Study on Neutron-Gamma Discrimination Methods Based on GMM-KNN and LabVIEW Implementation

Authors: Tingmeng Ding, Yuhang Jiang, Xuanxi Wang, Xiaofei Jiang, Xiaofei Jiang

Date: 2024-07-29T00:00:00+00:00

Abstract

Machine learning algorithms are considered to be effective methods to improve the effectiveness of neutron-gamma (n-γ) discrimination. In this paper, we propose an intelligent discrimination method, GMM-KNN, which combines Gaussian Mixture Model with K Nearest Neighbor. Firstly, the unlabeled training and test data is categorized into 3 energy ranges: 0-25 keV, 25-100 keV, and 100-2100 keV. Secondly, GMM-KNN achieves small batch clustering in three energy intervals with only the tail integral Qtail and the total integral Qtotal as the pulse features. Then we select pulses with probability greater than 99% from the GMM clustering results to construct training set. Finally, we improve the KNN algorithm, so that GMM-KNN achieves the classification and regression algorithms through LabVIEW language, the outputs of GMM-KNN are the category or regression predictions. The GMM-KNN not only constructs the training set using unlabeled real pulse data, but also achieves n-γ discrimination of 241Am-Be pulses with the LabVIEW program. The experimental results show that GMM-KNN is highly robust and flexible. Even when using only 1/4 of the training set, the execution time of GMM-KNN is only 2021ms, with a difference of just 0.13% compared to the results obtained with the full training set. Furthermore, the accuracy of GMM-KNN is superior to that of the Charge Comparison Method (CCM), correctly classifying 5.52% of the ambiguous pulses. In addition, the GMM-KNN regressor achieves a higher Figure of Merit (FOM), with FOM values of 0.877, 1.262 and 1.020 corresponding to the three energy ranges, in particular showing a 32.08% improvement in the 0-25 keV. In conclusion, the GMM-KNN algorithm demonstrates accurate and readily deployable real-time n-γ discrimination performance, rendering it suitable for on-site analysis.

Full Text

Preamble

Study on neutron-gamma discrimination methods based on GMM-KNN and LabVIEW implementation

Tingmeng Ding,[1] *Yuhang Jiang,*[1] *Xuanxi Wang,*[1] *and Xiaofei Jiang*[1]*,†*
[1]Big Data and Information Engineering, GuiZhou University, Guiyang 550025, China

Machine learning algorithms are considered effective methods for improving neutron-gamma (n-γ) discrimination performance. In this paper, we propose an intelligent discrimination method, GMM-KNN, which combines Gaussian Mixture Model with K-Nearest Neighbor. Firstly, the unlabeled training and test data are categorized into three energy ranges: 0–25 keV, 25–100 keV, and 100–2100 keV. Secondly, GMM-KNN achieves small-batch clustering in these three energy intervals using only the tail integral Qtail and the total integral Qtotal as pulse features. We then select pulses with probability greater than 99% from the GMM clustering results to construct the training set. Finally, we improve the KNN algorithm and implement both classification and regression through LabVIEW, with GMM-KNN outputs being either category predictions or regression values. The GMM-KNN method not only constructs the training set using unlabeled real pulse data but also achieves n-γ discrimination of ^{241}Am–Be pulses through the LabVIEW program. Experimental results demonstrate that GMM-KNN is highly robust and flexible. Even when using only one-quarter of the training set, the execution time is merely 2021 ms, with results differing by just 0.13% compared to those obtained with the full training set. Furthermore, GMM-KNN accuracy surpasses that of the Charge Comparison Method (CCM), correctly classifying 5.52% of previously ambiguous pulses. Additionally, the GMM-KNN regressor achieves higher Figure of Merit (FOM) values of 0.877, 1.262, and 1.020 across the three energy ranges, showing a particularly notable 32.08% improvement in the 0–25 keV range. In conclusion, the GMM-KNN algorithm demonstrates accurate and readily deployable real-time n-γ discrimination performance, making it suitable for on-site analysis.

Keywords: n-γ discrimination, GMM, KNN, LabVIEW, classification, regression

Introduction

Scintillation detectors are commonly used in neutron detection [**?**, **?**, **?**]. In these detectors, neutron and gamma-ray signals exhibit different time decay constants, with neutron signals having longer decay times compared to gamma-ray signals [**?**]. This difference in pulse shape enables pulse shape discrimination (PSD) [**?**] to distinguish between the two categories. In conventional n-γ discrimination algorithms, the Charge Comparison Method (CCM) uses the PSD factor as the discriminating index [**?**], where the PSD factor is defined as the ratio of the pulse tail integral (Qtail) to the total integral (Qtotal) [?, ?]. Generally, the PSD factor corresponding to neutrons is larger than that of gamma rays. While CCM is simple and easy to use, it demonstrates weak discrimination performance in the low-energy range.

Currently, machine learning algorithms are widely employed for n-γ discrimination [?, ?]. These algorithms primarily include unsupervised and supervised learning approaches. Unsupervised learning algorithms cluster data according to its distribution in feature space without relying on pre-labeled samples and possess some ability to identify anomalous pulse events [?, ?]. Gaussian Mixture Models (GMM), commonly used in unsupervised learning, have shown good performance in n-γ discrimination [?, ?]. However, direct clustering of complete pulse data by GMM still faces challenges. Firstly, clustering high-dimensional data directly leads to the "curse of dimensionality." Secondly, when performing direct clustering on massive datasets using GMM, a large number of significant errors occur, with neutrons being incorrectly identified as gamma rays even within clusters that have clear boundaries. Furthermore, GMM clustering is only suitable for fixed datasets [?, ?].

To address these limitations, Liu et al. (2023) [?] proposed a combined method of Principal Component Analysis (PCA) and GMM clustering. They extracted three features using PCA before applying the GMM clustering algorithm. Results indicated that PCA-GMM provided a higher Figure of Merit (FOM) for n-γ discrimination compared to the CCM method. However, this approach has strict data requirements, necessitating closely aligned pulse peak positions and including only pulse tails while disregarding differences in overall pulse characteristics. Wang et al. (2022) [?] proposed a method for identifying neutrons and gamma rays using a small-batch GMM clustering algorithm. This method achieved higher FOM than the Charge Comparison Method, but when there is a large mismatch between test pulses and the trained model, the exponent takes large negative values, indicating susceptibility to outliers.

