

Model-Based Deep Learning Algorithm for Pulse Shape Discrimination in High Event Rates

Itai Morad^{1,2,*}, Max Ghelman^{1,3}, Dmitry Ginzburg⁴, Alon Osovizky^{3,4}, and Nir Shlezinger¹

¹Ben-Gurion University of the Negev,

²Israel Atomic Energy Commission (IAEC),

³Nuclear Research Center Negev (NRCN),

⁴Radiation Detection Department, Rotem Industries Ltd

(*) itaimor@post.bgu.ac.il

Abstract—Pulse shape discrimination (PSD) is at the core of radioactive particles monitoring. Conventional PSD methods are geared towards low event rates, and struggle in the presence of pileups resulting from high rate. In this work we develop a PSD algorithm that combines classic approaches with deep learning techniques, that is highly suitable for coping with the dramatic challenges associated with classifying pulses in high event rates. Common PSD algorithms for high event rates limit their research to two piled-up pulses. Our algorithm is designed and tested under severe pileup conditions, where three or more pulses were piled-up. We tested the algorithm on simulated data based on Cs₂LiYCl₆:Ce (CLYC) based detector pulse shapes and compare its performance to both traditional PSD algorithms and data-driven deep neural network (DNN) based algorithms. In high event rates, ranging up to 10 Mcps, the algorithm demonstrates up to 8 times fewer miss-classifications than the traditional normalized cross-correlation (NCC) approach, and up to 1.7 times fewer miss-classifications than a purely data-driven DNN-aided method.

Keywords—Neutron, gamma ray, pulse shape discrimination, pulse pileup, deep learning, digital signal processing, high rate.

I. INTRODUCTION

THE process of radioactive decay may result in the emission of various radioactive particles such as neutrons, gamma rays, alphas, and betas. Some detectors have the ability to detect multiple types of radiation. Different particles interacting with these detectors produce different pulse shapes, and can therefor be distinguished using pulse shape discrimination (PSD). It is often important to be able to distinguish these different types of radiation for homeland security, nuclear safeguard, and neutron scattering applications. In some cases, these applications require accurate counting of multiple radiations in high event rates [1]–[4]. Operation within these high event rates often causes severe pileup conditions, notably distorting the shape of the pulses, in contrast to operation within low event rates [1], [3], [5]–[8]. It is therefor particularly important to design dedicated algorithms able to distinguish radioactive particles arriving at high event rates.

Various PSD techniques were proposed in the literature. Traditional digital signal processing PSD methods include charge integration [9], slice-fitting algorithms [10], histogram-difference methods [11], and discrete wavelet transforms [12], [13]. PSD methods based on traditional machine learning techniques include Gaussian mixture models [14], support

vector machines [15], and k-means [16]. Finally, many authors have proposed PSD methods based on deep neural networks (DNNs) [4]–[6], [17]–[20]. Even though many approaches to PSD have been proposed over the years, the vast majority is designed to discriminate particles at low event rates. Therefor, the discrimination capabilities of these approaches degrade when discriminating highly distorted pulse shapes. PSD approaches geared towards pileup conditions often limit their scope to cases where at most two pulses are piled-up, discarding severe conditions with three or more piled-up pulses [3]–[6], [21].

As detailed above, current approaches to PSD in high event rates can be generally divided into model-based methods and data-driven methods. Model-based methods (based on signal processing techniques) are often designed for specific pileup scenarios and may struggle to discriminate highly distorted pulse shapes. Purely data-driven methods (based on machine learning and deep learning tools) on the other hand lack knowledge about the underlying physics of the detector, missing out on important information for discrimination.

In this work, we present a PSD algorithm combining classic PSD approaches with deep learning techniques following model-based deep learning methodology [22]–[24]. By doing so we design a scheme that is both highly suitable for coping with the dramatic challenges associated with classifying pulses in high event rates while preserving the interpretability and generalization capabilities of classic PSD methods.

To derive our scheme, we first simulate the detector signal. Although the data used in this work is based on neutron and gamma ray interactions with a Cs₂LiYCl₆:Ce (CLYC) detector, the designed scheme can be generalized to other detectors and radiations. Next, we design our PSD algorithm intended for operation in high event rates by discriminating based on both classic features known to be useful in low rates and the raw signal, which are balanced based on features that are informative of the pileup conditions. The latter enables operation in both low rates and in high rates using the same model. Our experimental study focused on simulating the detector signal under various neutron and gamma rate conditions. We show that our proposed method successfully classifies the arriving pulses, and demonstrate its effectiveness compared to competing methods.

The rest of this article is organized as follows. section II

presents the system model and problem formulation. section III details the proposed method and the training procedure. In section IV, the proposed method is experimentally evaluated. The work is concluded in section V.

