Journal of Quality Measurement and Analysis *JQMA* 21(4) 2025, 223-247
e-ISSN: 2600-8602 *http://www.ukm.my/jqma*
https://doi.org/10.17576/jqma.2104.2025.12

STATISTICAL COMPUTING FOR FORENSIC FIREARM PATTERN IDENTIFICATION: EVALUATING A FISHER LINEAR DISCRIMINANT ANALYSIS-BASED ALGORITHM AGAINST FIXED-IMPULSE NOISE

(*Pengkomputeran Statistik untuk Pengecaman Pola Senjata Api Forensik: Penilaian Algoritma Berasaskan Analisis Pembezalayan Linear Fisher Terhadap Hingar Dedenyut Bernilai Tetap*)

ZUN LIANG CHUAN*, ABRAHAM LIM BING SERN, TAN CHEK CHENG,
ABDUL AZIZ JEMAIN & CHOONG-YEUN LIONG

ABSTRACT

Forensic laboratories analyze firearm-related evidence utilizing established ballistic identification systems such as the Integrated Ballistic Identification System (IBIS), Advanced Ballistics Analysis System (ALIAS), EVOFINDER Automated Ballistic Identification System, and CONDOR Ballistic Identification System. However, these systems require physical verification by experts, resulting in the process being time-consuming. Previous studies developed a Fisher Linear Discriminant Analysis (FLDA)-based ballistic identification algorithm to address this limitation, encompassing image pre-processing, feature extraction, and identification. This study evaluates the robustness of the FLDA-based ballistic identification algorithm against fixed-value impulse noise, including pepper, salt, and salt-and-pepper noise. A dataset of ballistic images from five Vektor Parabellum SP1 9mm pistols (Pistols A–E) was contaminated with noise levels ranging from 10% to 90%. The results demonstrate that the algorithm maintains high identification rates exceeding 90% for images with up to 90% pepper and salt noise, utilizing maximum-ranked and minimum-ranked ordered denoising spatial kernels. Similarly, high identification rates of up to 80% were achieved for salt-and-pepper noise. These findings highlight the robustness of FLDA-based statistical computing techniques in forensic firearm pattern identification, reducing reliance on physical verification and expediting forensic investigation. Furthermore, this study aligns with the United Nations Sustainable Development Goals (SDGs), particularly SDG9 (Industry, Innovation, and Infrastructure), by fostering Artificial Intelligence (AI)-driven forensic advancements and SDG16 (Peace, Justice, and Strong Institution) by strengthening forensic accuracy in criminal investigations, ultimately contributing to national security and judicial efficiency.

Keywords: Fisher Linear Discriminant Analysis (FLDA); ballistic pattern identification; fixed-value impulse noise; statistical computing; United Nations Sustainable Development Goals (SDGs)

ABSTRAK

Makmal forensik menganalisis bukti berkaitan senjata api dengan menggunakan sistem pengecaman balistik yang mantap seperti *Integrated Ballistic Identification System* (IBIS), *Advanced Ballistics Analysis System* (ALIAS), *EVOFINDER Automated Ballistic Identification System*, dan *CONDOR Ballistic Identification System*. Walau bagaimanapun, sistem-sistem ini memerlukan pengesahan fizikal oleh pakar, mengakibatkan proses tersebut memakan masa. Sorotan kajian terdahulu telah membangunkan satu algoritma pengecaman balistik berasaskan Analisis Pembezalayan Linear Fisher (FLDA) bagi menangani kekangan ini, yang merangkumi pra-pemprosesan imej, pengekstrakan fitur, dan pengecaman. Kajian ini menilai keteguhan algoritma pengecaman balistik berasaskan FLDA terhadap hingar dedenyut bernilai tetap, termasuk hingar lada, hingar garam, dan hingar garam-dan-lada. Satu set data imej balistik daripada lima pistol Vektor Parabellum SP1 9mm (Pistol A–E) tercemari dengan tahap hingar antara 10% hingga 90%. Hasil kajian menunjukkan bahawa algoritma tersebut mengekalkan kadar pengecaman yang tinggi melebihi 90% bagi imej yang tercemari dengan hingar lada dan

garam sehingga 90% tahap hingar, dengan menggunakan kernel nyahhingar reruang berpangkat maksimum dan minimum. Kadar pengecaman tertinggi turut tercapai bagi hingar garam-dan-lada pada tahap hingar sehingga 80%. Penemuan ini menekankan keteguhan kaedah pengkomputeran statistik berasaskan FLDA dalam pengecaman pola senjata api forensik, mengurangkan kebergantungan terhadap pengesahan fizikal dan mempercepatkan siasatan forensik. Tambahan pula, kajian ini selaras dengan Matlamat Pembangunan Mampan (SDG) Pertubuhan Bangsa-Bangsa Bersatu, khususnya SDG9 (Industri, Inovasi dan Infrastruktur) melalui pemerkasaan kemajuan forensik berasaskan Kecerdasan Buatan (AI), dan SDG16 (Keamanan, Keadilan dan Institusi Kukuh) melalui peningkatan ketepatan forensik dalam siasatan jenayah yang akhirnya menyumbang kepada keselamatan negara dan keberkesanan sistem kehakiman.

Kata kunci: Analisis Pembezalayan Linear Fisher (FLDA); pengecaman pola balistik; hingar dedenyut bernilai tetap; pengkomputeran statistik; Matlamat Pembangunan Mampan Pertubuhan Bangsa-Bangsa Bersatu (SDG)

1. Introduction

In law enforcement, compelling evidence is crucial, particularly in firearm-related cases. Recovered fired bullets and cartridge cases provide valuable insights into firearm characteristics such as type, caliber, model, and potential links to past criminal activities. Ballistic experts primarily analyze rifling impressions on bullets and distinctive impressions on cartridge cases, including breech faces, ejectors, and firing pin impressions (Figure 1). While rifling impressions may be distorted upon impact, ejector and breech face impressions lack consistency. Experts and literature (Ghani *et al*. 2010; Chuan *et al*. 2013a, b; Chuan 2014; Chuan *et al*. 2017; Chuan *et al*. 2023) identify firing pin impressions as the most reliable feature for ballistic identification.

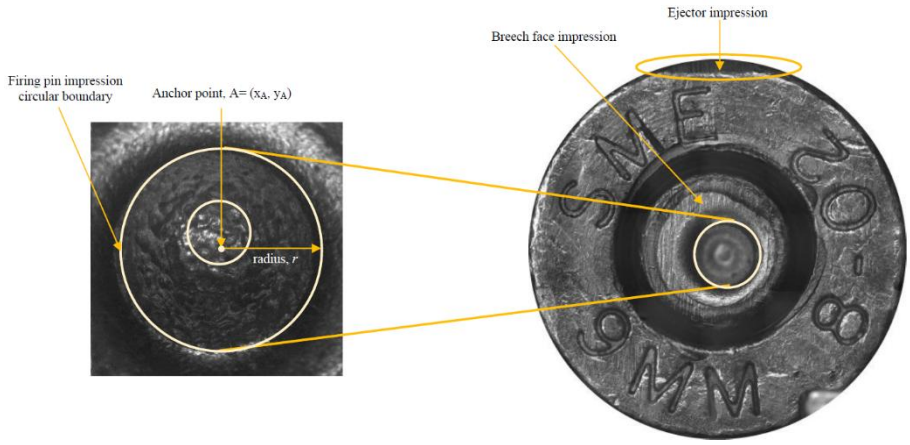

Figure 1: Distinctive impressions on fired cartridge cases

The origins of forensic ballistics identification trace back to Philip Gravelle, who pioneered the utilization of a low-contrast optical comparison microscope in forensic laboratories. The method was later applied by Calvin Goddard and Charles Waite during the investigation of the Saint Valentine's Day massacre in 1929 (Nichols 1997; Heard 2008). Despite its historical

significance, optical comparison remained subjective, frequently leading to potential errors, prolonged investigation times, and reliance on expert interpretation.

With advancements in technology, various automated ballistics identification systems have emerged for forensic applications. Commercial imaging systems such as the Integrated Ballistics Identification System (IBIS), Advanced Ballistics Analysis System (ALIAS), EVOFINDER Automated Ballistic Identification System, and CONDOR Ballistic Identification System have significantly reduced investigation times from weeks to hours by generating a ranked list of potential matches. However, these systems have inherent limitations, including illumination inconsistencies, reflection artifacts, and the need for expert validation. Consequently, the development of feature-based ballistic identification algorithms has gained traction in forensic research.

Existing studies have explored the effectiveness of these algorithms under controlled conditions; however, limited research has assessed their performance in noisy imaging environments. Chuan *et al*. (2017) previously evaluated the robustness of their Fisher Linear Discriminant Analysis (FLDA)-based ballistic identification algorithm under random-value impulse noise. Nevertheless, fixed-value impulse noise, which can result from signal errors during image acquisition, remains an overlooked challenge. To address this gap, this study systematically evaluates the performance of an FLDA-based ballistics identification algorithm against simulated fixed-value impulse noise, including pepper, salt, and salt-and-pepper noise.

The primary motivation for selecting the FLDA-based ballistic identification algorithm lies in its demonstrated computational efficiency, identification accuracy, and robustness under noise-free scenarios, making it a suitable candidate for further evaluation in noisy environments. Unlike conventional machine learning and deep learning algorithms that typically depend on hold-out or cross-validation techniques, this study introduces a novel statistical evaluation framework specifically designed to address overfitting and underfitting risks associated with limited datasets. Deep learning algorithms, particularly Convolutional Neural Network-based (CNN-based) architectures, including both single-stage and two-stage architectures, generally require large and diverse training datasets to generalize effectively. In contrast, the dataset utilized in this study is not merely limited in size but also highly sensitive to variations in lighting, background, and object orientation. Moreover, CNN-based architectures tend to be less effective for analyzing small regions of interest (**ROI**) such as the minute firing pin impression examined in this study, which occupies a tiny portion of a 9mm cartridge case.

Although CNN-based architectures are frequently praised for real-time processing capabilities, such performance is not essential in forensic ballistic identification. Additionally, CNN-based architectures impose considerable complexities in terms of architecture design and optimization that require careful tuning of architectural parameters such as the number of layers, neurons, epochs, and other hyperparameters. In contrast, the FLDA algorithm offers a more interpretable and computationally efficient alternative under the practical constraints of forensic casework. These factors collectively support the appropriateness of employing a conventional algorithm such as FLDA over more complex CNN-based architectures in this context.

The key contributions of this study include proposing a method to simulate fixed-value impulse noise for ballistic images, introducing a novel statistical evaluation approach specifically designed for limited sample forensic datasets, and assessing the robustness of the FLDA-based ballistic identification algorithm under various noise conditions. To facilitate a comprehensive analysis, the remainder of this article is structured as follows: Section 2 reviews related work, Section 3 presents the pseudo-code for simulating noisy images utilizing R statistical software, Section 4 outlines the research methodology, including denoising techniques and the Cross Industry Standard Process for Data Mining (CRISP-DM) framework,

Section 5 presents the analysis results and discussion, and Section 6 concludes the study with final remarks.