Supervised learning algorithms can classify or regress unknown signals but depend on prior knowledge. Common supervised learning algorithms include K-Nearest Neighbors (KNN) [?, ?, ?], Support Vector Machines (SVM) [?, ?], and Linear Discriminant Analysis (LDA) [?]. Among these, the KNN algorithm performs classification or regression prediction by calculating distances between test data and training samples. KNN is simple and easily portable, accurately accomplishing classification and regression tasks based on existing samples. However, the real-time performance of KNN remains debatable, and its stability under different conditions requires further exploration.

Durbin et al. (2021) [?] proposed a new method using KNN regression to improve PSD performance, enabling direct comparison with conventional PSD methods using FOM. However, the paper did not mention algorithm runtime, which is a critical indicator of real-time performance [?, ?, ?, ?]. All aforementioned studies focus on using machine learning algorithms to improve n-γ discrimination capabilities but do not address algorithmic portability and real-time performance. This paper proposes a combined GMM-KNN method to overcome limitations of single machine learning algorithms in n-γ discrimination, constructing a training set from unlabeled data to achieve discrimination of unknown pulses.

II. Method

In this paper, the GMM-KNN algorithm combines the supervised learning algorithm (GMM clustering) with the unsupervised learning algorithm (KNN classification and regression), implementing pulse discrimination through a LabVIEW program.

A. GMM Clustering

Gaussian Mixture Model is a probabilistic model that describes a dataset consisting of multiple Gaussian distributions [?]. For each component Gaussian, the probability density function is:

$$f(x|\mu,\Sigma) = \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right), \quad x \in \mathbb{R}^n$$

where n is the dimension of pulse data, μ is the n-dimensional mean vector, and Σ is the $n \times n$ covariance matrix. Obviously, the Gaussian distribution is determined by the mean vector μ and covariance matrix Σ. The initial pulse has 248 features, and such high dimensionality leads to the "curse of dimensionality." To reduce the number of pulse features, this paper uses only Qtail and Qtotal as the 2-D features of pulse data, i.e., $n = 2$.

Neglecting pulse stacking in n-γ discrimination, the Gaussian Mixture Model has only two components: neutrons and gamma rays. For a Gaussian mixture distribution with two mixed components, the probability density is:

$$f_M(x) = \sum_{i=1}^{2} \alpha_i \cdot f(x|\mu_i,\Sigma_i)$$

α_i is called the "mixture coefficient," representing the probability of selecting the ith component Gaussian, where $\alpha_i > 0$ and $\sum_{i=1}^{2} \alpha_i = 1$. The model parameters α_i, μ_i, and Σ_i are solved through iterative optimization of the Expectation-Maximization (EM) algorithm [?, ?]. Each EM iteration consists of two steps: the E-step estimates the expectation of hidden variables based on current parameters, and the M-step updates model parameters based on maximum likelihood estimation using computational results from the E-step.

Direct GMM clustering on the entire training set data produces numerous errors. To enhance clustering accuracy, we divide the data into three energy ranges [?]: 0–25 keV, 25–100 keV, and 100–2100 keV. GMM soft clustering outputs the probability that a pulse belongs to either neutrons or gamma rays. If the probability of a pulse belonging to neutrons exceeds 50%, it is classified as a neutron; otherwise, it is classified as a gamma ray. The primary objective of GMM clustering is to produce a dependable training set for subsequent use by the KNN algorithm. However, this classification method may encounter ambiguous pulses, potentially reducing KNN classification accuracy.

B. KNN Classification and Regression Algorithm

Since GMM clustering can only cluster a fixed dataset, supervised learning algorithms are required for real-time discrimination. This paper uses supervised learning to accurately represent the distribution of training set samples, which follow a Gaussian mixture distribution consisting of two components. The goal of selecting supervised learning is to ensure that test set data exhibits similar classification/clustering results as the training set. Among supervised learning algorithms such as SVM and LDA, KNN is selected for its simplicity and ease of implementation in LabVIEW programs. The KNN algorithm is known for its simplicity and accuracy, ensuring a generalization error no more than twice the error rate of the Bayes optimal classifier. To optimize KNN, it is important to determine the optimal K value, select an appropriate distance metric, and specify the decision rule.

Firstly, the key to the KNN algorithm is finding the optimal K value. When K is too large, distant points impact prediction results, leading to underfitting. Conversely, if K is too small, the model becomes less tolerant to noise and prone to overfitting. In this paper, the method to find the optimal K value is 10-Fold Cross-Validation [**?**]. This technique divides the training dataset D into 10 equal parts, using 9 parts to train the model and the remaining part to compute test accuracy. Cycling this process over each data part yields an average accuracy. The 10-fold cross-validation results exhibit a maximum value where accuracy first increases with K, then decreases after reaching the maximum. The K value corresponding to the highest average accuracy is selected as optimal. The dataset D must contain sufficient samples, and GMM clustering must produce a reliable training set. Therefore, GMM-KNN uses the training set obtained from GMM clustering as dataset D.

Additionally, the KNN algorithm calculates the distance between unknown pulses and each pulse in the training set (using Euclidean distance in this paper). For points X and Y with n features, the distance is calculated as:

$$D(X,Y)=\sqrt{(x_1-y_1)^2+(x_2-y_2)^2+\cdots+(x_n-y_n)^2}$$

Since Qtail and Qtotal are used as the two pulse features, the above equation has $n=2$. Calculating the distance between two pulses simply involves replacing x and y with Qtail and Qtotal, respectively.

Finally, the KNN algorithm outputs classification or regression results for pulses in the test set. For KNN classification, the result is the category occurring most frequently among the nearest K instances [?]. For KNN regression, the result is the average value of each feature over the nearest K instances [?].