II. SYSTEM MODEL

A. Problem Formulation

We consider the PSD task realized on the digitized output of the detector. We focus our work on neutrons and gamma rays interacting with the detector. Therefore, the label space $\mathcal{S} = \{0, 1\}$ consists of two elements, '0' for neutrons and '1' for gamma rays. The raw signal produced by the detector following the arrival of the k th particle, either a neutron or a gamma ray, is defined as \mathbf{x}_k , and its label is denoted by s_k . For some classification algorithm $g(\cdot)$ we define the algorithm's total accuracy, neutron accuracy, and gamma accuracy tested on K arriving particles in (1), (2), and (3) respectively.

Accordingly, by letting $\mathbb{1}_{\mathcal{A}}$ denote the indicator function over event \mathcal{A} , our main performance measures are the total discrimination accuracy, defined as

$$\text{Total accuracy}(g) = 100 \cdot \frac{\sum_{k=1}^K \mathbb{1}_{s=s_k} g(\mathbf{x}_k)}{K}, \quad (1)$$

while the particle-specific neutron and gamma accuracies are respectively defined as

$$\text{Neutron accuracy}(g) = 100 \cdot \frac{\sum_{k=1}^K \mathbb{1}_{s=s_k, s_k=0} g(\mathbf{x}_k)}{\sum_{k=1}^K \mathbb{1}_{s_k=0}}, \quad (2)$$

and

$$\text{Gamma accuracy}(g) = 100 \cdot \frac{\sum_{k=1}^K \mathbb{1}_{s=s_k, s_k=1} g(\mathbf{x}_k)}{\sum_{k=1}^K \mathbb{1}_{s_k=1}}. \quad (3)$$

B. Data

To design our algorithm, the detector signals used in this work are based on simulation. In order to simulate the pulse shapes, we leveraged the established analysis of the pulse shape characteristics of neutron and gamma ray interactions with CLYC scintillators coupled to a photomultiplier tube (PMT) [5], [25]–[30]. While shape characteristics differ due to the usage of different size and shape scintillators, different PMTs, and varying environmental conditions, the analyzing and modeling the pulse shapes allows for the creation of simulated pulses which preserve the characteristics of the experimental pulses.

Specifically, the simulated data used in our work is based on the characterization proposed in [5], with the specific configuration presented in [5, Eqn. (1), Tables 2-3]. The detector signal was simulated in the following manner: First, the arrival times of the neutrons and gammas were first drawn from two independent Poisson point processes; Next, the detector signal was created by summing simulated neutron pulses at the neutron arrival times, simulated gamma pulses at the gamma arrival times, and additive white Gaussian noise (AWGN). Simulating the detector signal under severe pileup conditions using the proposed additive model resembles a real detector signal under severe pileup conditions because the

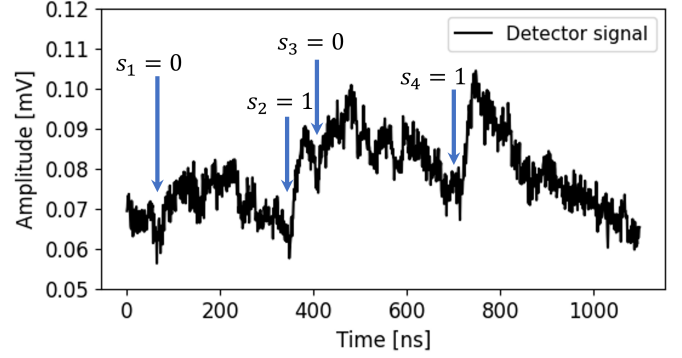


Fig. 1. A simulated detector signal under severe pileup conditions. Arriving particles and their corresponding label are marked with blue arrows. The simulated event rate is 2 Mcps, and pulse shapes were simulated based on the characterization proposed in [5], with the specific configuration presented in [5, Eqn. (1), Tables 2-3].

anode current of the PMT is proportional to the magnitude of the incident radiation flux, which would be linearly added when there are multiple pulses arriving within a short time window of tens of nanoseconds [4], [31]. A simulated detector signal under severe pileup can be seen in Fig. 1.

C. Effects of Pileup

We focus on the task of performing PSD when operating within high event rates, which is particularly challenging due to pulse pileup. Pileup occurs when multiple pulses arrive within a short period of time. The resulting overlap highly distorts the shape of the pulse, as illustrated in Fig. 1, making it challenging to discriminate. The main factors dictating the severity of pileup conditions are the pulse shapes, lengths, and arrival rate giving rise to the following challenges:

Challenge I: Mathematical modeling. PSD is usually performed based on extracted features, e.g., rise time, amplitude, and decay constants, that represent the pulses based on the mathematical description that makes them useful for discrimination. For non-piled-up pulses, the uncertainty depends solely on the signal-to-noise ratio (SNR) and the accurate representation of the underlying physics of the detector. However, extracting features of a piled-up pulse requires additional mathematical manipulations. While in cases where two pulses are piled-up, feature extraction can still be relatively accurate, in moderate and high pileup conditions where three or more pulses are piled-up onto each other, accurate feature extraction becomes extremely challenging.