2. Related Works

In response to the limitations of commercial ballistics identification systems highlighted in earlier studies, researchers have increasingly developed semi-automated and fully automated feature-based ballistic identification algorithms to support forensic firearm examination. These approaches incorporate advanced image processing techniques, statistical feature extraction, and machine learning or deep learning algorithms to enhance the accuracy, objectivity, and operational efficiency of ballistics identification. Early efforts include the work of Xin *et al*. (2000), who introduced an automatic ballistic identification algorithm that extracted image features from breech face, extractor, and center-firing pin impressions. Their method incorporated texture-based features and geometric properties, such as the firing pin radius (r), and the deviation distance of the anchor point (A). Although the identification rate exceeded 90%, the approach suffered from high computational costs due to reliance on the Circles Hough transform (CHT). To address this, Zhou *et al*. (2001) proposed a more advanced ballistic identification algorithm integrating local orientation analysis with an active snake model to enhance feature extraction. Nevertheless, their algorithm remained dependent on CHT and thus continued to face similar computational limitations.

Thumwarin (2008) explored moment invariants (MI) extracted from primer rotation, incorporating Fourier coefficients for feature computation. While the proposed ballistic identification algorithm achieved promising identification accuracy, it faced challenges related to primer segmentation and the robustness of moment features under translation and scaling. Subsequently, Leng and Huang (2012) introduced an improved automatic ballistic identification algorithm based on translation, rotation, and scale (TRS) invariant moment features. Their framework employed image pre-processing techniques such as power-law transformation, Otsu's thresholding, and Sobel sharpening. However, computational cost remained a concern, as features were extracted from the entire cartridge case rather than localized impressions.

Building on global efforts, researchers in Malaysia have made significant contributions to feature-based ballistic identification algorithms. Ghani *et al*. (2009a, 2009b; 2010; 2018) pioneered early ballistic identification algorithms utilizing statistical moments features and TRS-non-invariant geometric moments. Their research evolved from simple summary statistics of center-firing pin impressions to the extraction of geometric moment non-invariants, highlighting the trajectory of local forensic approaches. Liong *et al*. (2012) extended this line of work by applying Principal Component Analysis (PCA) to reduce feature dimensionality from 68 extracted features, and employed FLDA (Fisher 1936) as the identification algorithm, reporting improved identification performance.

In parallel, Kamaruddin *et al*. (2012) introduced a neural network-based ballistics identification algorithm by replacing the FLDA algorithm with a two-layer feed-forward backpropagation neural network (FBPNN) algorithm utilizing a tansig-tansig activation function. Ghani *et al*. (2018) later refined Kamaruddin *et al*.'s (2012) ballistic identification algorithm by incorporating a tansig-purelin configuration, improving feature learning. Despite these advances, the proposed algorithm required manual interpretation of center-firing pin impression locations and lacked TRS-invariant properties. To address these limitations, Chuan *et al*. (2013a, 2013b) and Chuan (2014) developed an enhanced automated feature-based ballistic identification algorithm utilizing Orthogonal Legendre Moment Invariants (OLMIs) within an FLDA-based ballistic identification framework. Their work prioritized translation

and scale invariants while rejecting Zernike Moment Invariants (ZMIs) due to computational complexity and unit disk constraints.

Chuan *et al*. (2017) further validated the FLDA-based ballistic identification algorithm by simulating impulse noise up to 70% and achieving over 90% identification accuracy utilizing median filtering. In contrast, Razak *et al*. (2017) applied Canny edge detection with CHT and achieved 93% identification accuracy, although Chuan (2014) demonstrated that Laplacian kernels outperformed Canny (1986) and other sharpening techniques such as Sobel, and Marr and Hildreth (1980) filters in feature enhancement. More recently, Liong *et al*. (2020) proposed a mobile-based ballistic identification system achieving 98% identification accuracy utilizing an FBPNN with a sigmoid-linear activation function. However, the approach lacked practical validation in segmentation and did not incorporate TRS-invariant features.

A comparative study by Chuan *et al*. (2023) evaluated five types of two-dimensional moment invariants for ballistic identification, including OLMIs, Hu Moment Invariants (HMIs), Tsirikolias-Mertzois Moment Invariants (TMMIs), Pan-Keane Moment Invariants (PKMIs), and Central Geometric Moments (CGMs), utilizing FLDA as the identification algorithm. Although CGMs achieved the highest identification accuracy for weakly associated feature sets, they lacked TRS invariants, confirming earlier limitations reported by Leng and Huang (2012). This limitation is particularly relevant to the dataset utilized in the present study, which is highly sensitive to variations in lighting, background, and object orientation.

Despite these advancements, the potential impact of fixed-value impulse noise, such as salt, pepper, and salt-and-pepper noise introduced during ballistic image acquisition, remains underexplored in existing literature. To address this gap, the present study re-evaluates the robustness of the FLDA-based ballistics identification algorithm under simulated noise conditions. This investigation fills a critical void by assessing the algorithm's performance under real-world signal degradation scenarios, thereby supporting its applicability in forensic environments. Although ANN-based architectures, particularly CNN-based architectures, have recently gained attention in ballistic identification, they remain impractical in forensic settings. Their effectiveness is frequently constrained by limited dataset sizes and high sensitivity to variations in lighting, background complexity, and object orientation. Moreover, CNN-based architectures are typically less effective when applied to small objects, such as in pistol detection tasks (Li *et al.* 2021; Sood *et al.* 2021). This is especially relevant to the firing pin impression examined in this study, which appears as a minute cavity on a cartridge case with an overall diameter of merely 9mm. Given these challenges, conventional algorithms such as FLDA offer a more practical and interpretable solution, particularly under the constrained and variable conditions typically found in forensic applications.

3. Simulated Noisy Image

In digital signal processing, noisy images refer to undesired errors that degrade the quality of digital images. These signal errors may occur during digitization and image formation, falling into four primary categories: additive noise, multiplicative noise, impulse noise, and quantization noise. However, this study specifically focuses on impulse noise due to its primary objective of evaluating the robustness of the FLDA-based ballistic identification algorithm utilizing simulated fixed-value impulse noisy images, including unipolar and bipolar noisy images.

Impulse noise, such as pepper, salt, and salt-and-pepper noise, arises from transmission errors, sensor malfunctions, memory damage, and timing errors during analog-digital conversion (Chuan 2014; Chuan *et al*. 2017; Alanazi *et al*. 2023; Toktas *et al*. 2023). Pepper and salt noises are classified as unipolar, with pixel values of 0 and 1, respectively,

contaminating gray-scale images when compressed from the original intensity range of 0–255 to 0–1. In contrast, salt-and-pepper noise is a bipolar mixture of salt and pepper noises, representing a typical form of impulse noise affecting images in this study.

Mathematically, suppose that $\mathbf{I} = g\left[\left(x,y\right)\right]_{\lambda^2}$ represents a digital image, and $\mathbf{I}_{\mathrm{c}} = g_{\mathrm{c}}\left[\left(x,y\right)\right]_{\lambda^2}$ represents an image $\mathbf{I}$ contaminated with $\theta\times100\%$ where $g, g_{\mathrm{c}} \in \frac{1}{255}\{0,1,2,\ldots,255\}$ and $x,\left(y\right) = 0,1,2,\ldots,\lambda-1,\left(\lambda-1\right).$ The pixel intensity, $g_{\mathrm{c}}\left(x,y\right)$ of an image $\mathbf{I}_{\mathrm{c}}$ can be expressed as Eq. (1).

$$g_{\mathrm{c}}\left(x,y\right) = g\left(x,y\right) + \left[\left(u<b\right)\times\min\left(g\left(x,y\right)\right)\right] + \left[\left(b\le u<d\right)\times\max\left(g\left(x,y\right)\right)\right] \tag{1}$$

where u is a uniform random variable, $u\left(0,1\right)$. When $d=0,$ the image $\mathbf{I}_{\mathrm{c}}$ is contaminated with pepper noise, while for $b=0,$ it is affected by salt noise. If $b=d,$ the image experiences salt-and-pepper noise. To simulate noisy images, the study implements pseudo-code in R statistical software for the generation of pepper (Figure 2), salt (Figure 3), and salt-and-pepper (Figure 4) noise at various contamination levels.

```
Function PerturbImage(P, theta)
   #Get dimensions of the input matrix P
   numRows <- nrow(P)
   numCols <- ncol(P)
   #Calculate the number of elements to perturb based on theta percentage
   numElementsToPerturb <- round(theta * numRows * numCols / 100)
   #Randomly select coordinates to perturb
   selectedRows <- sample(1:numRows, numElementsToPerturb, replace=True)
   selectedCols <- sample(1:numCols, numElementsToPerturb, replace=True)
   #Combine selected coordinates to avoid duplicates
   uniqueCoordinates <- unique(cbind(selectedRows, selectedCols))[1:numElementsToPerturb,]
   #Create an empty numeric vector to store perturbation values
   perturbationValues <- numeric(numElementsToPerturb)
   #Apply perturbation to the input matrix P
   P[cbind(uniqueCoordinates[, 1], uniqueCoordinates[, 2])] <- perturbationValues
   #Return the perturbed matrix P
   return(P)
End Function
```

Figure 2: Pseudo-code for simulating pepper noise

```
Function PerturbImage(P, theta)
   #Get dimensions of the input matrix P
   numRows <- nrow(P)
   numCols <- ncol(P)
   #Calculate the number of elements to perturb based on theta percentage
   numElementsToPerturb <- round(theta * numRows * numCols / 100)
   #Randomly select coordinates to perturb
   selectedRows <- sample(1:numRows, numElementsToPerturb, replace=True)
   selectedCols <- sample(1:numCols, numElementsToPerturb, replace=True)
   #Combine selected coordinates to avoid duplicates
   uniqueCoordinates <- unique(cbind(selectedRows, selectedCols))[1:numElementsToPerturb,]
   #Generate random values for perturbation
   perturbationValues <- sample(1, numElementsToPerturb, replace=True)
   #Apply perturbation to the input matrix P
   P[cbind(uniqueCoordinates[, 1], uniqueCoordinates[, 2])] <- perturbationValues
   #Return the perturbed matrix P
   return(P)
End Function
```

Figure 3: Pseudo-code for simulating salt noise

```
Function PerturbImage(P, theta)
   #Get dimensions of the input matrix P
   numRows <- nrow(P)
   numCols <- ncol(P)
   #Calculate the number of elements to perturb based on theta percentage
   numElementsToPerturb <- round(theta * numRows * numCols / 100)
   #Randomly select coordinates to perturb
   selectedRows <- sample(1:numRows, numElementsToPerturb, replace=True)
   selectedCols <- sample(1:numCols, numElementsToPerturb, replace=True)
   #Combine selected coordinates to avoid duplicates
   uniqueCoordinates <- unique(cbind(selectedRows, selectedCols))[1:numElementsToPerturb,]
   #Generate random binary value for perturbation
   pertubationValues <- sample(0:1, numElementsToPerturb, replace=True)
   #Apply perturbation to the input matrix P
   P[cbind(uniqueCoordinates[,1], uniqueCoordinates[,2])]<-pertubationValues
   #Return the perturbed matrix P
   return(P)
End Function
```

Figure 4: Pseudo-code for simulating salt-and-pepper noise

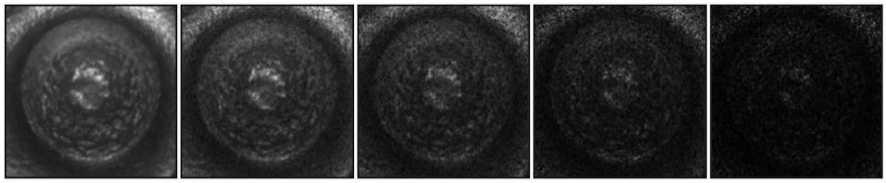

Figure 5: Simulated pepper noisy images contaminated with θ variations $\theta = 0.0, 0.2, 0.4, 0.6$ and 0.8