C. GMM-KNN Classification and Regression

Both GMM and KNN are effective methods for n-γ discrimination. GMM clustering performs well in PSD analysis, but its results are limited to the current dataset, and prior knowledge obtained from the training set cannot be directly applied to the test set. Conversely, KNN accurately captures sample distribution and can be easily implemented on hardware, providing strong real-time performance and potential for real-time n-γ discrimination. However, KNN's drawback is its requirement for pre-storing the sample set.

In this study, GMM clustering aims to construct a reliable training set, and KNN utilizes this training set for discriminating unknown pulses. To integrate GMM and KNN, we propose improvements to both methods. When GMM clusters data across the entire energy range, it leads to significant misclassifications. Therefore, in GMM-KNN, data is divided into three energy partitions to enhance clustering performance. Additionally, GMM clustering results often include confusing pulses (low-probability events). Hence, we select only pulses with classification probability greater than 99% for the training set. The GMM-KNN test set consists of numerous unknown pulses, each requiring distance calculation to all samples in the training set. Using the complete training set yields the most accurate classification results but significantly increases computational cost.

In GMM-KNN classification, reducing the training dataset size has minimal impact on classification results while substantially decreasing algorithm complexity. This allows flexible selection of sample quantity within the training set for GMM-KNN classification. However, for GMM-KNN regression, using the complete training set is crucial to ensure regression prediction accuracy. To assess real-time performance, it is necessary to use a unified programming language or First-In-First-Out (FIFO) data transfer with other devices. The LabVIEW program can call or directly read acquired data, facilitating real-time GMM-KNN implementation for pulse discrimination. Through parallel computing, the LabVIEW program can concurrently calculate two decision values, resulting in simpler decision logic and faster computation. Additionally, algorithm real-time performance must consider runtime and the memory footprint of the trained model. This paper proposes the GMM-KNN algorithm, which employs only two features—Qtail and Qtotal—for both GMM clustering and KNN classification and regression. These features not only reduce data dimensionality in GMM clustering but also significantly decrease computational cost and the memory footprint of the trained model.

The block diagram of the GMM-KNN algorithm procedure is shown in [Figure 1: see original paper]: First, the method divides preprocessed training and test data into three parts separately, corresponding to energy ranges of 0–25 keV, 25–100 keV, and 100–2100 keV. Second, GMM uses Qtail and Qtotal as pulse features and performs small-batch clustering in the three energy ranges. Then, the method selects a portion of the clustering results (probability > 99%) as the KNN training set. Finally, GMM-KNN implements classification and regression algorithms through LabVIEW programming and outputs category and regression prediction values.

D. Evaluation Metrics

The GMM-KNN classification output is binary, with 0 representing gamma rays and 1 representing neutrons. Both pulse types exhibit an elliptical distribution in feature space. Comparing the difference between outputs and ground truth allows qualitative assessment of n-γ discrimination effectiveness. The GMM-KNN regression outputs consist of average values of the nearest K pulses for Qtail and Qtotal, along with their ratio. We use this ratio to calculate the Figure of Merit (FOM), where higher FOM indicates better n-γ discrimination performance.

Since neutrons and gamma rays exhibit different ratios of slow charge to total charge, CCM selects the ratio of tail integral (Qtail) to total integral (Qtotal) as the PSD factor [**?**], then calculates FOM as the discrimination metric. This paper adjusts Qtail and Qtotal to obtain optimal FOM, with Qtail taken as 68 ns and Qtotal as 124 ns ([Figure 2: see original paper]). For GMM-KNN regression, pulse features are the predicted regression values, so PSD = (Regressed Qtail) / (Regressed Qtotal). After calculating PSD values for all pulses, GMM-KNN regression uses the updated FOM as the evaluation metric to further assess n-γ discrimination effectiveness.

The PSD histogram after Gaussian fitting contains two Gaussian peaks ([Figure 3: see original paper]), and FOM is calculated as follows:

$$FOM = \frac{\Gamma}{FWHM_\gamma + FWHM_n}$$

where Γ is the distance between the two Gaussian peaks; $FWHM_\gamma$ and $FWHM_n$ are the full width at half maximum of the gamma peak and neutron peak, respectively. A larger FOM value implies higher separation between the two Gaussian peaks, indicating more effective n/γ discrimination.

III. LabVIEW Implementation of GMM-KNN

LabVIEW is a graphical programming language primarily used in control, measurement, and data acquisition fields [**?**]. Nuclear signals output from detectors can be digitized and stored using LabVIEW host computer programs, which then process these signals using discrimination algorithms. To evaluate real-time performance for acquired data, we use a unified LabVIEW programming language. The KNN classification LabVIEW program pre-sorts the training set first, making distance values completely independent of index values and greatly simplifying judgment logic. Array operations in KNN regression allow simultaneous calculation of regression values for all features, substantially simplifying the LabVIEW program. In the LabVIEW program for real-time n-γ discrimination, unknown pulses are computed as arrays. In this study, GMM-KNN classification and regression algorithms were applied to the same test set for discrimination, with output results saved as .csv files.

A. Energy Division

Pulses in the test set are random signals collected in real experiments. We divide the pulses into three parts according to different energy ranges: 0–25 keV, 25–100 keV, and 100–2100 keV. [Figure 4: see original paper] shows the LabVIEW program diagram for pulse division based on energy. The conditional diagram uses energy as the determinant, and the program counts pulses in each energy range. By sorting energy in ascending order and setting the index of "array subset vi" based on count values, we obtain three sub-arrays.