Challenge II: Temporal correlation. A time window containing a non-piled-up pulse with AWGN is statistically independent of all other time windows. A time window containing a piled-up pulse on the other hand only contains fractions of the pulses which are piled-up onto the one that is being observed. The rest of these pulses lie within consecutive time windows, resulting in statistical dependencies between these time windows.

Challenge III: Interference. In cases where pulses are not piled-up onto each other, PSD capabilities highly depend on the SNR. Low energy pulses are usually harder to discriminate

due to noise interference. Pileup causes additional interference caused by overlapping pulses. In severe pileup conditions, distortions caused by overlapping pulse interference are much more dominant than the distortions introduced by noise.

The aforementioned challenges motivate deriving a PSD method that both exploits the available useful features and expected temporal dependence, while leveraging the abstractness of DNNs to cope with the challenging modelling and interference, as detailed in the following section.

III. METHOD

In this section, we introduce our PSD algorithm that follows model-based deep learning methodology [22]–[24]. The proposed method offers a hybrid technique, that combines knowledge about the underlying physics of the detector with a data-driven DNN to benefit from the advantages of both approaches. As our design is geared towards high rates, features providing relevant information in severe pileup conditions are used.

The proposed DNN-based method utilizes the raw pulse, classical features, and weighing features for classification, based on the following rationale: The raw pulse samples and classical features, which characterize the pulse, are fed into a DNN which classifies the pulse; The weighing features characterize the pileup conditions and are used to learn a weighing vector that weighs out which parts of the raw pulse and classical features hold the most important information for classification in a specific case. We next elaborate on these features and their processing procedure.

1) *Classical Features*: We use the term *classical features* to refer to features extracted from the raw pulses that are conventionally employed for classification in standard low-rate settings. The classical features used in this paper are only a handful of the overall features that can be extracted to represent these pulses [10]–[12]. Although **Challenge I** makes feature extraction difficult under pileup conditions, some feature extraction methods are less prone to error than others.

In order to cope with **Challenge I**, we analyzed various features in order to understand which of them preserve their information gain under pileup conditions. The key difference between general classical features and the classical features used in our method is the preservation of information in severe pileup rates. For example, estimating the decay constants of highly piled up pulses is much more difficult than estimating their rise time. In particular, for classifying an arrival at time t , we use the vector \mathbf{y}_t comprised of the following classical features:

- Y1 Rise time.
- Y2 Correlation coefficients: computing the correlation between the pulse with a neutron template, and the correlation between the pulse and a gamma template.
- Y3 Amplitude.
- Y4 Baseline: computing the baseline at the moment of arrival.
- Y5 Energy: calculating the integral in different sections of the pulse.

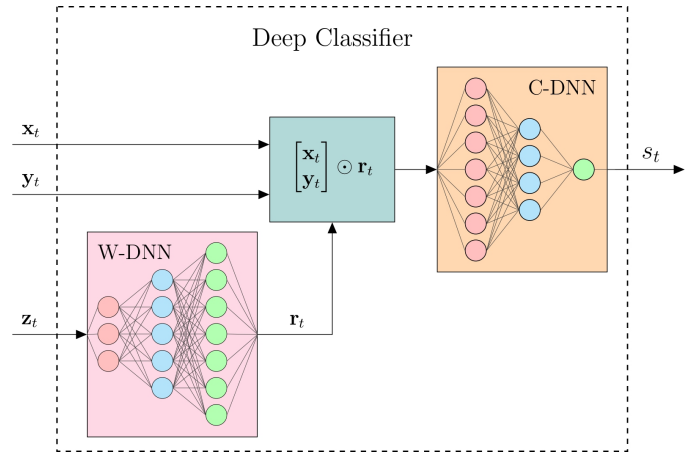


Fig. 2. Architecture illustration of the Deep Classifier.

2) *Weighing Features*: We use the term *weighing features* to denote features that are informative of the current pileup conditions. The rationale is to use such information to balance the reliability of the classical features, that are known to be useful in the absence of pileup.

Specifically, **Challenge II** and **Challenge III** introduce interference which is dependent both on the classifier pulse and consecutive time windows. Therefore, in order to cope with **Challenge II** and **Challenge III**, the weighing features for classifying an arrival at time t , denoted \mathbf{z}_t are fed into a DNN which outputs a vector that is used to weight the classical features, denoted \mathbf{r}_t . Essentially, a mapping between the pileup conditions and the importance of the raw pulse and classical features under those conditions is learned. The weighing features vector \mathbf{z}_t is comprised of:

- Z1 Distance from the pulse detected previously.
- Z2 Distance from the pulse detected next.
- Z3 The ratio between the energies of the current pulse and the pulse detected previously.
- Z4 The ratio between the energies of the current pulse and the pulse detected next.
- Z5 The ratio between the energy of the current pulse and the baseline at the time of arrival.