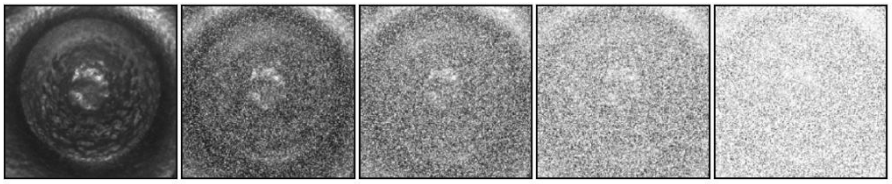

Figure 6: Simulated salt noisy images contaminated with θ variations $\theta = 0.0, 0.2, 0.4, 0.6$ and 0.8

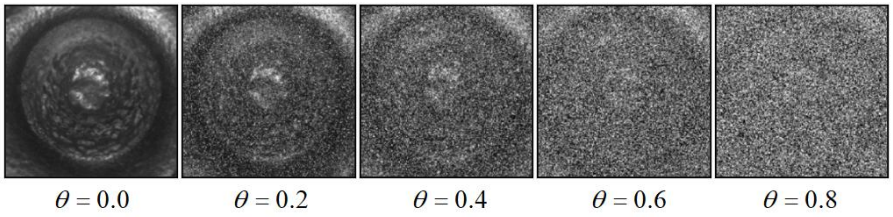

Figure 7: Simulated salt-and-pepper noisy images contaminated with θ variations $\theta = 0.0, 0.2, 0.4, 0.6$ and 0.8

The simulated images corresponding to different noise intensities $(\theta = 0.0, 0.2, 0.4, 0.6,$ and $0.8)$ (Figures 5–7) illustrate the contamination process. These illustrated noisy images contaminated by pepper, salt, and salt-and-pepper noise at $\theta = 0.0, 0.2, 0.4, 0.6,$ and $0.8,$ respectively. This study emphasizes the significance of the Signal-to-Noise Ratio (SNR) for assessing noisy image levels. However, it takes a different approach by not utilizing SNR to simulate noise levels on images. Instead, this study utilizes contamination percentages, offering distinct advantages. This approach allows for easy simulation of noise based on investigated levels and facilitates the identification of noise levels in images when unknown. This is particularly beneficial as varying spatial kernels may be necessary for effectively removing contaminated noise levels in noisy images.

4. Theoretical Background of the Proposed Algorithm

This study adopts the CRISP-DM framework, a widely recognized methodology applied in various disciplines, including education (Liang *et al*. 2024; Okoye *et al*. 2024), medical research (Mirza *et al*. 2023), and social sciences (Yunus & Loo 2024). Comprising six phases, including business understanding, data understanding, data preparation, modeling, evaluation, and deployment. This data science framework ensures a structured and systematic approach. The following sections describe its application in this study.

4.1. *Business understanding*

The primary objective of this study is to evaluate the robustness of the FLDA-based ballistic identification algorithm when applied to unipolar and bipolar fixed-value impulse noise conditions. Specifically, this study aims to determine the optimal spatial kernel type and size for denoising, ensuring that identification rates exceed 90% across all noise levels. Additionally, the statistical significance of the identification rate differences between noisy and non-noisy images is examined. To ensure reliable performance evaluation, ten random seeds were selected within the range of 1 to 9999, each representing a distinct scenario.

The dataset consists of 747 non-noisy images, acquired utilizing the commercialized ballistics identification system (CONDOR) at the Royal Malaysia Police (RMP) forensics laboratory in Cheras, Kuala Lumpur, Malaysia. Due to daily acquisition restrictions, obtaining non-noisy images posed challenges, necessitating standardized illumination conditions to maintain image consistency. This study also addresses financial constraints in software selection. While Matrix Laboratory (MATLAB) is widely utilized for image processing and pattern recognition, its high cost led to the adoption of R statistical software as a cost-effective alternative. This decision not merely validates the FLDA-based ballistic identification algorithm but also reduces execution time and expenses associated with ballistic identification tasks. Figure 8 provides a schematic representation of the FLDA-based ballistic identification algorithm, with subsequent sections detailing each phase.

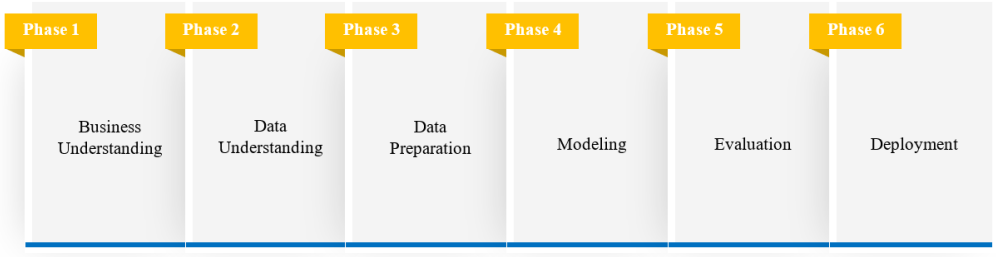

Figure 8: Schematic representation of the FLDA-based ballistic identification algorithm

4.2. *Data understanding*

The data understanding phase establishes a foundational basis for subsequent data preparation. In this study, a dataset of 747 non-noisy images was collected from five semi-automatic Parabellum Vektor SP1 9mm pistols, labeled as Pistol A, Pistol B, Pistol C, Pistol D, and Pistol E. These models were selected based on guidance from the RMP, ensuring consistency in age, caliber, and structure design. This selection was particularly crucial as the center-firing pin impressions on cartridge cases from these models are difficult to distinguish without specialized tools. These pistols were selected due to their frequent utilization in criminal activities in Malaysia.

Initially, 150 cartridge cases were collected for each pistol. However, a preliminary screening identified three cartridges with one from Pistol D and two from Pistol E that exhibited center-firing pin impressions inconsistent with the rest of the dataset. Upon verification by the RMP, these were classified as outliers and excluded from this study. The remaining 747 center-firing pin impression images were captured utilizing the CONDOR system and stored in Joint Photographic Experts Group (JPEG) format, each with a resolution of 389 pixels$\times$ 389 pixels for further analysis. Subsequently, 50 center-firing pin impression images were randomly selected for each pistol utilizing the simple random sampling method applied in this study.

4.3. *Data preparation*

Data preparation involves a series of essential steps, including data selection, cleaning, formatting, and integration. This study focuses exclusively on the center-firing pin impressions of fired cartridge cases, as this feature has been shown to be structurally resilient and resistant to deformation compared to other surfaces. Its durability makes it an ideal candidate for ballistic identification (Ghani *et al.* 2010; Chuan *et al.* 2013a). This study deliberately excluded rim-firing pin impression, given that pistols utilizing rim-firing mechanisms lack the capability to inflict fatal injuries. As a result, they are rarely utilized in serious criminal activities. By concentrating on center-firing pin impressions, this study ensures the forensic relevance of the dataset. All images were acquired in a controlled environment utilizing the CONDOR system, maintaining consistency in lighting and positioning to facilitate accurate identification.

4.3.1. *Denoising noisy images*

To evaluate the robustness of the denoising process, artificial noise, such as pepper, salt, and salt-and-pepper, was introduced into the dataset at varying levels $\theta\times100\%$. Initial identification tests revealed a significant reduction in identification accuracy when denoising was not applied. Consequently, a spatial kernel-based denoising approach was implemented before feature extraction, following the FLDA-based ballistic identification algorithm. The denoising kernel was selected based on the physical characteristics of the center-firing pin impressions. Table 1 summarizes the statistical spatial kernels utilized for denoising, each tailored to a specific noise type. The contraharmonic mean, minimum, first quartile (Q_1), median, third quartile (Q_3), and maximum filters were applied based on their effectiveness in removing different types of noise.

Table 1: Denoising spatial kernels for a noisy image

Noise	**Denoising spatial kernel**					
	Contraharmonic Mean	**Minimum**	Q_1	**Median**	Q_3	**Maximum**
Pepper	Yes	Yes	Yes	Yes	No	No
Salt	Yes	No	No	Yes	Yes	Yes
Salt-and-pepper	No	No	No	Yes	No	No

Suppose that $\mathbf{K}=\left[g_{\mathrm{c}}\left(x,y\right)\right]_{h^2}$ represents the denoising spatial kernel with a size of $h^2=\left(2\varsigma+1\right)^2$ as expressed in Eq. (2).

$$\mathbf{K}=\begin{bmatrix} g_{\mathrm{c}}(x-\varsigma, y+\varsigma) & \cdots & g_{\mathrm{c}}(x+0, y+\varsigma) & \cdots & g_{\mathrm{c}}(x+\varsigma, y+\varsigma) \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ g_{\mathrm{c}}(x-\varsigma, y+0) & \cdots & g_{\mathrm{c}}(x+0, y+0) & \cdots & g_{\mathrm{c}}(x+\varsigma, y+0) \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ g_{\mathrm{c}}(x-\varsigma, y-\varsigma) & \cdots & g_{\mathrm{c}}(x+0, y-\varsigma) & \cdots & g_{\mathrm{c}}(x+\varsigma, y-\varsigma) \end{bmatrix} \tag{2}$$

where $g_{\mathrm{c}}(x+0, y+0)$ is the response value for g_{c}, and $\varsigma = 1, 2, 3, \ldots, 7$. The kernel $\mathbf{K}$ was applied to the noisy image $\mathbf{I}_{\mathrm{c}}$ utilizing convolution, producing the denoised image $\hat{\mathbf{I}}$. Mathematically, this transformation is expressed as $\hat{\mathbf{I}} = \left[\hat{g}(x, y)\right]_{(\lambda-2\varsigma)^2}$, where $x, (y) = 0, 1, 2, \ldots, \lambda-\varsigma, (\lambda-\varsigma)$, and the denoised intensity value, $\hat{g} \in \frac{1}{255}\{0, 1, 2, \ldots, 255\}$ is scaled within the range $[0, 255]$. In particular, the pixel intensity $\hat{g}(x+0, y+0)$ is estimated utilizing contraharmonic mean, quartile-based filters, and maximum filtering, as formulated in Eqs. (3)-(4).

$$\hat{g}(x, y) = \frac{\sum_{i_0=-\varsigma}^{\varsigma} \sum_{j_0=-\varsigma}^{\varsigma} \left[g(x-i_0, y-j_0)\right]^{\ddot{\tau}+1}}{\sum_{i_0=-\varsigma}^{\varsigma} \sum_{j_0=-\varsigma}^{\varsigma} \left[g(x-i_0, y-j_0)\right]^{\ddot{\tau}}} \tag{3}$$

$$\hat{g}(x+0, y+0) = P_{\tau}\left(\mathrm{vec}(\mathbf{K})\right) \tag{4}$$

where P_{τ} represents the percentile-based denoising function. This approach effectively restores image clarity while preserving critical structural details, ensuring reliable feature extraction for ballistic identification.