B. GMM-KNN Classification Algorithm

The program block diagram of GMM-KNN classification is shown in [Figure 5: see original paper]. GMM-KNN classification determines pulse category through three steps:

Step 1: Training set composition. In the GMM-KNN algorithm, the training set processed by GMM clustering contains 26,172 pulses (probability $> 99\%$). Test pulses calculate distances with each pulse in the training set. Direct use of the full training set is computationally intensive, so we sort training set data by probability value and take one sample every three values. These samples form a 2-D array training set M_c. The columns of M_c are the two features of a pulse, Qtail and Qtotal, and each row represents a pulse. The first m rows of the 2-D array are gamma rays and the remaining k rows are neutrons. The expression for M_c is:

$$M_c = \begin{bmatrix} \mathrm{Qtail}(\gamma_1) & \mathrm{Qtotal}(\gamma_1) \\ \mathrm{Qtail}(\gamma_2) & \mathrm{Qtotal}(\gamma_2) \\ \vdots & \vdots \\ \mathrm{Qtail}(\gamma_m) & \mathrm{Qtotal}(\gamma_m) \\ \mathrm{Qtail}(n_1) & \mathrm{Qtotal}(n_1) \\ \mathrm{Qtail}(n_2) & \mathrm{Qtotal}(n_2) \\ \vdots & \vdots \\ \mathrm{Qtail}(n_k) & \mathrm{Qtotal}(n_k) \end{bmatrix}$$

Step 2: Distance values computation. Compute distance values between the test pulse feature vector $\mu = [\mathrm{Qtail(test)}, \mathrm{Qtotal(test)}]$ and all pulses in the training set. These distance values form the 1-D array $D(M_c, \mu)$. The elements of $D(M_c, \mu)$ are Euclidean distance values between each row of array M_c and the feature vector μ:

$$D(M_c, \mu) = \begin{bmatrix} D(\gamma_1(\text{train}), \mu) \\ D(\gamma_2(\text{train}), \mu) \\ \vdots \\ D(\gamma_m(\text{train}), \mu) \\ D(n_1(\text{train}), \mu) \\ D(n_2(\text{train}), \mu) \\ \vdots \\ D(n_k(\text{train}), \mu) \end{bmatrix} = D_\gamma(M_c, \mu) + D_n(M_c, \mu)$$

Step 3: Pulse classification. The distance subsets $D_\gamma(M_c, \mu)$ and $D_n(M_c, \mu)$ correspond to gamma rays and neutrons, respectively. Take the smallest K values from each distance subset. The 10-fold cross-validation results show that the highest accuracy of 99.317% occurs at $K = 5$, so K is set to 5. Sort the two distance subsets in ascending order separately, then extract the smallest 5 gamma rays and 5 neutrons from each subset, denoted as $\gamma_1, \gamma_2, \gamma_3, \gamma_4, \gamma_5$ and n_1, n_2, n_3, n_4, n_5, respectively.

When $\gamma_3 < n_3$, we have:

$$\begin{cases} n_1 < n_2 < n_3 \\ \gamma_1 < \gamma_2 < \gamma_3 < n_3 \end{cases}$$

This relationship holds for $\gamma_4 < n_3$ or $\gamma_5 < n_3$. In these cases, the number of gamma rays must exceed the number of neutrons among the 5 nearest neighbor values.

When $n_3 < \gamma_3$, we have:

$$\begin{cases} \gamma_1 < \gamma_2 < \gamma_3 \\ n_1 < n_2 < n_3 < \gamma_3 \end{cases}$$

This relationship holds for $n_4 < \gamma_3$ or $n_5 < \gamma_3$. In these cases, the number of neutrons must exceed the number of gamma rays among the 5 nearest neighbor values.

In summary, the third values of the two subsets (γ_3 and n_3) serve as judgment values. When $\gamma_3 < n_3$, pulses are judged to be gamma rays. Conversely, when $n_3 < \gamma_3$, pulses are judged to be neutrons. In classification, the traditional KNN algorithm requires 5 judgments and 5 counts, while GMM-KNN classification needs only 1 judgment, improving algorithm efficiency and reducing complexity.

C. GMM-KNN Regression Algorithm

GMM-KNN regression requires judging the nearest K neighbors ($K = 5$) and calculating their average. The algorithm does not divide the training set into two groups but requires the complete training set. Similar to GMM-KNN classification, the LabVIEW implementation of GMM-KNN regression also consists of three steps ([Figure 6: see original paper]):

Step 1: Calculate distance values between the test pulse $\mu = [\text{Qtail(test)}, \text{Qtotal(test)}]$ and the n pulses in the training set M_r, obtaining a 1-D distance array $D(M_r, \mu)$.

$$D(M_r, \mu) = \begin{bmatrix} D(v_1, \mu) & D(v_2, \mu) & \cdots & D(v_n, \mu) \end{bmatrix}$$

Step 2: Combine this 1-D array $D(M_r, \mu)$ with the 2-D array M_r of the training set into a new 2-D array M_n. The first row of M_n contains distance values, the second row contains Qtail values of pulses in the training set, and the third row contains Qtotal values. Then sort M_n in ascending order according to the first row and intercept the first 5 columns to obtain M_5:

$$M_5 = \begin{bmatrix} D_{\text{min1}} & D_{\text{min2}} & \cdots & D_{\text{min5}} \\ \text{Qtail}(1) & \text{Qtail}(2) & \cdots & \text{Qtail}(5) \\ \text{Qtotal}(1) & \text{Qtotal}(2) & \cdots & \text{Qtotal}(5) \end{bmatrix}$$

Step 3: Calculate the KNN regression values. The first eigenvalue E_1 is the average of the first 5 columns in row 2, the second eigenvalue E_2 is the average of the first 5 columns in row 3, and the third eigenvalue $PSD_{\text{GMM-KNN}}$ is the ratio of the first two eigenvalues:

$$E_1 = \frac{1}{5}\sum_{i=1}^{5} \text{Qtail(mini)}, \quad E_2 = \frac{1}{5}\sum_{i=1}^{5} \text{Qtotal(mini)}$$

$$PSD_{\text{GMM-KNN}} = \frac{E_1}{E_2}$$

The PSD values of all pulses are stored in .csv files, and GMM-KNN regression uses PSD as the discriminating index. These PSD values can be used to compute FOM, which evaluates the n-γ discrimination ability of GMM-KNN regression. The GMM-KNN regression first calculates distance values, then synthesizes them with original pulse features into a two-dimensional array, allowing calculation of the average of K values rather than regression values for all pulses. We do not need to calculate regression values for all pulses, and the regression values of Qtail and Qtotal are calculated in parallel, greatly reducing computational complexity.