3) *Architecture*: The architecture of the Deep Classifier, illustrated in Fig. 2, consists of two DNNs, each being a multi-layered perceptron. The weighing features \mathbf{z}_t are fed into the weighing deep neural network (W-DNN) which outputs a weighing vector \mathbf{r}_t . The resulting vector then weights the stacking of the raw pulse \mathbf{x}_t and the classical features \mathbf{y}_t via Hadamard product, which in turn is fed into the classifying deep neural network (C-DNN). The C-DNN has a single output neuron with sigmoid activation, such that the output of the Deep Classifier with trainable parameters θ , raw pulse \mathbf{x}_t , classical features \mathbf{y}_t , and weighing features \mathbf{z}_t , is $f_\theta(\mathbf{x}_t, \mathbf{y}_t, \mathbf{z}_t) \in [0, 1]$. A threshold is set on the output neuron, values above it are classified as neutrons, and values under it are classified as gammas. The PSD procedure of the Deep Classifier is summarized as Algorithm 1.

The Deep Classifier is using a dataset \mathcal{D} comprised of tuples of $\mathbf{x}, \mathbf{y}, \mathbf{z}$ and their corresponding label $s \in \{0, 1\}$,

Algorithm 1: Deep Classification

Init: θ ; Threshold THR

Input : Raw pulse x_t

- 1 Calculate classical features Y1-Y5 as y_t ;
 - 2 Calculate weighing features Z1-Z5 as z_t ;
 - 3 Apply Deep Classifier $\hat{s}_t = f_{\theta}(x_t, y_t, z_t)$;
 - 4 **if** $\hat{s}_t > THR$ **then**
 - └ **Output:** Neutron at t
 - 5 **else**
 - └ **Output:** Gamma at t
-

representing gamma and neutron, respectively. Training is done in two stages. In the first stage, the weighing vector is not computed and only the C-DNN is trained to map the raw pulse and classical features to the output. After a basic mapping between the raw pulse and classical features to the output has been learned, the second stage begins. In this second stage, the C-DNN and the W-DNN are jointly trained, boosting the performance of the Deep Classifier by weighing the data used for classification based on the pileup conditions. The binary cross-entropy (BCE) loss, which is arguably the most common loss used for binary classification tasks, is used for training, resulting in the following loss measure:

$$\mathcal{L}_{\mathcal{D}}(\theta) = \frac{1}{|\mathcal{D}|} \sum_{(x_l, y_l, z_l, s_l) \in \mathcal{D}} s_l \cdot \log(f_{\theta}(x_l, y_l, z_l)) + (1 - s_l) \cdot \log(1 - f_{\theta}(x_l, y_l, z_l)). \quad (4)$$

The overall two-stage training algorithm based on conventional mini-batch stochastic gradient descent is summarized as Algorithm 2.

Algorithm 2: Training the Deep Classifier

Init: Set θ randomly

Fix learning rate $\mu > 0$ and epochs i_{mid}, i_{max}

Input: Training set \mathcal{D}

- 1 **for** $i = 0, 1, \dots, i_{mid} - 1$ **do**
 - 2 Fix $r_t = 1$;
 - 3 Randomly divide \mathcal{D} into Q batches $\{\mathcal{D}_q\}_{q=1}^Q$;
 - 4 **for** $q = 1, \dots, Q$ **do**
 - 5 Apply model with parameters θ to \mathcal{D}_q ;
 - 6 Compute batch loss $\mathcal{L}_{\mathcal{D}_q}(\theta)$ by (4);
 - 7 Update $\theta \leftarrow \theta - \mu \nabla_{\theta} \mathcal{L}_{\mathcal{D}_q}(\theta)$;
 - 8 **for** $i = i_{mid}, i_{mid} + 1, \dots, i_{max} - 1$ **do**
 - 9 Randomly divide \mathcal{D} into Q batches $\{\mathcal{D}_q\}_{q=1}^Q$;
 - 10 **for** $q = 1, \dots, Q$ **do**
 - 11 Apply model with parameters θ to \mathcal{D}_q ;
 - 12 Compute batch loss $\mathcal{L}_{\mathcal{D}_q}(\theta)$ by (4);
 - 13 Update $\theta \leftarrow \theta - \mu \nabla_{\theta} \mathcal{L}_{\mathcal{D}_q}(\theta)$;
 - 14 **return** θ
-

4) *Discussion:* The usage of dedicated features to extract weighing coefficients is inspired by the emerging technique of context-aware modulation [32], which realizes a form of

hypernetworks whose mapping is not fixed but is affected by additional features [33]. Accordingly, we identify that the classification mapping and how it balances the classical features y_t and the pulse data should vary between different pileup conditions. Hence the incorporation of the weighing features, that are selected due to their being informative regarding pileups. The validity and usefulness of our approach are numerically demonstrated in section IV.

IV. RESULTS

The proposed Deep Classifier alongside four other PSD algorithms were tested under various event rates. The event rate at which pileup becomes severe is dependent on the length of the pulses produced by the detector. Therefore, the event rates under which the proposed Deep Classifier was tested were adjusted to the simulated CLYC detector data used in this work. For the CLYC detector, high event rates where PSD becomes challenging are considered to be in the range of several Mcps [1], [3], [5]–[8]. The data set \mathcal{D} used for testing consisted of 50,000 simulated neutrons and 50,000 simulated gammas.