4.3.2. *Image pre-processing and feature engineering*

Building upon the methodologies outlined in Section 4.3.1, this study further refines the extracted image dataset by implementing a series of image processing and feature engineering techniques. Specifically, the enhancement, segmentation, and extraction of relevant image characteristics were crucial to ensuring robust feature representation for subsequent analysis. To achieve this, a sequence of image processing operators was applied, including the Laplacian sharpening spatial kernel, the Otsu threshold method, and the Moura and Kitney (1991) least-square circle fitting algorithm. These techniques were systematically employed to highlight and quantify the physical attributes of the center-firing pin impression. Following the enhancement phase, noisy images were subjected to a denoising process utilizing appropriate denoising spatial kernels. Subsequently, the **ROI** segmentation was performed, followed by feature engineering, which involved feature creation, extraction, and selection. The feature selection phase utilized stepwise selection techniques and correlation analysis to retain the most informative attributes. The chronological procedure for image processing, denoising, segmentation, and feature engineering is outlined as follows:

Step 1: Input the image $\mathbf{I}_c$ into R statistical software.

Step 2: Enhance image edges utilizing the Laplacian sharpening spatial kernel resulting in an enhanced image, $\mathbf{I}_{\text{CLSK}} = \left[g_{\text{CLSK}}(x, y) \right]_{(\lambda - 2\varsigma - 2)^2}$, where $g_{\text{CLSK}}(x, y)$ is computed based on Eq. (5).

$$g_{\text{CLSK}}(x, y) = \left\| 255 \operatorname{vec}'(\mathbf{K}_{\text{CLSK}}) \operatorname{vec}(\mathbf{K}_{\text{CLSK}}) \right\| \tag{5}$$

where $\mathbf{K}_{\text{CLSK}} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$ represents the Laplacian kernel coefficient.

Step 3: Normalize the image intensity, $g_{\text{CLSK}}(x, y)$ utilizing Eq. (6) to ensure that $g_{\text{CN}}(x, y) \in [0, 1]$ is compressed within the range $[0, 1]$.

$$g_{\text{CN}}(x, y) = \frac{1}{255} \left\lfloor 255 \cdot \frac{g_{\text{CLSK}}(x, y) - \min\limits_{x,y \in \mathbf{I}_{\text{CLSK}}} \left(g_{\text{CLSK}}(x, y) \right)}{\max\limits_{x,y \in \mathbf{I}_{\text{CLSK}}} \left(g_{\text{CLSK}}(x, y) \right) - \min\limits_{x,y \in \mathbf{I}_{\text{CLSK}}} \left(g_{\text{CLSK}}(x, y) \right)} \right\rfloor \tag{6}$$

This results in the normalized image, $\mathbf{I}_{\text{CN}} = \left[g_{\text{CN}}(x, y) \right]_{(\lambda - 2\varsigma - 2)^2}$.

Step 4: Binarize the image $\mathbf{I}_{\text{CN}}$ utilizing the Otsu threshold method, producing a binary image, $\mathbf{I}_{\text{CB}} = \left[g_{\text{CB}}(x, y) \right]_{(\lambda - 2\varsigma - 2)^2}$ with $g_{\text{CB}} = \begin{cases} 1, & g_{\text{CN}} \geq \tilde{t}_{\text{EB}} \\ 0, & g_{\text{CN}} < \tilde{t}_{\text{EB}} \end{cases}$. The threshold $\tilde{t}_{\text{EB}}$ is determined by minimizing within-class variance as expressed in Eq. (7).

$$\tilde{t}_{\text{EB}} = \frac{1}{255} \arg\min_{t} \left(\mu_t \omega_t + \mu_{255-t} \omega_{255-t} \right) \tag{7}$$

where $\mu_t = \sum_{i=0}^{t} 255 i \varphi_{\text{CN}}(i)$, $\mu_{255-t} = \sum_{i=t+1}^{255} 255 i \varphi_{\text{CN}}(i)$,

$\omega_t = (H - 2\varsigma - 2)^2 \sum_{i=0}^{t} \varphi_{\text{CN}}(i) \log\left(\frac{\mu_t}{i} \right)$, and

$\omega_{255-t} = (H - 2\varsigma - 2)^2 \sum_{i=t+1}^{255} \varphi_{\text{CN}}(i) \log\left(\frac{\mu_{255-t}}{i} \right)$.

Step 5: Estimate the geometric properties (A and *r*) of the center-firing pin impressions (Figure 1-left) utilizing Moura and Kitney's least square circle fitting algorithm, given as Eqs. (8)-(9)

$$\mathrm{A}=\begin{bmatrix} 2\left(\lambda\alpha_{20}-\alpha_{00}\alpha_{10}^{2}\right) & 2\left(\lambda\alpha_{11}-\alpha_{10}\alpha_{01}\right) \\ 2\left(\lambda\alpha_{11}-\alpha_{10}\alpha_{01}\right) & 2\left(\lambda\alpha_{02}-\alpha_{00}\alpha_{01}^{2}\right) \end{bmatrix}^{-1} \times \begin{bmatrix} \lambda\left(\alpha_{30}+\alpha_{12}\right)-\alpha_{10}\left(\alpha_{20}+\alpha_{02}\right) \\ \lambda\left(\alpha_{03}+\alpha_{21}\right)-\alpha_{01}\left(\alpha_{20}+\alpha_{02}\right) \end{bmatrix} \tag{8}$$

$$r=\sqrt{\frac{\sum_{x,y\in\mathbf{I}_{\mathrm{CB}}}\left(\left(x_{\mathrm{A}}-x\right)^{2}+\left(y_{\mathrm{A}}-y\right)^{2}\right)}{\left(H-2\varsigma-2\right)^{2}}} \tag{9}$$

where $\lambda=\sum_{x,y\in\mathbf{I}_{\mathrm{CB}}} g_{\mathrm{CB}}\left(x,y\right)$, and $\alpha_{pq}=\sum_{x,y\in\mathbf{I}_{\mathrm{CB}}} x^{p}y^{q}g_{\mathrm{CB}}\left(x,y\right)$.

Step 6: Denoise the images utilizing appropriate spatial kernels (Table 1), resulting in $\hat{\mathbf{I}}$.

Step 7: Segmented the **ROI**, extracting $\mathbf{ROI}=\left[\hat{g}_{\mathrm{D}}\left(x,y\right)\right]_{\left(r/2-1\right)^{2}}$ based on the estimated A and *r*.

Step 8: Extract features such as OLMI_{10}, OLMI_{11}, OLMI_{02}, OLMI_{20}, OLMI_{12}, OLMI_{21}, and OLMI_{22} from the **ROI** utilizing $\left(u+v\right)$th order of the OLMI framework as expressed in Eq. (10).

$$\mathrm{OLMI}_{uv}=\frac{\left(2u+1\right)\left(2v+1\right)}{4}\sum_{a=0}^{u}\sum_{b=0}^{v}\psi_{ua}\psi_{vb}m_{ab} \tag{10}$$

where $m_{ab}=\sum_{x=0}^{r/2-1}\sum_{y=0}^{r/2-1}\left(\frac{2x}{r/2}-1\right)^{u}\left(\frac{2y}{r/2}-1\right)^{v}\hat{g}\left(x,y\right)$ represents the geometric moments, and, ψ_{ua}, ψ_{vb} are Legendre polynomial coefficients defined in Eq. (11).

$$\psi_{ua}=\begin{cases} \dfrac{\left(-1\right)^{(u-a)/2}\left(u+a\right)!}{2^{u}a!\left(\dfrac{u-a}{2}\right)!\left(\dfrac{u+a}{2}\right)!}, & \left(u-a\right)\text{ even} \\ 0, & \left(u-a\right)\text{ odds} \end{cases} \tag{11}$$

This study selected the Laplacian sharpening spatial kernel due to its superior performance compared to Sobel, Canny, and Marr-Hildreth in a similar environment. Notably, variations in Laplacian kernel coefficients did not affect the identification accuracy rate in controlled conditions. For binarization, the Otsu threshold method outperformed entropy-based methods such as cross-entropy, Shannon entropy, and Tsallis entropy (Chuan *et al*. 2013b; Chuan 2014). Since the modified Otsu threshold method yielded similar identification accuracy, it was not selected for processing noisy images (Chuan 2014).

In segmentation, Moura and Kitney's algorithm was preferred over Albano's (1974) conic arcs fitting algorithm, which performed worse. The CHT was avoided due to its high computational cost. This study also compared square-window and circular-window **ROI**s, selecting the square-window **ROI** at $\left(r/2\right)^2$ and $\left(r/4\right)^2$ as literature supports utilizing $\left(r/2\right)^2$ (Ghani *et al*. 2010). The square-window **ROI** offered advantages in computational cost, execution time, pixel intensity distribution, and identification accuracy (Chuan *et al*. 2013b, 2017; Chuan 2014; Liong *et al*. 2020). In contrast, the circular-window **ROI**, due to square pixel shapes, did not accurately capture circular structures and frequently had lower identification accuracy.

4.4. *Modeling*

The primary objective of the modeling phase in this study is to identify pistol types based on the selected feature set, with OLMIs being selected due to their invariant properties concerning TRS. Given that the response variable is categorical with five classes, this study employs the FLDA algorithm. which has been widely validated in ballistics forensics literature (Ghani *et al*. 2010; Chuan *et al*. 2013a, b, 2023; Chuan 2014; Liong *et al*. 2020). Although ANN-based architectures, including CNNs, have gained popularity and demonstrated strong performance in various studies, they were not adopted in this study. This is primarily due to their limited interpretability stemming from the complexity of network design involving multiple hidden layers, numerous nodes, and various activation functions, as well as their high computational demands and extended execution time.

Let N be the total number of images collected from F pistols, expressed as $N = \sum_{f=1}^{F} N_f$, with each image represented by a 7-dimensional feature vector $\mathbf{z}$ extracted from the **ROI**. The feature vector extracted from the pistol f is denoted as $\mathbf{z}_{fg}$, where $g = 1, 2, 3, \ldots, N_f$. The identification process is performed by discriminating the image into pistol f utilizing the minimum Mahalanobis distance between the selected OLMI features vector $\mathbf{z}$ and the class centroid $\bar{\mathbf{Z}}_f$, formulated as Eq. (12).

$$R_f = \arg\min_f \left(\sum_{h=1}^{H} \left(\mathbf{e}_h' \left(\mathbf{z} - \bar{\mathbf{Z}}_f \right) \right)^2 \right) \tag{12}$$

where $\mathbf{e}_h$ represents the H eigenvectors of $\mathbf{W}^{-1}\mathbf{B}$ with $h \le \min(H, 7)$. The matrices $\mathbf{W}$ and $\mathbf{B}$ denote the within-classes and between-classes variance-covariance matrices, respectively. Mathematically, $\mathbf{W}$, $\mathbf{B}$, $\bar{\mathbf{Z}}_f$, and $\bar{\mathbf{Z}}$ denote as Eqs. (13)-(16), respectively.

$$\mathbf{W} = \sum_{f=1}^{F} \sum_{g=1}^{N_f} \left(\mathbf{z}_{fg} - \bar{\mathbf{Z}}_f \right) \left(\mathbf{z}_{fg} - \bar{\mathbf{Z}}_f \right)' \tag{13}$$

$$\mathbf{B} = \sum_{f=1}^{F} \left(\bar{\mathbf{Z}}_f - \bar{\mathbf{Z}} \right) \left(\bar{\mathbf{Z}}_f - \bar{\mathbf{Z}} \right)' \tag{14}$$

$$\bar{\mathbf{Z}}_f = \frac{1}{N_f} \sum_{g=1}^{N_f} \mathbf{z}_{fg} \tag{15}$$

$$\bar{\mathbf{Z}} = \frac{\sum_{f=1}^{F} \sum_{g=1}^{N_f} \mathbf{z}_{fg}}{\sum_{f=1}^{F} N_f} \tag{16}$$

4.5. *Evaluation and deployment*

The principal objective of the evaluation phase was to assess the effectiveness and robustness of the FLDA-based ballistic identification algorithm. To achieve this, an FLDA algorithm was implemented utilizing the Statistical Package for Social Sciences (SPSS), where identification accuracy rates were computed based on the true identification rates derived from confusion matrices. Since this study focused on five classes of pistols, the identification accuracy rates were averaged across these classes to provide a comprehensive assessment. Additionally, the stability of the FLDA-based ballistic identification algorithm was examined by employing ten distinct random seeds, each assumed to represent a unique set of noisy image scenarios. To determine whether the identification accuracy remained statistically significant under different noise levels, the Wilcoxon signed-rank test was conducted.