IV. Results and Discussion

The neutron source used in this experiment was a ^{241}Am–Be source, the detector was an organic liquid scintillation detector EJ-301 [**?**, **?**], and the digitizer was DT5730B. The detector collected current pulses that were digitized to obtain raw data. After preprocessing steps including smoothing filtering, normalization, and baseline recovery, the raw data was transformed into initial data and stored in the computer. The preprocessed dataset of 60,000 pulses was divided into two parts: 30,000 pulses were used for GMM clustering to obtain a reliable training set, and the remaining 30,000 pulses were reserved for testing the feasibility of the GMM-KNN algorithm.

A. GMM Clustering

Based on whether the probability value exceeds 50%, we can classify GMM clustering results into two categories. Figure 7: see original paper shows direct clustering results, where red squares and blue circles represent neutrons and gamma rays, respectively. Significant errors occur in pulse discrimination, with numerous neutrons misclassified as gamma rays. Figure 7: see original paper shows small-batch clustering results in three energy ranges, where only a few ambiguous pulses exist. Obviously, small-batch clustering reduces the misclassification rate and improves n-γ discrimination effectiveness.

Although small-batch clustering yields better results than direct clustering, [Figure 7: see original paper] exhibits red downward protrusion at energy segmentation boundaries (25 keV and 100 keV). To obtain a reliable training set, it is necessary to remove these ambiguous pulses. Within the 0–25 keV range, we progressively increase the probability threshold for pulses in the training set ([Figure 8: see original paper]). Figure 8: see original paper, (b), and (c) correspond to complete pulses, pulses with probability above 90%, and pulses with probability above 99%, respectively. [Figure 8: see original paper] demonstrates that higher probability leads to better pulse separation in the training set but also results in decreased pulse numbers in the 0–25 keV range.

presents the quantity distribution in the training set at different probability thresholds. In the 25–2100 keV range, minimal variation occurs in pulse numbers. The majority of ambiguous pulses concentrate in the 0–25 keV range. As probability increases, the number of pulses in 0–25 keV decreases significantly. When the training set consists of pulses with probability exceeding 99.9%, the quantity of pulses in 0–25 keV is even lower than that in 25–2100 keV. Additionally, the PSD histogram of the test set exhibits three distinct peaks and poor fitting performance at this stage ([Figure 9: see original paper]). Therefore, this study uses a training set composed of pulses with probability exceeding 99%.

[Figure 10: see original paper] shows the training set consisting of pulses with probability above 99%. In this case, samples with PSD values below the threshold ($PSD_{\text{threshold}}$) are removed at energy segmentation boundaries. At this stage, neutrons and gamma rays are completely separated, and the GMM-KNN algorithm only needs to select remaining pulses from this portion to construct the training set. Once constructed, GMM-KNN can flexibly choose a subset of data from the training set and implement n-γ discrimination using the LabVIEW program.

B. GMM-KNN Classification

In real-time n-γ discrimination, algorithmic efficiency is crucial, influenced by both training set size and number of pulse features. This study investigates time consumption and error rates of the GMM-KNN algorithm by employing average sampling techniques to select subsets of the complete training set at proportions of 1/2, 1/3, 1/4, 1/5, and 1/6 (). Comparing time consumption and error rates across different training set sizes reveals that when only one-quarter of the complete dataset is used, classifier time consumption reduces to approximately one-quarter of the original, resulting in an average execution time per pulse of only 67 µs, compared to 294.27 µs for the complete training set. The discriminative results between the reduced dataset and complete training set exhibit a difference of merely 0.13% (41 pulses, with 21 falling in the 0–25 keV range). This sampling method significantly reduces computational costs while ensuring reliable discrimination results, enabling flexible training set selection based on experimental latency requirements.

KNN algorithms typically use the non-zero amplitude portion of pulses as feature points, encompassing 64 points. shows that the time consumption of the 64-point KNN algorithm is 4325 ms in the 0–25 keV range. In contrast, GMM-KNN employs only two pulse features, Qtail and Qtotal, reducing execution time to 2021 ms for the same test volume. The program is parallelized, and selection of Qtail and Qtotal drastically improves processing speed. Notably, even when using the 64-point KNN algorithm, execution time remains at 4325 ms, indicating that the algorithm is not very demanding in terms of pulse feature count and can adapt to different pulse types and experimental requirements.

Imbalanced classification presents a significant challenge. In this study, we apply GMM clustering to partition a test dataset containing 10,000 gamma rays and 10,000 neutrons into 20 subsets, each comprising 1,000 pulses. We vary the gamma/neutron (γ/n) ratios from 10:1 to 10:10 and from 10:10 to 1:10, resulting in 19 different ratios. Comparing GMM-KNN classifier results with prior knowledge yields error rates and average execution times per pulse for each ratio, as shown in [Figure 11: see original paper]. Among the seven ratios ranging from 10:8 to 6:10, all exhibit error rates below 5%. The lowest error rate of 1.01% (1.23%) and average execution time per cycle of 301.84 µs (289.60 µs) are achieved at the 9:10 (10:9) ratio, consistent with the γ/n ratio of approximately 9:10 (12,542 gamma rays to 13,619 neutrons) observed in the training set. The GMM-KNN classifier demonstrates excellent real-time performance, strong adaptability, and robustness, showing great potential for real-time discrimination and on-site analysis, serving as a reference for offline n-γ discrimination analysis.