Firstly, the proposed Deep Classifier is compared to two classic PSD algorithms, e.g., the charge integration (CI) method and the normalized cross-correlation (NCC) method. The parameters of both methods were optimized for the simulated event rates and the algorithms were slightly altered to better fit operation in high event rates, using operations such as baseline subtraction to boost performance. The CI based algorithm was realized using a short and a long integration gate, while the NCC algorithm was realized by computing two templates from the neutron and gamma distributions respectively.

In addition, the proposed method is compared to two DNN based methods. The first method is a DNN applied to the raw pulse, which is essentially x entering the C-DNN. The second method is a DNN applied to the stacking of the raw pulse and the classical features, e.g., Y1-Y5, which is essentially x and y entering the C-DNN. Comparing the Deep Classifier to a DNN applied to the raw pulse allows us to validate the effectiveness of our method compared to a purely data-driven DNN based method which is widely used. Moreover, comparing the Deep Classifier to both methods allows us to demonstrate and validate the contribution of the different parts of our method. Comparison between the DNN applied to the raw pulse and the DNN applied to the stacking of the raw pulse and the classical features demonstrates the contribution of the classical features and their validity in high event rates. Comparison between the DNN applied to the stacking of the raw pulse and the classical features to our proposed method demonstrates the contribution of the weighing features and the boost in performance achieved by the mapping between the pileup conditions and the importance of the raw pulse and classical features.

The first simulation was carried out as follows: The detector's signal was simulated as detailed in Subsection II-B. The neutrons and gammas arrived at identical count rates, making the total event rate vary between 500 kcps and 10 Mcps. The

total accuracy of the methods defined in (1) was calculated for each event rate. The results of the proposed Deep Classifier and the four benchmarks in the first simulation are displayed in Fig. 3.

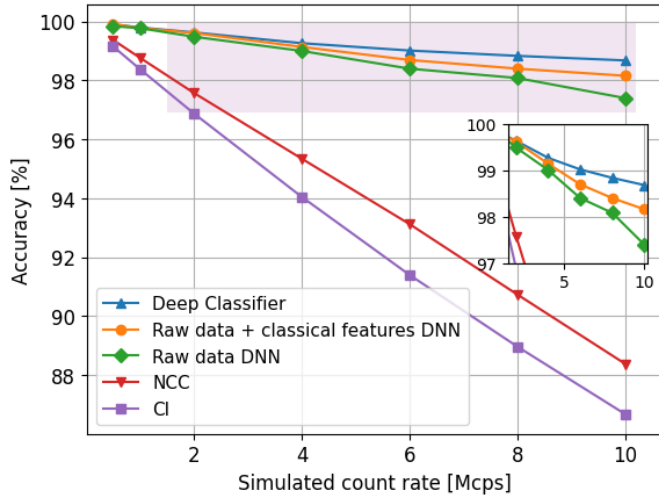


Fig. 3. The total accuracy achieved by the Deep Classifier and the four benchmark algorithms at event rates ranging between 500 kcps and 10 Mcps.

Observing Fig. 3, we note that for the lowest event rate displayed, 500 kcps, all five methods achieve a high accuracy of over 99%. The accuracies of the classic methods are slightly lower than the accuracies of the DNN based methods. As the event rate goes higher, the accuracy of all five methods becomes smaller, as expected. A clear trend can be seen where the accuracy of the classic methods falls dramatically behind the accuracy of the DNN based methods. The NCC based method achieves slightly better results than the CI based method. This result shows that data-driven DNN based methods may be more suitable than classical NCC and CI based methods for PSD in high event rates.

The zoomed-in area highlighted in light purple in Fig. 3 provides a better view of the accuracy achieved by the DNN based methods. It can be seen that all DNN based methods achieve similar accuracies at event rates of 1 Mcps and under. As the event rate goes higher, a clear trend can be seen where the lowest accuracy is achieved by the DNN applied to the raw pulse, followed by the DNN applied to the stacking of the raw pulse and the classical features, and finally, the best accuracy is achieved by the proposed Deep Classifier. These results validate the effectiveness of our method at high event rates and demonstrate the contribution of both the classical features y and the weighing features z .

For most applications, the neutron and the gamma fluxes are not the same. In particular, the gamma flux is often larger than the neutron flux. The following simulation was conducted to assess and characterize the accuracy of DNN based methods across various ratios between the gamma and neutron flux. The accuracies of the methods were evaluated under ratios ranging from 1:1 up to 25:1 between the gamma flux and the neutron flux respectively. In all cases, the total event rate was 4 Mcps. The resulting total accuracies defined in (1) are displayed in Fig. 4. The neutron accuracies defined in (2) are displayed in

Fig. 5 and the gamma accuracies defined in (3) are displayed in Fig. 6.