In defining robustness, this study established a criterion requiring the average identification accuracy rates across ten simulated scenarios to exceed 90%. Furthermore, the presence of statistically significant differences between identification accuracy rates with and without utilizing a denoising spatial kernel was considered indicative of the algorithm's robustness. This novel evaluation approach not merely assessed the algorithm's generalization performance but also provided insights into its stability across diverse noisy scenarios. However, specificity, sensitivity, and F1 scores were not considered due to the inherent difficulty in establishing thresholds for decision-making based on these measurements.

The potential superiority and authentication of the FLDA-based ballistic identification algorithm in the evaluation phase could enable its development as a mobile application. The utilization of a smartphone camera sensor, while less sophisticated than a dedicated ballistic identification system, may introduce noise during image acquisition, potentially degrading image quality. However, the development of the mobile application was not pursued at this stage, as further authentication within a forensic ballistics laboratory was deemed necessary. Instead, this study has been deployed as a research article to obtain constructive feedback from experts in the field, particularly those with strong mathematics and statistics backgrounds.

5. Analysis Results and Discussion

This section presents the analysis results obtained utilizing three distinct software tools, as detailed in Section 4. These included the CONDOR ballistic identification system, utilized for acquiring and storing center-firing pin images, R statistical software, employed for data preparation, and SPSS, utilized for modeling. Notably, this study deviated from the CRISP-DM framework outlined in Section 4, particularly in the Business Understanding, Data Understanding, and Data Preparation phases, as these primarily involved pre-processing acquired images. Instead, the focus was placed on the modeling and evaluation phases.

The following subsections present identification accuracy rates across different noise types, denoising spatial kernels, and segmentation square-window sizes, supported by statistical hypothesis testing. The $\left(r/2\right)^2$ square-window size serves as a benchmark, while the optimum segmented square-window size proposed by the FLDA-based ballistic identification algorithm is $\left(r/4\right)^2$. The study explored various square-window sizes, including $(r)^2$, $\left(r/2\right)^2$, $\left(r/4\right)^2$, $\left(r/6\right)^2$, $\left(r/8\right)^2$, and $\left(r/10\right)^2$.

In addition, the analysis further highlights that ballistic identification accuracy is significantly affected by several OLMIs, particularly $OLMI_{10}$, $OLMI_{11}$, $OLMI_{02}$, $OLMI_{20}$, $OLMI_{12}$, $OLMI_{21}$, and $OLMI_{22}$. These moments reflect essential statistical characteristics of the pixel intensity within the **ROI**. For instance, $OLMI_{10}$ characterizes the center of mass of the pixel intensity along the *x*-axis, while $OLMI_{11}$ captures the moment of inertia along the diagonal axis. $OLMI_{02}$ and $OLMI_{20}$ describe the moment of inertia along with *y*-axis and *x*-axis, respectively. $OLMI_{12}$ and $OLMI_{21}$ quantify the degree of asymmetry in the pixel intensity distribution along the *y*- and *x*-axes, and $OLMI_{22}$ represents the tailedness or kurtosis-like behavior of the pixel intensity distribution across both axes. These interpretations provide a meaningful link between the extracted features and the visual characteristics of the firing pin impression, reinforcing the transparency of the algorithm. These interpretations underscore the clarity and interpretability of the FLDA-based ballistic identification algorithm, attributes that are frequently obscured in complex CNN-based architectures, particularly when applied to limited and noisy datasets such as those encountered in real-world forensic scenarios.

5.1. *Pepper noise*

The simulation results presented in Table 2 assess the robustness of the FLDA-based ballistic identification algorithm under pepper noise contamination, specifically when applying a median denoising spatial kernel. Notably, identification accuracy rates were unavailable in cases where mathematical constraints prevented feature extraction and identification, as indicated by the "-" symbol. This limitation arose in the contraharmonic mean denoising spatial kernel, where the zero denominators in Eq. (3) constructed normalization, OLMI feature extraction, and FLDA algorithm implementation. Similarly, the minimum-ranked ordered, median, and maximum-ranked ordered denoising spatial kernels resulted in zero-extracted OLMI values, making the FLDA algorithm infeasible. These findings highlight critical operational constraints in noise-affected environments, particularly for methods reliant on precise feature extraction.

Despite these challenges, the FLDA-based ballistic identification algorithm demonstrated remarkable resilience against pepper noise, maintaining identification accuracy rates above 90% for $\theta \leq 0.7$ even without a denoising spatial kernel, provided the segmentation size of **ROI** was

$\left(r/2\right)^2$. However, as the noise level increased to $\theta \leq 0.9$, a smaller segmentation size of $\left(r/4\right)^2$ proved more effective in preserving identification accuracy, likely by reducing noise interference at the pixel level and enhancing feature extraction fidelity.

Interestingly, the findings suggest that denoising spatial kernels may not frequently be necessary for managing pepper noise, as the algorithm demonstrated intrinsic robustness under moderate noise conditions. However, for higher noise levels, selective denoising techniques further refined identification accuracy while balancing computation efficiency. Among these methods, this study concludes that the maximum-ranked order denoising spatial kernel with $\varsigma = 2$, combined with an **ROI** size of $\left(r/4\right)^2$, is the most effective approach. This configuration not merely maintained high identification accuracy but also minimized computation cost and execution time, requiring the fewest resources compared to other denoising spatial kernels.

These findings underscore the importance of strategic parameter selection in noise-affected ballistic identification, demonstrating that **ROI** segmentation and denoising kernel selection must be optimized simultaneously to achieve the best balance of accuracy, efficiency, and computational feasibility.

Table 2: Enhancing identification accuracy: Impact of denoising spatial kernels and segmentation window size on pepper noise levels

Kernel	r	ς	**Identification accuracy rates (%) at noise level of θ (standard deviation (%))**								
			0.1	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**	**0.8**	**0.9**
Without kernel	$r/2$	-	94.32 (0.37)	94.32 (0.41)	94.44 (0.58)	94.20 (0.66)	93.84 (0.69)	92.56 (1.00)	92.08 (1.11)	89.69 (1.16)	83.12 (1.84)
Contraharmonic Mean ($l = 0.5$)	$r/2$	1	**94.84 (0.23)**	-	-	-	-	-	-	-	-
		2	**94.92 (0.19)**	**94.96 (0.21)**	**95.12 (0.17)**	**95.08 (0.27)**	-	-	-	-	-
		3	**95.08 (0.33)**	**95.04 (0.21)**	**95.16 (0.13)**	**95.16 (0.13)**	**94.88 (0.32)**	**94.76 (0.13)**	-	-	-
		4	**95.20 (0.19)**	**95.20 (0.00)**	**95.08 (0.19)**	**95.12 (0.17)**	**94.84 (0.40)**	**94.56 (0.39)**	**94.52 (0.27)**	-	-
		5	**95.16 (0.23)**	**95.20 (0.00)**	**95.20 (0.19)**	**95.16 (0.13)**	**94.68 (0.19)**	**94.88 (0.32)**	**94.64 (0.34)**	**94.80 (0.38)**	-
		6	**95.44 (0.34)**	**95.32 (0.27)**	**95.20 (0.00)**	**95.20 (0.00)**	**95.00 (0.43)**	**94.96 (0.21)**	**94.80 (0.33)**	**95.12 (0.37)**	-
		7	**95.56 (0.40)**	**95.52 (0.32)**	**95.36 (0.21)**	**95.24 (0.13)**	**95.12 (0.53)**	**95.08 (0.19)**	**95.00 (0.21)**	**94.88 (0.45)**	**95.08 (0.53)**
Contraharmonic Mean ($l = 1.0$)	$r/2$	1	**94.92 (0.46)**	-	-	-	-	-	-	-	-
		2	**94.84 (0.31)**	**94.88 (0.13)**	**95.12 (0.13)**	**95.12 (0.18)**	-	-	-	-	-
		3	**94.96 (0.43)**	**95.04 (0.21)**	**95.16 (0.13)**	**95.12 (0.17)**	**94.76 (0.35)**	**94.92 (0.19)**	-	-	-
		4	**95.00 (0.43)**	**95.28 (0.17)**	**95.28 (0.17)**	**95.16 (0.13)**	**94.84 (0.55)**	**95.04 (0.21)**	**94.84 (0.30)**	-	-
		5	**95.16 (0.40)**	**95.28 (0.32)**	**95.36 (0.21)**	**95.24 (0.23)**	**95.08 (0.46)**	**95.04 (0.21)**	**94.84 (0.30)**	**94.88 (0.37)**	-
		6	**95.16 (0.35)**	**95.52 (0.17)**	**95.40 (0.21)**	**95.40 (0.28)**	**95.44 (0.21)**	**95.08 (0.27)**	**95.20 (0.27)**	**94.88 (0.41)**	-
		7	**95.48 (0.19)**	**95.60 (0.27)**	**95.48 (0.19)**	**95.44 (0.28)**	**95.20 (0.27)**	**95.08 (0.38)**	**95.24 (0.30)**	**95.04 (0.60)**	**94.60 (0.69)**
Contraharmonic Mean ($l = 1.5$)	$r/2$	1	**95.12 (0.25)**	-	-	-	-	-	-	-	-
		2	**94.80 (0.33)**	**94.88 (0.17)**	**95.12 (0.17)**	**95.12 (0.17)**	-	-	-	-	-
		3	**95.20 (0.33)**	**95.12 (0.17)**	**95.20 (0.00)**	**95.12 (0.17)**	**95.12 (0.32)**	**94.76 (0.23)**	-	-	-
		4	**94.88 (0.25)**	**95.24 (0.13)**	**95.36 (0.28)**	**95.24 (0.30)**	**95.12 (0.45)**	**94.92 (0.19)**	**94.68 (0.38)**	-	-

Table 2 (Continued)