Using the complete training set, GMM-KNN performs classification simultaneously on three energy ranges (0–25 keV, 25–100 keV, and 100–2100 keV). The discrimination results from these three ranges are concatenated to obtain the complete energy range result ([Figure 12: see original paper]). The classification exhibits small errors at energy boundaries, consistent with GMM clustering results, indicating that test results accurately reflect training set data distribution. Unlike CCM, which judges pulse categories using threshold $PSD_{\text{threshold}}$, GMM-KNN classification results are more consistent with the Gaussian mixture distributions of the two components. In the low-energy range, neutrons and gamma rays overlap, and classification results cannot be directly observed from [Figure 12: see original paper]. Only by comparing CCM and GMM-KNN classification effects in the feature space containing Qtail and Qtotal can we effectively evaluate their performances.

Using Qtail, Qtotal, and energy as the X, Y, and Z axes respectively, we obtain a 3D plot of classification results in feature space ([Figure 13: see original paper]). Red cubes and blue spheres represent neutrons and gamma rays, respectively. Figure 13: see original paper shows CCM classification results, while Figure 13: see original paper shows GMM-KNN classification results. The figures reveal that for most pulses, neutrons and gamma rays have distinct cone-shaped distributions, making them easy to distinguish. However, for some lower-energy pulses, no clear boundary exists between neutrons and gamma rays (indicated by the flattened red region in [Figure 13: see original paper]), resulting in mixture and making differentiation difficult. To further compare classification results in the low-energy range, we focus on the performance of both algorithms in feature space.

[Figure 14: see original paper] shows the projection of the 3D visualization from [Figure 13: see original paper] onto the X-Y plane. In the 2D feature space, we can more intuitively observe classification results of CCM (left) and GMM-KNN (right). The two pulse types approximately exhibit elliptical distributions, with blue representing gamma rays and red representing neutrons. CCM determines pulse categories based on threshold $PSD_{\text{threshold}}$, constructing a histogram of pulse PSD and fitting it with a Gaussian distribution to separately fit neutron and gamma peaks. The midpoint between the two peaks serves as threshold $PSD_{\text{threshold}}$ for distinguishing neutrons and gamma rays. In [Figure 14: see original paper], $PSD_{\text{threshold}}$ is represented by a straight line. In comparison, GMM works better for elliptical clustering, enabling GMM-KNN to correctly classify 1,657 (5.52%) additional gamma rays. Although GMM-KNN still cannot completely discriminate neutrons and gamma rays in the low-energy range, it classifies better than CCM, with a larger number of discriminable pulses in feature space.

For pulses difficult to distinguish by both methods (the flattened red region in the 3D visualization), both CCM and GMM-KNN tend to classify these ambiguous pulses as neutrons. This occurs because gamma rays exhibit more concentrated distribution, resulting in a higher peak in the PSD histogram.

As $PSD_{\text{threshold}}$ moves toward the gamma peak, more pulses are classified as neutrons. In GMM-KNN classification, the training set is also generated by GMM, resulting in similar classification preferences to CCM.

Comparing CCM and GMM-KNN classification results qualitatively shows that GMM-KNN improves n-γ discrimination. For quantitative analysis, we employ the GMM-KNN regression algorithm.

C. GMM-KNN Regression

To obtain a more generalizable metric, GMM-KNN employs regression to compute a quantifiable FOM value. The difference between neutrons and gamma rays becomes larger at higher energies, resulting in varying FOM values across the three energy ranges. We create PSD histograms ([Figure 15: see original paper]) for CCM. Figure 15: see original paper, (b), and (c) correspond to test results within 0–25 keV, 25–100 keV, and 100–2100 keV, respectively. Figure 15: see original paper corresponds to the lowest energy range (0–25 keV), where the two Gaussian-fit peaks are least separated. In this range, neutrons and gamma rays are most difficult to discriminate, and misclassified pulses are most numerous. As energy increases, the difference between the two pulse type distributions becomes larger, and separation between Gaussian-fit peaks increases, as shown in Figure 15: see original paper and (c). These pulses can be easily discriminated. CCM is a simple and effective method in the 25–2100 keV range, discriminating well for high-energy pulses, but at lower energies, numerous ambiguous pulses exist, with ambiguity increasing as energy decreases.

For GMM-KNN regression, we also create PSD histograms ([Figure 16: see original paper]). Figure 16: see original paper, (b), and (c) correspond to test results within 0–25 keV, 25–100 keV, and 100–2100 keV, respectively. As shown in Figure 16: see original paper, for GMM-KNN, separation between the two Gaussian fitting peaks becomes larger, and the FOM value greatly improves compared to CCM. GMM-KNN discrimination ability is significantly better than CCM in the 0–25 keV range. However, in higher energy ranges (25–100 keV and 100–2100 keV), FOM improvement is smaller, with only slightly larger separation between Gaussian-fitted peaks. When n-γ discrimination effectiveness improves, PSD becomes highly concentrated in some bins because the training set is not a standard Gaussian distribution but contains several small protrusions of varying heights. After KNN regression, pulses in certain intervals become more concentrated, leading to more pronounced peaks in the histogram.

CCM and GMM-KNN regression results are shown in . In the 0–25 keV range, neither method can completely separate the two pulse types. However, GMM-KNN FOM improves by 32.08%, significantly enhancing neutron-gamma discrimination. This allows further reduction in the energy threshold for discriminable pulses, meaning neutrons and gamma rays can be distinguished across a larger energy range. Additionally, GMM-KNN FOM improvement becomes smaller as energy increases. In the 25–2100 keV range, CCM already achieves basic separation between neutrons and gamma rays, diminishing additional benefits from machine learning methods.

