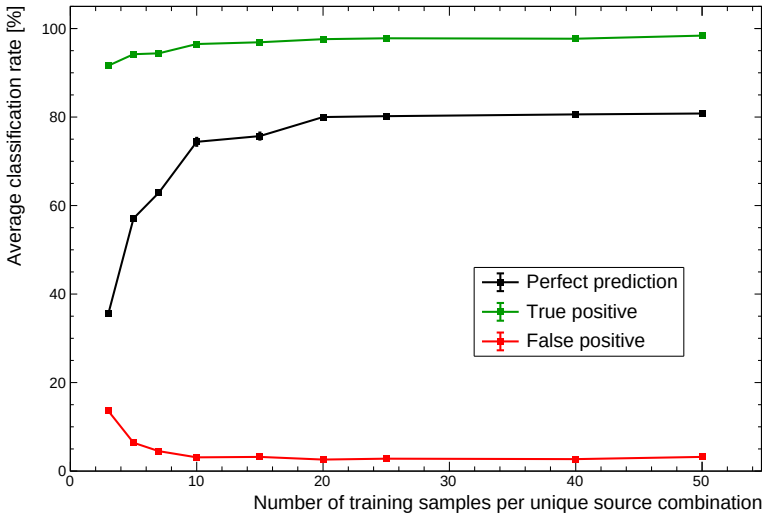



**Figure 7.** Comparison of performance with number of training samples. While there is a sharp rise in performance as more training spectra are added, the performance plateaus for these particular data. For more complex data sets, additional training samples are expected to become far more important.

#### 4.4.2. Including real data

Simulated data are only ever an approximation of reality. The generalised approach using only simulated training data provided a rapid development environment, but they can exclude the subtleties of real spectra. Ideally all training data would be experimentally obtained, but a compromise often made is to supplement the available on-site data with those of a simulation.

New training sets were generated to contain an even mix of real and simulated point source data at 10 cm, averaging 10 samples per source combination for direct comparison with the reference model. The PPR averaged an excellent 89.8(9)%, with ROC curves showing that the $^{44}$Ti, $^{22}$Na, and $^{60}$Co sources were, once again, the cause of more incorrect predictions. Including as many real data as possible is therefore highly recommended, whatever the application, and these results clearly demonstrate the advantages of doing so for the purposes of RIID.

#### 4.4.3. Generalisation

These models have already been shown to perform relatively well at different stand-off distances and levels of shielding. In each case, the model was trained on a specific simulated data set, and tested on the experimental version. However, many data sets may be combined to make the single, generalised model that real world applications need. A large data set was made by combining every individual data set, each with 10 training samples per source combination. For the sake of comparison, the training data were entirely simulated. The reference model was then trained on this all-inclusive set. This generalised model was finally used to make predictions on all of the test sets separately for direct comparison with the performance of previous sections (Table 5).

**Table 5.** Average perfect classification rates for all scenarios. The generalised model is compared with those trained individually on each set. Reference model developed on marked data set (*).

| Test set | Individual (%) | Generalised (%) |
|---|---|---|
| 1 cm stand-off | 66.5(5) | 72.6(9) |
| 5 cm stand-off | 73.9(6) | 90.5(4) |
| 10 cm stand-off* | 74.4(9)* | 90.9(4) |
| 1 Al plate | 73.9(9) | 88.5(7) |
| 2 Al plates | 59.9(7) | 81.5(6) |
| 3 Al plates | 34.9(8) | 60.4(7) |

The results across all sets are much improved, even exceeding the addition of real spectra for the 10 cm stand-off data set. This follows expectations, given that the training data here are 6× larger, and far more comprehensive. Challenging conditions still make heavy shielding and close geometries more difficult, mirroring the trends seen with the more focused models trained on individual data sets. However, there is clearly scope for training a generalised model to handle spectra produced under all kinds of external challenges, on top of complications present in every spectrum as a result of physical detection mechanisms.