		5	**95.04 (0.42)**	**95.40 (0.28)**	**95.36 (0.28)**	**95.40 (0.54)**	**95.12 (0.32)**	**94.92 (0.38)**	**94.88 (0.32)**	**94.72 (0.49)**	-
		6	**95.36 (0.39)**	**95.32 (0.27)**	**95.28 (0.32)**	**95.24 (0.30)**	**95.24 (0.48)**	**94.72 (0.41)**	**94.68 (0.46)**	**94.92 (0.38)**	-
		7	**95.32 (0.33)**	**95.40 (0.21)**	**95.24 (0.30)**	**95.60 (0.33)**	**94.68 (0.27)**	**94.88 (0.49)**	**94.80 (0.38)**	**94.48 (0.49)**	**94.68 (0.65)**
Median	$r/2$	1	94.60 (0.63)	94.36 (0.51)	94.04 (0.97)	90.00 (0.96)	82.88 (0.90)	73.00 (2.04)	58.68 (2.65)	44.52 (2.58)	30.32 (2.67)
		2	94.60 (0.43)	94.04 (0.67)	93.48 (0.65)	88.40 (2.17)	70.52 (1.27)	51.44 (3.02)	36.12 (2.03)	28.60 (1.31)	-
		3	94.40 (0.65)	93.24 (0.44)	93.40 (0.51)	88.52 (1.27)	59.44 (2.49)	38.68 (3.86)	29.08 (2.13)	21.96 (0.74)	-
		4	94.44 (0.35)	93.44 (0.47)	93.32 (0.76)	90.28 (1.28)	53.40 (2.10)	31.84 (2.74)	25.60 (1.89)	-	-
		5	93.76 (0.63)	93.40 (0.54)	93.08 (0.46)	92.12 (1.10)	49.16 (3.76)	30.24 (2.95)	22.24 (0.80)	-	-
		6	94.00 (0.53)	93.20 (0.42)	93.08 (0.73)	91.72 (1.00)	45.16 (2.69)	28.72 (2.88)	-	-	-
		7	93.88 (0.42)	93.16 (0.44)	93.12 (0.67)	92.32 (0.88)	43.24 (2.19)	26.48 (2.94)	-	-	-
Q_3	$r/2$	1	**94.72 (0.45)**	**94.92 (0.19)**	**94.88 (0.41)**	94.60 (0.34)	93.92 (1.06)	91.96 (1.21)	85.92 (1.10)	72.96 (2.67)	51.96 (2.40)
		2	**94.88 (0.42)**	**94.80 (0.28)**	**95.00 (0.45)**	94.80 (0.37)	94.32 (0.85)	92.76 (1.09)	80.04 (1.34)	56.04 (3.60)	34.12 (3.20)
		3	**95.04 (0.54)**	**95.00 (0.21)**	94.84 (0.40)	**95.12 (0.59)**	94.24 (0.63)	92.68 (0.78)	76.76 (2.15)	44.16 (2.68)	25.20 (1.79)
		4	**95.00 (0.47)**	**94.96 (0.28)**	**94.96 (0.28)**	**95.28 (0.37)**	94.24 (0.63)	93.08 (0.68)	76.48 (2.86)	37.88 (3.33)	25.24 (1.75)
		5	**95.32 (0.50)**	**95.00 (0.43)**	**94.88 (0.32)**	94.64 (0.51)	93.92 (0.59)	93.12 (0.56)	75.84 (2.30)	34.24 (2.87)	-
		6	**95.32 (0.33)**	**94.96 (0.39)**	94.76 (0.44)	94.56 (0.21)	94.08 (0.75)	93.28 (0.59)	78.60 (2.02)	30.60 (1.81)	-
		7	**95.20 (0.53)**	94.80 (0.42)	94.72 (0.53)	94.40 (0.42)	93.72 (0.73)	92.88 (0.86)	80.00 (1.53)	30.36 (2.49)	-
Maximum	$r/2$	1	94.60 (0.28)	**94.48 (0.41)**	94.60 (0.28)	**94.76 (0.30)**	94.88 (0.25)	94.52 (0.27)	**94.32 (0.37)**	**93.60 (0.38)**	87.92 (1.65)
		2	93.76 (0.47)	93.56 (0.58)	93.52 (0.45)	93.76 (0.47)	94.12 (0.33)	93.88 (0.27)	**94.04 (0.44)**	**93.68 (0.41)**	**93.40 (1.02)**
		3	93.24 (0.61)	93.40 (0.57)	93.48 (0.60)	93.72 (0.27)	93.52 (0.56)	**93.36 (0.43)**	**93.36 (0.51)**	**93.28 (0.53)**	**93.28 (1.03)**
		4	92.84 (0.23)	93.20 (0.46)	93.48 (0.57)	93.20 (0.42)	93.28 (0.70)	93.08 (0.57)	**93.20 (0.60)**	**93.20 (0.82)**	**93.52 (0.90)**
		5	92.48 (0.59)	92.88 (0.62)	93.24 (0.87)	93.04 (0.57)	93.08 (0.80)	93.20 (0.90)	93.12 (0.98)	**93.28 (0.56)**	**92.64 (0.87)**
		6	92.12 (0.57)	92.48 (0.41)	93.04 (0.74)	93.20 (0.75)	92.72 (1.25)	93.48 (0.94)	**92.88 (0.80)**	**92.76 (1.14)**	**91.84 (0.93)**
		7	91.88 (0.68)	92.72 (0.56)	92.64 (0.85)	92.96 (1.00)	92.36 (1.09)	92.52 (1.02)	92.44 (1.01)	**91.88 (0.50)**	**91.44 (0.89)**
Without kernel	$r/4$	-	94.16 (0.54)	94.40 (0.63)	93.88 (0.46)	94.32 (0.56)	93.92 (0.65)	93.88 (0.80)	93.68 (0.67)	92.80 (1.32)	90.88 (1.33)
Contraharmonic Mean ($l = 0.5$)	$r/4$	1	94.04 (0.48)	-	-	-	-	-	-	-	-
		2	94.04 (0.76)	94.16 (0.47)	94.20 (0.28)	94.40 (0.38)	-	-	-	-	-
		3	94.56 (0.39)	94.88 (0.41)	**94.32 (0.32)**	94.32 (0.32)	**94.64 (0.34)**	**94.92 (0.19)**	-	-	-
		4	94.52 (0.60)	94.68 (0.53)	**94.64 (0.28)**	94.68 (0.19)	**94.48 (0.25)**	**94.96 (0.34)**	**94.84 (0.23)**	-	-
		5	94.68 (0.65)	**95.00 (0.47)**	**94.76 (0.23)**	**94.76 (0.23)**	**94.84 (0.35)**	**95.00 (0.21)**	**94.76 (0.13)**	**94.88 (0.25)**	-
		6	**94.96 (0.34)**	**94.96 (0.51)**	**94.64 (0.43)**	94.76 (0.40)	**94.96 (0.39)**	**95.12 (0.25)**	**94.88 (0.17)**	**94.76 (0.48)**	-
		7	**94.80 (0.46)**	94.68 (0.46)	**94.72 (0.32)**	**94.88 (0.25)**	**94.80 (0.46)**	**95.16 (0.40)**	**95.40 (0.34)**	**95.08 (0.19)**	**95.28 (0.41)**
Contraharmonic Mean ($l = 1.0$)	$r/4$	1	94.00 (0.63)	-	-	-	-	-	-	-	-
		2	94.56 (0.51)	94.48 (0.32)	**94.24 (0.28)**	94.28 (0.42)	-	-	-	-	-
		3	**94.76 (0.30)**	94.72 (0.53)	**94.72 (0.37)**	94.44 (0.40)	**94.92 (0.27)**	**94.84 (0.23)**	-	-	-
		4	**94.80 (0.33)**	94.64 (0.34)	**94.76 (0.35)**	94.72 (0.25)	**94.88 (0.25)**	**94.84 (0.23)**	**94.96 (0.21)**	-	-
		5	**94.76 (0.40)**	**94.92 (0.38)**	**94.84 (0.35)**	**94.84 (0.35)**	**94.80 (0.42)**	**94.84 (0.23)**	**94.84 (0.24)**	**94.96 (0.34)**	-
		6	**94.96 (0.63)**	94.84 (0.30)	**94.84 (0.44)**	94.68 (0.42)	**94.96 (0.39)**	**95.00 (0.28)**	**94.80 (0.63)**	**95.12 (0.49)**	-

Table 2 (Continued)

		7	**95.04 (0.51)**	**94.92 (0.38)**	**94.68 (0.42)**	94.68 (0.42)	**95.00 (0.39)**	**95.28 (0.49)**	**95.32 (0.38)**	**95.08 (0.42)**	**95.36 (0.57)**
Contraharmonic Mean ($l=1.5$)	$r/4$	1	94.40 (0.46)	-	-	-	-	-	-	-	-
		2	94.44 (0.55)	94.68 (0.19)	**94.60 (0.47)**	94.56 (0.28)	-	-	-	-	-
		3	**94.80 (0.42)**	94.72 (0.41)	**94.56 (0.28)**	94.68 (0.46)	**94.76 (0.30)**	**94.84 (0.30)**	-	-	-
		4	**94.92 (0.33)**	94.92 (0.50)	**94.92 (0.33)**	**94.92 (0.27)**	**95.04 (0.51)**	**95.12 (0.17)**	**95.08 (0.27)**	-	-
		5	**94.72 (0.37)**	**94.96 (0.39)**	**94.64 (0.34)**	**94.88 (0.32)**	**95.12 (0.45)**	**95.04 (0.43)**	**94.88 (0.41)**	**95.00 (0.47)**	-
		6	**94.96 (0.43)**	94.76 (0.35)	**94.84 (0.40)**	**94.84 (0.40)**	**94.92 (0.57)**	**95.28 (0.37)**	**94.96 (0.60)**	**94.88 (0.84)**	-
		7	**95.00 (0.39)**	**95.00 (0.28)**	**94.96 (0.28)**	**95.08 (0.33)**	**95.24 (0.35)**	**95.48 (0.33)**	**95.40 (0.74)**	**95.08 (0.84)**	**95.60 (0.60)**
Median	$r/4$	1	93.84 (0.43)	94.24 (0.74)	94.04 (0.72)	92.96 (1.04)	92.04 (1.63)	86.96 (2.13)	72.12 (2.00)	50.44 (2.91)	31.68 (2.48)
		2	93.76 (0.76)	94.12 (0.33)	**94.32 (0.59)**	93.04 (1.21)	85.24 (1.95)	62.36 (2.30)	37.92 (1.36)	26.88 (1.66)	-
		3	93.76 (0.69)	94.56 (0.43)	**95.24 (0.72)**	94.44 (0.74)	74.89 (2.32)	43.80 (2.06)	29.20 (2.71)	-	-
		4	94.12 (0.50)	94.84 (0.89)	**95.28 (0.73)**	**95.32 (1.13)**	65.12 (3.19)	33.08 (3.89)	24.68 (1.35)	-	-
		5	94.36 (0.55)	94.80 (0.53)	**95.96 (0.55)**	**95.84 (0.76)**	56.28 (3.38)	30.60 (2.64)	-	-	-
		6	94.48 (0.49)	**95.96 (0.67)**	**96.44 (0.55)**	**96.56 (0.95)**	52.58 (3.46)	27.52 (2.39)	-	-	-
		7	**95.24 (0.44)**	**96.32 (0.49)**	**96.76 (0.51)**	**96.88 (0.49)**	45.88 (3.27)	25.64 (1.58)	-	-	-
Q_3	$r/4$	1	94.20 (0.28)	93.96 (0.30)	94.16 (0.43)	94.32 (0.37)	**94.68 (0.63)**	94.36 (0.83)	93.48 (1.27)	85.52 (1.45)	62.64 (1.76)
		2	94.36 (0.51)	94.20 (0.43)	**94.24 (0.21)**	94.32 (0.41)	**94.60 (0.51)**	94.04 (0.95)	90.76 (0.95)	67.52 (4.13)	35.16 (1.53)
		3	94.20 (0.63)	94.20 (0.43)	**94.28 (0.33)**	94.28 (0.27)	**94.68 (0.53)**	**95.16 (0.67)**	88.88 (1.20)	52.28 (3.43)	27.76 (1.98)
		4	94.36 (0.35)	94.12 (0.71)	94.08 (0.41)	94.36 (0.44)	**94.88 (0.53)**	**95.32 (0.46)**	87.56 (2.30)	40.76 (2.72)	22.84 (0.61)
		5	94.28 (0.71)	94.04 (0.61)	94.08 (0.41)	94.16 (0.54)	94.12 (0.60)	**95.40 (0.71)**	88.52 (2.07)	36.02 (3.15)	-
		6	93.52 (0.43)	93.72 (0.46)	94.00 (0.42)	94.24 (0.51)	**95.12 (0.62)**	**95.80 (0.83)**	89.64 (1.42)	31.10 (3.15)	-
		7	94.00 (0.42)	93.88 (0.42)	93.96 (0.40)	94.88 (0.59)	**95.16 (0.40)**	**96.64 (0.47)**	90.28 (1.47)	29.76 (2.86)	-
Maximum	$r/4$	1	94.52 (0.60)	94.24 (0.47)	94.24 (0.34)	94.16 (0.28)	94.20 (0.34)	94.08 (0.37)	94.08 (0.67)	**94.44 (0.51)**	**93.48 (0.76)**
		2	**95.28 (0.77)**	**94.88 (0.32)**	**94.76 (0.55)**	**95.16 (0.55)**	**95.00 (0.34)**	**94.72 (0.75)**	**94.76 (0.40)**	**94.60 (0.34)**	**94.56 (0.57)**
		3	**95.24 (0.58)**	**95.36 (0.43)**	**95.40 (0.28)**	**95.20 (0.38)**	**95.68 (0.37)**	**95.52 (0.49)**	**94.80 (0.65)**	**94.96 (0.66)**	**94.80 (0.60)**
		4	**95.32 (0.33)**	**95.32 (0.42)**	**94.84 (0.51)**	**95.40 (0.43)**	**95.40 (0.54)**	**95.24 (0.64)**	**95.32 (0.68)**	**95.12 (0.41)**	**94.64 (0.69)**
		5	94.48 (0.49)	94.76 (0.35)	**94.72 (0.70)**	94.88 (0.59)	**95.48 (0.42)**	**95.48 (0.50)**	**95.40 (0.83)**	**95.36 (0.80)**	**95.08 (0.53)**
		6	**94.80 (0.73)**	94.76 (0.44)	**95.44 (0.60)**	**95.72 (0.53)**	**95.24 (0.61)**	**95.24 (0.51)**	**95.60 (0.63)**	**96.04 (0.67)**	**95.84 (0.69)**
		7	95.04 (0.54)	**95.56 (0.64)**	**95.72 (0.33)**	**96.04 (0.44)**	**95.72 (0.53)**	**95.64 (0.69)**	**96.12 (0.73)**	**95.72 (0.82)**	**95.88 (1.12)**