V. Conclusions

We designed a new intelligent discrimination algorithm, GMM-KNN. This method can construct a training set from unlabeled data and achieve n-γ discrimination of unknown pulses. GMM-KNN selects Qtail and Qtotal as pulse features to reduce dimensionality and flexibly selects samples from the training set, greatly reducing algorithm complexity. GMM-KNN also uses LabVIEW programming to achieve KNN classification and regression. The LabVIEW program can execute in parallel, though array memory consumption must be strictly controlled. We improved the KNN algorithm, particularly for classification, significantly enhancing running speed. Compared to conventional KNN, the GMM-KNN classifier requires only half the time to test the same dataset. Moreover, GMM-KNN can flexibly choose training set sample numbers based on specific experimental latency requirements. When using only one-quarter of the sample set data, the GMM-KNN classifier takes only approximately one-quarter of the time (2021 ms) required for the full dataset, while discrimination results differ by only 0.13% (41 pulses). Additionally, this method maintains stable performance across a wide range of gamma/neutron ratios, making it suitable for different experimental data. Prior to running GMM-KNN classification, we pre-sorted the two pulse sample types in the training set and executed LabVIEW parallel calculations, reducing judgment logic complexity by using only the size relationship between two judgment values to determine pulse category. GMM-KNN regression first calculates distance values, then synthesizes them with original pulse features into a two-dimensional array, allowing calculation of the average of K values rather than regression values for all pulses. To evaluate GMM-KNN discrimination effectiveness, we qualitatively analyzed scatter distributions and quantitatively calculated FOM values. In feature space, GMM-KNN classification better fit near-elliptical distributions, correctly classifying approximately 5.52% of gamma rays. The FOM values for GMM-KNN regression in the three energy ranges were 0.877, 1.262, and 1.020, respectively. Compared with CCM, this method achieved higher discrimination factors in each energy range, especially in the low-energy domain (<25 keV), where $FOM_{\text{GMM-KNN}}$ increased by 32.08% compared to FOM_{CCM}. This study verified the effectiveness and feasibility of the GMM-KNN method, which exhibits excellent real-time performance, strong adaptability, and robustness, showing great potential for real-time discrimination and making it suitable for on-site analysis while providing new ideas and methods for the n-γ discrimination field. In future work, GMM-KNN will be deployed on NI Field-Programmable Gate Array (FPGA) to form a composite device with the host computer, achieving high-accuracy real-time n-γ discrimination.

VI. List of Abbreviations

GMM: Gaussian Mixture Model
KNN: K Nearest Neighbor
CCM: Charge Comparison Method
FOM: Figure of Merit
PSD: Pulse Shape Discrimination
SVM: Support Vector Machines
LDA: Linear Discriminant Analysis
PCA: Principal Component Analysis
EM algorithm: Expectation-Maximization algorithm
FIFO: First-In, First-Out
FPGA: Field-Programmable Gate Array

References

[1] T. Alharbi, Distance metrics for digital pulse-shape discrimination of scintillator detectors. *Radiat. Phys. Chem.* 156, 205–209 (2019). doi: 10.1016/j.radphyschem.2018.11.014

[2] R.A. Winyard, J.E. Lutkin, G.W. McBeth, Pulse shape discrimination in inorganic and organic scintillators. I, *Nucl. Instrum. Methods* 95, 141–153 (1971). doi: 10.1016/0029-554X(71)90054-1

[3] G. Liu, M.D. Aspinall, X. Ma et al., An investigation of the digital discrimination of neutrons and γ rays with organic scintillation detectors using an artificial neural network. *Nucl. Instrum. Methods Phys. Res. A* 607 (2009). doi: 10.1016/j.nima.2009.06.027

[4] G. Tian, X.P. Ou-Yang, H.G. Liang et al., Digital n/γ discrimination measurement of low intensity pulsed neutron. *Nucl. Tech.* 38, 060204 (2015). doi: 10.11889/j.0253-3219.2015.hjs.38.060204 (in Chinese)

[5] Li, KN., Zhang, XP., Gui, Q. et al., Characterization of the new scintillator Cs2LiYCl6:Ce3+. *NUCL SCI TECH* 29, 11 (2018). doi: 10.1007/s41365-017-0342-4

[6] Cai, JL., Li, DW., Wang, PL. et al. Fast pulse sampling module for real-time neutron–gamma discrimination. *NUCL SCI TECH* 30, 84 (2019). doi: 10.1007/s41365-019-0595-1

[7] J.M. Adams, G. White, A versatile pulse shape discriminator for charged particle separation and its application to fast neutron time-of-flight spectroscopy, *Nucl. Instrum. Methods* 156, 459–476 (1978). doi: 10.1016/0029-554X(78)90746-2

[8] J.H. Heltsley, L. Brandon, A. Galonsky et al., Particle identification via pulse-shape discrimination with a charge-integrating ADC, *Nucl. Instrum. Methods A* 263, 441–445 (1988). doi: 10.1016/0168-9002(88)90984-9

[9] T. Sanderson, C. Scott, M. Flaska et al., Machine learning for digital pulse shape discrimination. *IEEE Nucl. Sci. Symp. Med. Imaging Conf. Rec.* 89, 1–4 (2012). doi:10.1109/NSSMIC.2012.6551092

[10] P. Raj, A. Raman, *Handbook of Research on Cloud and Fog Computing Infrastructures for Data Science*, 1st edn. (IGI global, Pennsylvania), 1–400 (2018). doi: 10.4018/978-1-5225-2974-7

[11] C. Fu, A. Di Fulvio, S.D. Clarke et al., Artificial neural network algorithms for pulse shape discrimination and recovery of piled-up pulses in organic scintillators. *Ann. Nucl. Energy* 120, 410–421 (2018). doi:10.1016/j.anucene.2018.05.054

[12] S. Conroy, G. Ericsson, J. Nyberg et al., An artificial neural network based neutron–gamma discrimination and pile-up rejection framework for the BC-501 liquid scintillation detector. *Nucl. Instrum. Methods A* 610, 534–539 (2009). doi:10.1016/j.nima.2009.08.064

[13] G. Andrew, C. Qi, A.D. Kaplan et al., Pulse pileup rejection methods using a two-component Gaussian Mixture Model for fast neutron detection with pulse shape discriminating scintillator. *Nucl. Instrum. Methods A* 988, 164905 (2021). doi:10.1016/j.nima.2020.164905

[14] Lufeng Liu, Hui Shao, Study on neutron–gamma discrimination method based on the KPCA-GMM, *Nucl. Instrum. Methods A* 1056, 168604 (2023). doi:10.1016/j.nima.2023.168604