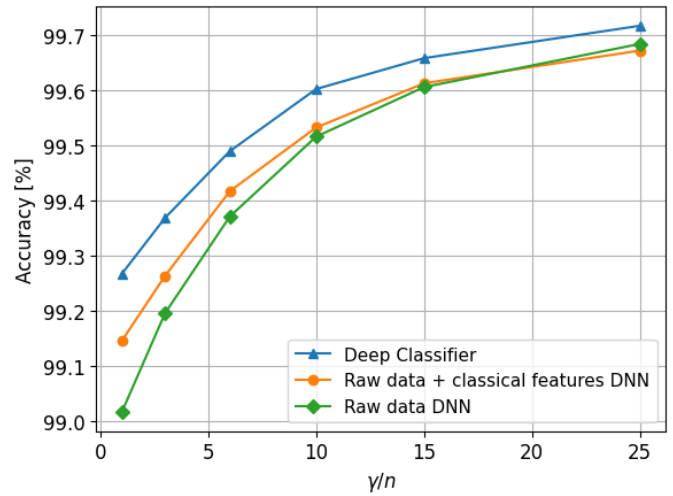


Fig. 4. The total accuracy achieved by the Deep Classifier and the two DNN based benchmark algorithms for ratios between the gamma and the neutron flux ranging from 1:1 up to 25:1 respectively. The total event rate for all ratios is 4 Mcps.

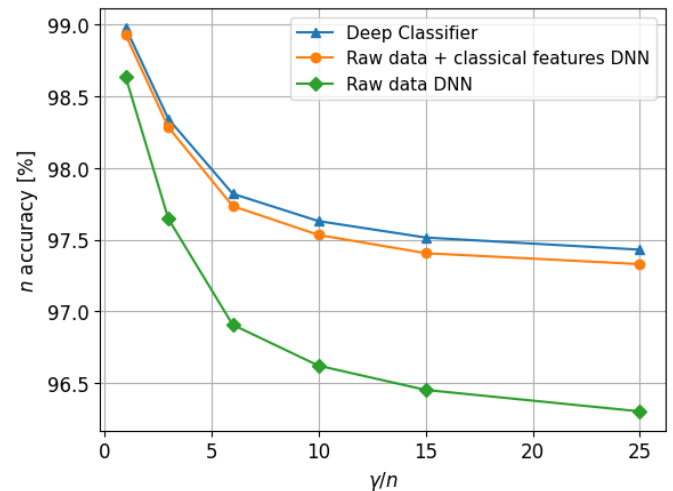


Fig. 5. The neutron accuracy achieved by the Deep Classifier and the two DNN based benchmark algorithms for ratios between the gamma and the neutron flux ranging from 1:1 up to 25:1 respectively. The total event rate for all ratios is 4 Mcps.

Firstly, it can be seen in Fig. 4 that the total accuracy of all three methods improves as the relative amount of gammas increases. This observation could suggest that the methods find it comparatively less challenging to classify gamma pulses as opposed to neutron pulses. This presumption gains further support upon examining Fig. 5 and Fig. 6, which clearly demonstrate that, across all γ/n ratios, the neutron accuracy consistently remains below that of gamma accuracy. From Fig. 4 it can be seen that as the γ/n ratio grows, the difference in total accuracy between the methods grows smaller and begins plateauing. Of particular interest is the observation in Fig. 4, where the total accuracy of the DNN applied to the

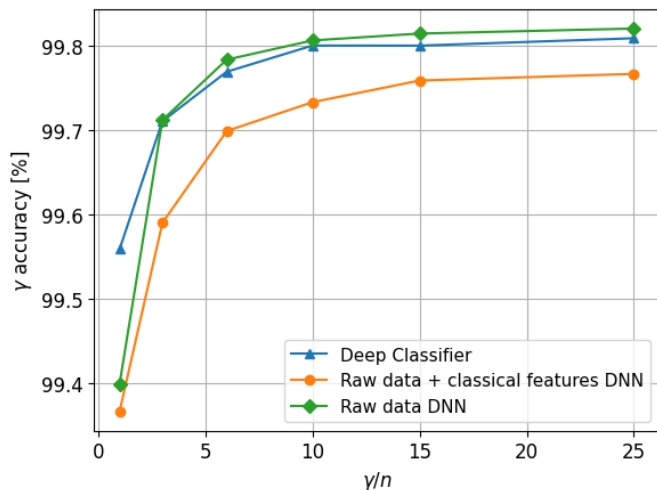


Fig. 6. The gamma accuracy achieved by the Deep Classifier and the two DNN based benchmark algorithms for ratios between the gamma and the neutron flux ranging from 1:1 up to 25:1 respectively. The total event rate for all ratios is 4 Mcps.

raw pulse eventually surpasses the total accuracy of the DNN applied to the stacking of the raw pulse and the classical features. Moreover, it can be seen from Fig. 6 that the DNN applied to the raw pulse achieves the highest gamma accuracy for almost all γ/n ratios, surpassing the gamma accuracy of the proposed Deep Classifier. Yet, this fact is accompanied by a trade-off, as evident in Fig. 5, where the DNN applied to the raw pulse shows noticeably lower neutron accuracy compared to the proposed Deep Classifier.