## 5. Conclusions

This work demonstrates that even basic CNNs may be used for the rapid, accurate identification of multi-isotope sources. The focus of this work was the methodology and performance of a simple model when applied to challenging data sets, even where experimental data are not readily available. A sequential CNN was applied to the RIID problem space to identify arbitrary mixtures of up to nine nuclides. Precision and the AUC metric are often too close for meaningful comparisons, so taking the full set of possible sources, 'perfect' classification rates were used as a complementary performance metric. An average perfect rate of ∼74% was seen for uncalibrated experimental spectra that contained large gain shifts, low energy thresholds, often poor statistics, and were trained exclusively on a small sample of simulated data.

Applying a consistent model architecture to the conditions that most challenge RIID showed comparable performance in most cases. In the extremes it was seen that predictions faltered for heavily shielded isotopes, primarily those with low energy emissions due to a loss of critical features in the spectra. In close geometries, multi-isotope spectra become far more complex due to extra summation between photon emissions. Despite this, models tested on the 1 cm stand-off data still performed well.

Inclusion of many additional training spectra raised perfect prediction rates to ∼81% in a trend typical of deep learning problems. The variation in training set characteristics demonstrates a strong influence on performance, so evaluation of models for RIID should not be confined to just the CNN architecture and its hyperparameters. Indeed, most real-world applications require representation of many different scenarios to account for geometries, crystal sizes, intrinsic resolutions, and any other physical effect that will distort the profile of a spectrum. The importance of representation is highlighted when real spectra are mixed into the pool of training samples. The perfect prediction rate for these same test sets used for the reference model was increased by ∼15%, a trend that the work of Ref. [19] also notes for a similar application. Lastly, this work also indicates that CNN-based models lend themselves particularly well to a generalised RIID model that can accommodate a range of conditions expected in the deployment of real systems.

To summarise, this work shows that relatively simple CNN models can be successfully applied to pragmatic nuclide identification. A constantly evolving landscape of deep learning techniques often produces new and advanced convolution architectures. These, combined with the domain knowledge necessary for effective pre-processing in gamma spectrometry, may prove ideal for rapid and robust real-time CNN-based RIID systems.

## 6. Future work

Current efforts seek to combine further data sets for the purpose of training a general purpose model. The preliminary results presented demonstrate the potential for such a model. Fortunately, the techniques outlined in this work are extremely flexible, allowing expansion of training sets to combinations of any nuclides and conditions that may be simulated. However, it must be noted that scalability of a single model in this way is limited. Future work therefore also seeks to compare performance to a modular ensemble network. Instead of a single model, there would be a combination of nine separate CNNs performing a binary classification for each source, the results of which are brought together later similar to the approach of Ref. [10].

**Author Contributions:** conceptualization, A.T, C.W, Tz.KW, M.G, L.P, J.B, and M.F; methodology, A.T; software, A.T; validation, A.T; formal analysis, A.T; investigation, A.T; resources, A.T; data curation, A.T; writing–original draft preparation, A.T; writing–review and editing, A.T, C.W, Tz.KW, M.G, L.P, J.B; visualization, A.T; supervision, C.W, Tz.KW, M.G, L.P, J.B, and M.F; project administration, C.W, L.P, and J.B; funding acquisition, C.W, L.P, J.B, and M.F

**Funding:** This research is funded in equal part by the University of Birmingham and AWE plc.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

| | |
|---|---|
| RIID | Radio-Isotope IDentification |
| CNN | Convolutional Neural Network |
| ANN | Artificial Neural Network |
| ROC | Receiver Operator Characteristic |
| TPR | True Positive Rate |
| FPR | False Positive Rate |
| PPR | Perfect Performance Rate |
| AUC | Area Under Curve |