[15] B. Blair, C. Chen, A. Glenn et al., Gaussian mixture models as automated particle classifiers for fast neutron detectors. *Stat. Anal. Data Min.* 12, 479–488 (2019). doi:10.1002/sam.11432

[16] Lufeng Liu, Hui Shao, Neutron-gamma discrimination method based on KPCA-GMM-ANN, *Radiat. Phys. Chem.* 203, 110602 (2023). doi:10.1016/j.radphyschem.2022.110602

[17] M. Durbin, M.A. Wonders, M. Flaska et al., K-Nearest Neighbors regression for the discrimination of gamma rays and neutrons in organic scintillators, *Nucl. Instrum. Methods A* 987, 164826 (2021). doi:10.1016/j.nima.2020.164826

[18] Y. Qian, W. Zhou, J. Yan et al., Comparing Machine Learning Classifiers for Object-Based Land Cover Classification Using Very High Resolution Imagery. *Remote Sens.* 7, 153–168 (2015). doi:10.3390/rs70100153

[19] J. G. Wang, P. Neskovic, L. N. Cooper, Neighborhood size selection in the k-nearest-neighbor rule using statistical confidence. *Pattern. Recogn.* 39, 417–423 (2006). doi:10.1016/j.patcog.2005.08.009

[20] Arahmane, H., Hamzaoui, EM., Ben Maissa, Y. et al. Neutron-gamma discrimination method based on blind source separation and machine learning. *NUCL SCI TECH* 32, 18 (2021). doi:10.1007/s41365-021-00850-w

[21] X. Yu, J. Zhu, S. Lin et al., Neutron/gamma discrimination based on support vector machine method. *Nucl. Instrum. Methods A* 777, 80–84 (2015). doi:10.1016/j.nima.2014.12.087

[22] J. Wen, J. Zhu, T. Xue et al., Performance of linear classification algorithms on n/γ discrimination for LaBr3:Ce scintillation detectors with various pulse digitizer properties. *J. Instrum.* 15, P02004 (2020). doi:10.1088/1748-0221/15/02/P02004

[23] A. Fernandes, N. Cruz; B. Santos et al., FPGA Code for the Data Acquisition and Real-Time Processing Prototype of the ITER Radial Neutron Camera. *IEEE Trans. Nucl. Sci.* 66, 1318–1323 (2019). doi: 10.1109/TNS.2019.2903646

[24] N. Cruz, B. Santos, A. Fernandes et al., The Design and Performance of the Real-Time Software Architecture for the ITER Radial Neutron Camera. *IEEE Trans. Nucl. Sci.* 32, 1310–1317 (2019). doi: 10.1109/TNS.2019.2907056

[25] A. Fernandes, R.C. Pereira, B. Santos et al., New FPGA based hardware implementation for JET gamma-ray camera upgrade, *Fusion. Eng. Des.* 128, 188–192 (2018). doi: 10.1016/j.fusengdes.2018.02.038

[26] L. M. Simms, B. Blair, J. Ruz et al., Pulse discrimination with a Gaussian mixture model on an FPGA. *Nucl. Instrum. Methods A* 900, 1–7 (2018). doi: 10.1016/j.nima.2018.05.039

[27] B. Blair, C. Chen, A. Glenn et al., Gaussian mixture models as automated particle classifiers for fast neutron detectors. *Stat. Anal. Data Min.* 12, 479–488 (2019). doi:10.1002/sam.11432

[28] Shahmohammadi Beni, M., Krstic, D., Nikezic, D. et al. Studies on unfolding energy spectra of neutrons using maximum-likelihood expectation–maximization method. *NUCL SCI TECH* 30, 134 (2019). doi: 10.1109/ICIP.2001.958974

[29] A.P. Dempster, N.M. Laird, D.B. Rubin, Maximum Likelihood from Incomplete Data Via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)* 39, 1–22 (1977). doi:10.1111/j.2517-6161.1977.tb01600.x

[30] F.P. Wang, M.H. Yang, J.Y. Wang et al., A comparison of small-batch clustering and charge-comparison methods for n/γ discrimination using a liquid scintillation detector. *Nucl. Instrum. Methods A* 1028, 166379 (2022). doi:10.1016/j.nima.2022.166379

[31] X.X. Wang, Q.L. Tang, X.F. Jiang, Simulation of n-γ pulse signal discrimination based on KNN classification algorithm. *Electronic Measurement Technology* 45, 164–170 (2022). doi:10.19651/j.cnki.emt.2209025 (in Chinese)

[32] S. Zhang, X. Li, M. Zong et al., Efficient kNN Classification With Different Numbers of Nearest Neighbors. *IEEE Trans. Neur. Net. Lear.* 29, 1774–1785 (2018). doi:10.1109/TNNLS.2017.2673241

[33] J. Goldberger, S. Roweis, G. Hinton et al. Neighbourhood components analysis, *Adv. Neural Inf. Process. Syst* 17, 513–520 (2005). doi:10.1109/TCSVT.2013.2242640

[34] Zhang, YQ., Hu, LQ., Zhong, GQ. et al. Development of a high-speed digital pulse signal acquisition and processing system based on MTCA for liquid scintillator neutron detector on EAST. *NUCL SCI TECH* 34, 150 (2023). doi:10.1007/s41365-023-01318-9

[35] L. Chang, Y.D. Liu, L. Du et al. Pulse shape discrimination and energy calibration of EJ301 liquid scintillation detector. *Nucl. Tech.* 38, 020501 (2015). doi:10.11889/j.0253-3219.2015.hjs.38.020501 (in Chinese)

[36] L. Stevanato, D. Cester, G. Nebbia et al. Neutron detection in a high gamma-ray background with EJ-301 and EJ-309 liquid scintillators. *Nucl. Instrum. Methods A* 690, 96–101 (2012). doi:10.1016/j.nima.2012.06.047

[37] G.Q. Zhong, *Dissertation, Investigation of fusion neutron diagnostic technology on EAST device.* University of Science and Technology of China (2017) (in Chinese)

Note: Figure translations are in progress. See original paper for figures.