V. CONCLUSION

We developed general purpose PSD algorithm based on model-based deep learning methodology, which is designed for severe pileup conditions due to high event rates. The proposed method combines classical PSD features, relevant for high event rates, alongside a unique DNN based weighing architecture, inspired by the emerging technique of context-aware modulation. Discrimination capabilities were evaluated using simulated data from a CLYC based detector and compared to the discrimination capabilities of other methods. The proposed method provides superior discrimination capabilities compared to CI, NCC, and DNN based methods. Moreover, the effectiveness and contribution of the different blocks inside the proposed method are evaluated. While the results in this work are demonstrated on simulated data from a specific detector, the algorithm is beneficial for other detectors with long scintillation decay times operating in high event rates.

REFERENCES

[1] A. Dutta, P. Chandhran, K. E. Holbert, and E. B. Johnson, "Using decay time to discriminate neutron and gamma ray pulses from a clyc detector," in *2015 IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*. IEEE, 2015, pp. 1–7.

[2] S. Richards, G. Sykora, and M. Taggart, "High count rate pulse shape discrimination algorithms for neutron scattering facilities," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 989, p. 164946, 2021.

[3] A. Dutta and K. E. Holbert, "Discrimination of neutron-gamma ray pulses with pileup using normalized cross correlation and principal component analysis," *IEEE Transactions on Nuclear Science*, vol. 63, no. 6, pp. 2764–2771, 2016.

[4] C. Fu, A. Di Fulvio, S. Clarke, D. Wentzloff, S. Pozzi, and H. Kim, "Artificial neural network algorithms for pulse shape discrimination and recovery of piled-up pulses in organic scintillators," *Annals of nuclear energy*, vol. 120, pp. 410–421, 2018.

[5] J. Han, J. Zhu, Z. Wang, G. Qu, X. Liu, W. Lin, Z. Xu, Y. Huang, M. Yan, X. Zhang *et al.*, "Pulse characteristics of clyc and piled-up neutron-gamma discrimination using a convolutional neural network," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 1028, p. 166328, 2022.

[6] S. Peng, Z. Hua, Q. Wu, J. Han, S. Qian, Z. Wang, Q. Wei, L. Qin, L. Ma, M. Yan *et al.*, "Piled-up neutron-gamma discrimination system for clb using convolutional neural network," *Journal of Instrumentation*, vol. 17, no. 08, p. T08001, 2022.

[7] E. B. Johnson, C. Whitney, S. Vogel, J. F. Christian, K. Holbert, and P. Chandhran, "High event rate, pulse shape discrimination algorithm for clyc," in *2015 IEEE International Symposium on Technologies for Homeland Security (HST)*. IEEE, 2015, pp. 1–7.

[8] X. Wen and A. Enqvist, "Pulse shape discrimination of ^{137}Cs detectors at high count rate based on triangular and trapezoidal filters," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 866, pp. 129–133, 2017.

[9] Z. Zhang, Y. Zhang, C. Hu, S. Feng, J. Zhu, B. Liao, D. Liu, Q. Zhou, and H. Wang, "Detection and discrimination of α - and β -radiation with plastic scintillators based on pulse shape discrimination," *Journal of Instrumentation*, vol. 15, no. 04, p. P04008, 2020.

[10] J. Polack, M. Flaska, A. Enqvist, C. Sosa, C. Lawrence, and S. Pozzi, "An algorithm for charge-integration, pulse-shape discrimination and estimation of neutron/photon misclassification in organic scintillators," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 795, pp. 253–267, 2015.

[11] R. M. French, M. Thevenin, M. Hamel, and E. Montbarbon, "A histogram-difference method for neutron/gamma discrimination using liquid and plastic scintillators," *IEEE Transactions on Nuclear Science*, vol. 64, no. 8, pp. 2423–2432, 2017.

[12] M. Hammad, H. Kasban, R. Fikry, M. I. Dessouky, O. Zahran, S. M. Elaraby, and F. E. A. El-Samie, "Digital pulse processing algorithm for neutron and gamma rays discrimination," *Analog Integrated Circuits and Signal Processing*, vol. 101, pp. 475–487, 2019.

[13] H. Singh and R. Mehra, "Discrete wavelet transform method for high flux n - γ discrimination with liquid scintillators," *IEEE Transactions on Nuclear Science*, vol. 64, no. 7, pp. 1927–1933, 2017.

[14] B. Blair, C. Chen, A. Glenn, A. Kaplan, J. Ruz, L. Simms, and R. Wurtz, "Gaussian mixture models as automated particle classifiers for fast neutron detectors," *Statistical Analysis and Data Mining: The ASA Data Science Journal*, vol. 12, no. 6, pp. 479–488, 2019.

[15] M. Gelfusa, R. Rossi, M. Lungaroni, F. Belli, L. Spolladore, I. Wyss, P. Gaudio, A. Murari, and J. Contributors, "Advanced pulse shape discrimination via machine learning for applications in thermonuclear fusion," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 974, p. 164198, 2020.