***Note:** The bold values indicate a statistically significant difference in identification accuracy rates between the algorithm with and without denoising spatial kernel

5.2. *Salt noise*

Table 3 presents the average identification rates and standard deviations across ten distinct simulated salt noise image scenarios, considering variations in denoising spatial kernels, segmentation window sizes, and kernel window sizes. The analysis reveals the FLDA-based ballistic identification algorithm achieves identification accuracy rates exceeding 90% at noise levels $\theta = 0.2$ and $\theta = 0.4$, when utilizing segmentation square-window sizes of $\left(r/2\right)^2$ and $\left(r/4\right)^2$, respectively, without the application of denoising spatial kernels. However, the results

indicate that the algorithms exhibit reduced robustness in the absence of denoising spatial kernels. Notably, the segmentation square-window size of $\left(r/4\right)^2$ outperforms $\left(r/2\right)^2$, which is consistent with findings reported in previous literature (Chuan *et al.* 2013b).

At higher noise levels, particularly at $\theta = 0.9$, the application of appropriate denoising spatial kernels leads to substantial improvements in identification accuracy, especially when employing a segmentation square-window size of $\left(r/4\right)^2$. The analysis identifies several effective denoising spatial kernels, including the contraharmonic mean denoising spatial kernel with order $l = -0.5$ for $\varsigma = 6$ and 7, as well as the contraharmonic mean denoising spatial kernel with order $l = -1.0$ for $\varsigma = 4, 5, 6$ and 7. Furthermore, the contraharmonic mean denoising spatial kernel with order $l = -1.5$ demonstrates strong performance for $\varsigma = 3, 4, 5, 6,$ and 7, while the minimum-ranked ordered denoising spatial kernel proves effective with for $\varsigma = 3, 4, 5,$ and 6. The findings presented in Table 3 further support these observations.

From a computational perspective, the implementation of a denoising spatial kernel with a smaller window size is advantageous due to its reduced computational cost and shorter execution time compared to larger window sizes. Among the tested denoising approaches, the contraharmonic mean denoising spatial kernel with order $l = -1.5$ and $\varsigma = 3$ emerges as particularly effective in mitigating salt noise. However, a more comprehensive evaluation reveals that the minimum-ranked ordered denoising spatial kernel consistently outperforms other methods, demonstrating superior identification accuracy, particularly at moderate noise levels where $\theta \le 0.5$. This suggests that the minimum-ranked ordered denoising spatial kernel provides the most robust and reliable denoising performance for salt noise conditions.

Table 3: Enhancing identification accuracy: Impact of denoising spatial kernels and segmentation window size on salt noise levels

Kernel	r	ς	**Identification accuracy rates (%) at noise level of θ (standard deviation (%))**								
			0.1	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**	**0.8**	**0.9**
Without kernel	$r/2$	-	93.12 (1.06)	92.64 (2.01)	89.84 (1.43)	86.20 (1.75)	83.24 (1.68)	80.48 (1.61)	74.24 (1.92)	67.16 (1.69)	54.56 (3.58)
Contraharmonic Mean ($l = -0.5$)	$r/2$	1	**94.40 (0.78)**	93.91 (0.71)	**93.60 (0.65)**	**93.64 (0.91)**	**91.04 (1.18)**	89.48 (1.37)	86.36 (2.15)	80.00 (2.56)	69.32 (2.49)
		2	**94.32 (0.56)**	93.88 (0.82)	**93.32 (1.18)**	**93.48 (0.76)**	**91.64 (1.15)**	**90.64 (1.04)**	87.12 (1.47)	82.40 (1.12)	70.76 (2.06)
		3	**94.40 (0.46)**	94.20 (0.34)	**93.76 (0.54)**	**93.24 (0.74)**	**92.40 (2.01)**	**90.84 (0.81)**	88.40 (1.48)	82.68 (2.52)	70.12 (1.79)
		4	**94.52 (0.63)**	94.16 (0.60)	**93.32 (0.92)**	**93.64 (0.74)**	**91.96 (1.58)**	**91.00 (0.93)**	88.16 (2.22)	82.08 (1.86)	71.68 (2.56)
		5	**94.24 (0.51)**	**94.40 (0.90)**	**93.84 (1.25)**	**93.52 (1.51)**	**92.12 (1.54)**	**90.04 (1.17)**	88.96 (1.96)	83.00 (2.67)	72.60 (3.11)
		6	**94.48 (0.73)**	93.96 (0.74)	**93.20 (0.88)**	**93.20 (0.63)**	**92.08 (1.17)**	**90.80 (1.77)**	87.52 (1.56)	83.00 (1.28)	72.40 (2.53)
		7	94.12 (0.57)	94.12 (0.71)	93.96 (0.74)	93.28 (0.82)	91.72 (1.28)	90.40 (1.47)	87.92 (1.48)	83.60 (2.61)	73.60 (1.88)
Contraharmonic Mean ($l = -1.0$)	$r/2$	1	**94.36 (0.55)**	**94.56 (0.39)**	**93.96 (0.64)**	**93.84 (0.57)**	**93.00 (0.98)**	**92.12 (1.27)**	**91.20 (1.98)**	86.84 (1.37)	77.44 (2.51)
		2	**94.32 (0.41)**	**94.28 (0.71)**	**94.00 (0.80)**	**94.20 (0.60)**	**93.52 (1.20)**	**93.24 (0.64)**	**92.32 (1.33)**	88.52 (1.39)	80.12 (1.71)
		3	93.80 (0.57)	94.00 (0.38)	**93.96 (0.89)**	**94.00 (0.75)**	**93.84 (0.76)**	**92.72 (0.80)**	**91.60 (1.19)**	89.84 (1.48)	80.80 (3.02)
		4	93.84 (0.66)	94.04 (0.35)	**94.12 (0.76)**	**94.00 (0.63)**	**93.72 (1.07)**	**93.08 (1.15)**	**92.32 (1.80)**	89.36 (1.15)	81.88 (1.71)
		5	93.72 (0.68)	93.68 (0.45)	**94.24 (0.85)**	**94.20 (0.47)**	**92.96 (0.43)**	**92.88 (1.05)**	**91.72 (1.27)**	89.88 (1.25)	80.24 (2.13)
		6	93.60 (0.53)	93.64 (0.64)	**93.60 (0.53)**	**93.72 (0.68)**	**92.88 (0.98)**	**92.48 (0.88)**	**91.48 (0.80)**	89.44 (0.47)	81.08 (1.00)
		7	93.68 (0.56)	93.80 (0.74)	**93.64 (0.48)**	**93.72 (0.53)**	**93.48 (0.84)**	**93.24 (1.04)**	**91.12 (1.22)**	89.64 (1.97)	82.48 (2.78)

Table 3 (Continued)