## References

## References

1. Olmos *et al.*, P. A new approach to automatic radiation spectrum analysis. *IEEE Trans. Nucl. Sci.* **1991**, *38*, 971–975. doi:10.1109/23.83860.
2. Vigneron, V.; Morel, J.; Lèpy, M.; Martinez, J. Statistical modelling of neural networks in $\gamma$-spectrometry. *Nucl. Instrum. Methods Phys. Res. A* **1996**, *369*, 642–647. doi:10.1016/S0168-9002(96)80068-4.
3. Pilato, V.; Tola, F.; Martinez, J.; Huver, M. Application of neural networks to quantitative spectrometry analysis. *Nucl. Instrum. Methods Phys. Res. A* **1999**, *422*, 423–427. doi:10.1016/S0168-9002(98)01110-3.
4. Chen, L.; Wei, Y. Nuclide identification algorithm based on K–L transform and neural networks. *Nucl. Instrum. Methods Phys. Res. A* **2009**, *598*, 450–453. doi:10.1016/j.nima.2008.09.035.
5. Yoshida, E.; Shizuma, K.; Endo, S.; Oka, T. Application of neural networks for the analysis of gamma-ray spectra measured with a Ge spectrometer. *Nucl. Instrum. Methods Phys. Res. A* **2002**, *484*, 557–563. doi:10.1016/S0168-9002(01)01962-3.
6. Kamuda, M.; Stinnett, J.; Sullivan, C. Automated Isotope Identification Algorithm Using Artificial Neural Networks. *IEEE Trans. Nucl. Sci.* **2017**, *64*, 1858–1864. doi:10.1109/TNS.2017.2693152.
7. Vargas, A.; Camp, A.; Serrano, I.; Duch, M. Development of a neural network approach to characterise $^{226}$Ra contamination at legacy sites using gamma-ray spectra taken from boreholes. *J. Environ. Radioact.* **2015**, *140*, 130–140. doi:10.1016/j.jenvrad.2014.11.011.
8. Kamuda, M.; Zhao, J.; Huff, K. A comparison of machine learning methods for automated gamma-ray spectroscopy. *Nucl. Instrum. Methods Phys. Res. A* **2018**, *64*, 1858–1864. doi:10.1016/j.nima.2018.10.063.
9. Liang *et al.*, D. Rapid nuclide identification algorithm based on convolutional neural network. *Semin. Nucl. Med.* **2019**, *133*, 483–490. doi:10.1016/j.anucene.2019.05.051.
10. Daniel, G.; Ceraudo, F.; Limousin, O.; Maier, D.; Meuris, A. Automatic and Real-time Identification of Radionuclides in Gamma-ray Spectra: A new method based on Convolutional Neural Network trained with synthetic data set. *IEEE Trans. Nucl. Sci.* **2020**, p. Early Access. doi:10.1109/TNS.2020.2969703.
11. Agostinelli *et al.*, S. Geant4 - a simulation toolkit. *Nucl. Instrum. Methods Phys. Res. A* **2003**, *506*, 250–303. doi:10.1016/S0168-9002(03)01368-8.

12. Turner, A.; Wheldon, C.; Gilbert, M.; Packer, L.; Burns, J.; Kokalova, T.; Freer, M. Generalised gamma spectrometry simulator for problems in nuclide identification. *J. Phys.: Conf. Ser.* **2019**, *1643*. 27th International Nuclear Physics Conference (INPC2019), doi:10.1088/1742-6596/1643/1/012211.
13. Brun, R.; Rademakers, F. ROOT - An object oriented data analysis framework. *Nucl. Instrum. Methods Phys. Res. A* **1997**, *389*, 81–86. doi:10.1016/S0168-9002(97)00048-X.
14. Abadi *et al.*, M. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. *Online resource, arXiv* **2015**.
15. Kingma, D.; Ba, J. Adam: A method for stochastic optimization. *arXiv: 1412.6980* **2014**, p. Preprint.
16. Bishop, C. *Pattern recognition and machine learning*; Springer: Cambridge, UK, 2006. p.209.
17. Hinton, G.; Srivastava, N.; Swersky, K. Lecture 6d - a separate, adaptive learning rate for each connection. *Slides of Lecture Neural Networks for Machine Learning* **2012**.
18. He, K.; Zhang, X.; Ren, S.; Sun, J. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. *ICCV* **2015**, pp. 1026–1034.
19. Kim, J.; Park, K.; Cho, G. Multi-radioisotope identification algorithm using an artificial neural network for plastic gamma spectra. *Applied Radiation and Isotopes* **2019**, *147*, 83–90. doi:10.1016/j.apradiso.2019.01.005.