[16] E. Doucet, T. Brown, P. Chowdhury, C. Lister, C. Morse, P. Bender, and A. Rogers, "Machine learning n/γ discrimination in clyc scintillators," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 954, p. 161201, 2020.

[17] T. Tambouratzis, D. Chernikova, and I. Pzsit, "Pulse shape discrimination of neutrons and gamma rays using kohonen artificial neural networks," *Journal of Artificial Intelligence and Soft Computing Research*, vol. 3, no. 2, pp. 77–88, 2013.

[18] J. Griffiths, S. Kleinegesse, D. Saunders, R. Taylor, and A. Vacheret, "Pulse shape discrimination and exploration of scintillation signals using convolutional neural networks," *Machine Learning: Science and Technology*, vol. 1, no. 4, p. 045022, 2020.

[19] M. Astrain, M. Ruiz, A. V. Stephen, R. Sarwar, A. Carpeño, S. Esquerri, A. Murari, F. Belli, and M. Riva, "Real-time implementation of the neutron/gamma discrimination in an fpga-based daq mtca platform using a convolutional neural network," *IEEE Transactions on Nuclear Science*, vol. 68, no. 8, pp. 2173–2178, 2021.

- [20] X. Fabian, G. Baulieu, L. Ducroux, O. Stézowski, A. Boujrad, E. Clément, S. Coudert, G. de France, N. Erduran, S. Ertürk *et al.*, “Artificial neural networks for neutron/ γ discrimination in the neutron detectors of neda,” *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 986, p. 164750, 2021.
- [21] N. M. Michels, A. J. Jimia, S. D. Clarke, H.-S. Kim, S. A. Pozzi, and D. D. Wentzloff, “Real-time classification of radiation pulses with piled-up recovery using an fpga-based artificial neural network,” *IEEE Access*, 2023.
- [22] N. Shlezinger, J. Whang, Y. C. Eldar, and A. G. Dimakis, “Model-based deep learning,” *Proc. IEEE*, vol. 111, no. 5, pp. 465–499, 2023.
- [23] N. Shlezinger, Y. C. Eldar, and S. P. Boyd, “Model-based deep learning: On the intersection of deep learning and optimization,” *IEEE Access*, vol. 10, pp. 115 384–115 398, 2022.
- [24] N. Shlezinger and Y. C. Eldar, “Model-Based Deep Learning,” *Foundations and Trends® in Signal Processing*, vol. 17, no. 4, pp. 291–416, 2023.
- [25] B. S. Budden, A. J. Couture, L. C. Stonehill, A. V. Klimenko, J. R. Terry, and J. O. Perry, “Analysis of cs 2 liycl 6: Ce 3+(clyc) waveforms as read out by solid state photomultipliers,” in *2012 IEEE Nuclear Science Symposium and Medical Imaging Conference Record (NSS/MIC)*. IEEE, 2012, pp. 347–350.
- [26] B. S. Budden, L. C. Stonehill, J. R. Terry, A. V. Klimenko, and J. O. Perry, “Characterization and investigation of the thermal dependence of cs₂liycl₆ : ce³⁺ (clyc) waveforms,” *IEEE Transactions on Nuclear Science*, vol. 60, no. 2, pp. 946–951, 2012.
- [27] N. D’olympia, P. Chowdhury, C. Lister, J. Glodo, R. Hawrami, K. Shah, and U. Shirwadkar, “Pulse-shape analysis of clyc for thermal neutrons, fast neutrons, and gamma-rays,” *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 714, pp. 121–127, 2013.
- [28] N. Dinar, D. Celeste, M. Silari, V. Varoli, and A. Fazzi, “Pulse shape discrimination of clyc scintillator coupled with a large sipm array,” *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 935, pp. 35–39, 2019.
- [29] T. Huang and Z. Zhang, “Characterization of 1-inch clyc scintillator coupled with 8× 8 sipm array,” *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 999, p. 165225, 2021.
- [30] C. Wang, X. Sun, J. Yu, J. Lv, Y. Deng, J. Li, H. Jiang, L. Luo, X. Fan, F. Gu *et al.*, “n/ γ discrimination of sipm array-based clyc detector,” *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 1036, p. 166870, 2022.
- [31] R. W. Engstrom, *Photomultiplier handbook*. RCA Corporation, 1980.
- [32] N. Ding, Y. Qin, G. Yang, F. Wei, Z. Yang, Y. Su, S. Hu, Y. Chen, C.-M. Chan, W. Chen *et al.*, “Parameter-efficient fine-tuning of large-scale pre-trained language models,” *Nature Machine Intelligence*, vol. 5, no. 3, pp. 220–235, 2023.
- [33] V. K. Chauhan, J. Zhou, P. Lu, S. Molaei, and D. A. Clifton, “A brief review of hypernetworks in deep learning,” *arXiv preprint arXiv:2306.06955*, 2023.