Contraharmonic Mean ($l=-1.5$)	$r/2$	1	**94.32** **(0.25)**	94.08 (0.25)	**94.04** **(0.58)**	**94.44** **(0.55)**	**94.52** **(0.80)**	**94.00** **(0.63)**	**92.36** **(0.72)**	88.72 (1.79)	82.20 (3.54)
		2	93.72 (0.38)	93.88 (0.53)	**93.88** **(0.42)**	**94.12** **(0.42)**	**93.72** **(0.96)**	**93.72** **(0.73)**	**92.72** **(0.77)**	**91.84** **(1.43)**	85.44 (1.99)
		3	**93.20** **(0.42)**	93.84 (0.47)	**93.80** **(0.60)**	**93.96** **(0.55)**	**94.12** **(0.53)**	**93.92** **(0.73)**	**93.44** **(0.78)**	**92.44** **(1.01)**	**93.56** **(3.53)**
		4	**93.36** **(0.28)**	93.24 (0.30)	**93.52** **(0.37)**	**93.76** **(0.34)**	**93.88** **(0.76)**	**93.52** **(0.80)**	**92.80** **(0.96)**	**92.24** **(1.09)**	86.96 (2.75)
		5	**93.36** **(0.28)**	93.48 (0.33)	**93.56** **(0.58)**	**93.76** **(0.69)**	**93.40** **(0.63)**	**93.36** **(1.00)**	**93.16** **(0.72)**	**92.52** **(0.87)**	86.40 (1.46)
		6	**93.44** **(0.43)**	93.56 (0.55)	**93.64** **(0.40)**	**93.48** **(0.65)**	**93.36** **(0.95)**	**93.32** **(0.50)**	**92.80** **(0.63)**	**92.08** **(1.16)**	87.20 (1.57)
		7	**93.32** **(0.42)**	93.24 (0.40)	**93.28** **(0.53)**	**93.24** **(0.55)**	**93.48** **(0.73)**	**92.92** **(0.60)**	**91.40** **(1.20)**	**92.76** **(1.51)**	86.56 (1.52)
Minimum	$r/2$	1	93.08 (0.38)	93.00 (0.28)	**93.36** **(0.34)**	**93.04** **(0.34)**	**93.40** **(0.57)**	**92.48** **(0.94)**	**90.84** **(1.29)**	82.68 (2.55)	63.88 (3.67)
		2	92.96 (0.43)	92.84 (0.48)	**92.92** **(0.42)**	**92.92** **(0.73)**	**92.88** **(0.41)**	**92.68** **(0.68)**	**92.72** **(0.59)**	**91.64** **(0.76)**	78.32 (1.94)
		3	91.96 (0.51)	91.72 (0.63)	**91.80** **(0.57)**	**92.52** **(0.60)**	**92.36** **(0.61)**	**92.88** **(0.82)**	**92.28** **(0.65)**	**92.24** **(1.21)**	89.80 (1.28)
		4	88.60 (0.78)	89.04 (0.76)	89.04 (1.15)	89.60 (0.84)	89.96 (0.89)	**90.12** **(1.08)**	**90.84** **(1.02)**	**90.64** **(0.71)**	**91.20** **(1.25)**
		5	86.68 (0.76)	86.92 (0.50)	87.52 (0.80)	87.72 (0.76)	87.60 (0.68)	89.16 (1.34)	89.20 (1.40)	89.16 (1.01)	89.92 (1.89)
		6	86.68 (0.76)	86.36 (1.15)	86.08 (1.14)	86.60 (1.12)	86.20 (1.30)	86.28 (1.18)	86.44 (1.47)	86.88 (1.22)	88.64 (1.36)
		7	86.68 (0.76)	85.24 (1.04)	84.84 (1.17)	85.04 (1.34)	85.44 (1.21)	85.20 (0.57)	84.72 (1.45)	85.32 (1.28)	88.60 (1.24)
Q_1	$r/2$	1	**93.76** **(0.43)**	**94.28** **(0.33)**	**94.28** **(0.42)**	**92.96** **(0.85)**	89.64 (1.35)	81.24 (1.80)	66.72 (3.55)	51.60 (3.19)	39.92 (3.57)
		2	93.36 (0.34)	93.56 (0.30)	**93.88** **(0.38)**	**94.20** **(0.34)**	**94.08** **(0.73)**	85.96 (1.97)	63.80 (2.89)	45.04 (1.96)	33.16 (3.28)
		3	93.36 (0.28)	93.60 (0.42)	**93.72** **(0.19)**	**93.76** **(0.34)**	**94.56** **(0.76)**	**91.80** **(0.81)**	64.96 (1.72)	40.00 (2.90)	30.44 (2.27)
		4	93.40 (0.28)	93.84 (0.28)	**93.84** **(0.34)**	**94.08** **(0.77)**	**93.80** **(0.76)**	**94.16** **(0.78)**	65.00 (1.68)	37.12 (3.25)	30.68 (1.74)
		5	93.44 (0.28)	93.60 (0.27)	**93.72** **(0.60)**	**93.96** **(0.40)**	**94.12** **(0.53)**	**94.24** **(0.89)**	68.08 (2.44)	34.20 (2.21)	27.60 (3.23)
		6	93.48 (0.33)	93.52 (0.45)	**93.48** **(0.53)**	**93.76** **(0.87)**	**94.24** **(0.83)**	**94.20** **(0.66)**	72.24 (1.93)	33.40 (2.71)	27.12 (1.67)
		7	93.24 (0.46)	93.44 (0.34)	**93.20** **(0.50)**	**93.52** **(0.59)**	**94.32** **(0.73)**	**94.36** **(0.67)**	76.36 (1.80)	32.92 (2.39)	26.60 (1.68)
Median	$r/2$	1	**94.88** **(0.32)**	**94.20** **(0.51)**	90.60 (1.23)	80.08 (3.23)	66.04 (2.53)	56.52 (3.30)	44.92 (3.19)	37.88 (2.27)	32.36 (2.09)
		2	**94.80** **(0.27)**	**94.88** **(0.32)**	**93.88** **(0.63)**	81.52 (1.62)	59.04 (4.35)	43.52 (2.02)	35.80 (2.95)	32.80 (2.57)	26.64 (1.80)
		3	**94.52** **(0.42)**	**94.40** **(0.27)**	**95.00** **(0.21)**	84.72 (1.72)	53.60 (3.78)	37.84 (2.87)	34.12 (2.95)	28.08 (1.41)	27.00 (2.38)
		4	**94.36** **(0.35)**	94.44 (0.35)	**94.84** **(0.48)**	88.76 (1.01)	50.68 (1.89)	35.48 (2.04)	32.56 (2.14)	27.44 (1.25)	26.00 (1.47)
		5	94.16 (0.43)	**94.44** **(0.44)**	**95.12** **(0.25)**	**91.48** **(1.02)**	45.84 (4.00)	32.04 (1.90)	30.84 (1.30)	36.72 (0.98)	27.08 (1.39)
		6	**94.28** **(0.27)**	**94.80** **(0.50)**	**95.20** **(0.57)**	**92.72** **(1.01)**	45.88 (4.25)	32.12 (2.62)	30.12 (1.18)	26.96 (0.83)	25.52 (1.88)
		7	**94.56** **(0.39)**	**94.56** **(0.66)**	**95.40** **(0.51)**	**94.12** **(0.87)**	45.60 (3.20)	30.12 (0.23)	30.12 (1.22)	27.24 (0.74)	25.44 (1.82)
Without kernel	$r/4$	-	94.28 (0.53)	94.20 (0.63)	94.20 (0.76)	94.08 (1.10)	93.32 (0.78)	92.08 (1.58)	90.00 (1.26)	86.92 (1.41)	79.60 (2.32)
Contraharmonic Mean ($l=-0.5$)	$r/4$	1	94.48 (0.59)	94.64 (0.47)	94.24 (0.71)	94.44 (0.40)	94.04 (1.01)	93.76 (0.60)	92.84 (0.89)	92.36 (0.97)	87.96 (0.74)
		2	**94.84** **(0.44)**	94.68 (0.27)	94.52 (0.53)	94.52 (0.53)	**94.36** **(0.79)**	**94.28** **(0.68)**	**94.12** **(0.96)**	**93.08** **(1.19)**	89.40 (1.25)
		3	**95.00** **(0.54)**	**95.12** **(0.41)**	94.56 (0.51)	**95.12** **(0.59)**	94.12 (0.82)	**94.64** **(0.76)**	**94.16** **(0.80)**	**92.64** **(1.44)**	89.32 (1.69)
		4	**95.48** **(0.46)**	**95.12** **(0.49)**	**95.20** **(0.96)**	**95.12** **(0.49)**	**94.88** **(1.03)**	**94.48** **(0.98)**	**93.48** **(0.98)**	**93.64** **(1.04)**	89.64 (1.01)
		5	**95.64** **(0.44)**	**95.28** **(0.41)**	**95.00** **(0.39)**	95.00 (0.83)	**94.76** **(0.87)**	**94.60** **(0.98)**	**94.36** **(1.02)**	**92.76** **(1.54)**	89.76 (1.43)
		6	**96.08** **(0.37)**	**95.44** **(0.51)**	**95.48** **(0.87)**	**95.48** **(0.80)**	**94.92** **(0.65)**	**94.56** **(1.02)**	**94.56** **(0.83)**	**93.24** **(1.43)**	**90.52** **(1.33)**
		7	**95.80** **(0.54)**	**96.04** **(0.72)**	**95.48** **(0.50)**	**95.60** **(0.57)**	**95.28** **(0.75)**	**94.64** **(1.20)**	**94.32** **(0.65)**	**93.48** **(1.18)**	**90.16** **(0.93)**

Table 3 (Continued)

Contraharmonic Mean ($l=-1.0$)	$r/4$	1	94.60 (0.43)	94.56 (0.28)	94.60 (0.69)	94.76 (0.40)	**94.80 (0.92)**	**94.48 (0.77)**	**94.68 (0.78)**	**93.40 (1.05)**	**90.04 (0.81)**
		2	**95.04 (0.28)**	**94.84 (0.23)**	94.60 (0.39)	**94.92 (0.38)**	**94.64 (0.66)**	**94.52 (0.65)**	**94.44 (0.67)**	**93.48 (1.03)**	**92.20 (0.91)**
		3	**95.36 (0.28)**	**95.28 (0.37)**	**95.20 (0.78)**	94.88 (0.41)	**94.68 (0.89)**	**94.70 (0.58)**	**94.40 (0.65)**	**94.12 (1.16)**	**91.68 (1.33)**
		4	**95.56 (0.23)**	**95.44 (0.43)**	**95.28 (0.37)**	**95.08 (0.38)**	**94.84 (0.40)**	**94.84 (0.67)**	**94.44 (0.99)**	**93.56 (0.99)**	**92.24 (1.15)**
		5	**95.72 (0.27)**	**95.48 (0.33)**	**95.64 (0.48)**	**95.60 (0.68)**	**95.24 (0.76)**	**95.00 (0.47)**	**94.60 (0.74)**	**94.36 (1.09)**	**92.60 (0.95)**
		6	**96.00 (0.33)**	**96.12 (0.42)**	**95.72 (0.38)**	**95.64 (0.67)**	**95.60 (0.60)**	**94.92 (0.60)**	**94.80 (0.46)**	**94.08 (1.10)**	**92.64 (1.32)**
		7	**96.40 (0.38)**	**96.60 (0.39)**	**96.16 (0.57)**	**95.96 (0.55)**	**95.40 (0.43)**	**95.24 (0.76)**	**95.00 (0.76)**	**93.96 (0.67)**	**92.96 (1.40)**
Contraharmonic Mean ($l=-1.5$)	$r/4$	1	**94.92 (0.27)**	**94.84 (0.40)**	94.52 (0.50)	**94.84 (0.48)**	**94.36 (0.30)**	**94.52 (0.65)**	**94.96 (0.54)**	**93.92 (0.70)**	**91.68 (1.32)**
		2	**94.88 (0.53)**	**95.08 (0.42)**	94.76 (0.40)	94.84 (0.72)	**95.04 (0.71)**	**94.96 (0.78)**	**94.48 (0.88)**	**93.84 (0.66)**	**93.20 (0.98)**
		3	**95.28 (0.25)**	**95.60 (0.33)**	**95.56 (0.48)**	**95.40 (0.54)**	**95.40 (0.60)**	**95.08 (0.42)**	**95.00 (0.51)**	**94.88 (0.80)**	**92.92 (0.96)**
		4	**95.64 (0.30)**	**95.36 (0.34)**	**95.48 (0.42)**	**95.68 (0.53)**	**95.08 (0.73)**	**95.16 (0.74)**	**94.84 (0.61)**	**94.36 (0.72)**	**93.40 (1.09)**
		5	**96.12 (0.38)**	**95.92 (0.25)**	**96.08 (0.41)**	**96.12 (0.53)**	**95.68 (0.37)**	**95.28 (0.45)**	**94.96 (0.66)**	**95.00 (0.51)**	**93.76 (0.91)**
		6	**96.88 (0.25)**	**96.92 (0.42)**	**96.68 (0.42)**	**96.48 (0.49)**	**95.92 (0.67)**	**95.76 (0.57)**	**95.64 (0.48)**	**94.60 (0.76)**	**93.48 (0.68)**
		7	**97.20 (0.19)**	**96.80 (0.27)**	**97.04 (0.34)**	**96.76 (0.40)**	**96.16 (0.57)**	**95.72 (0.53)**	**95.32 (0.65)**	**94.84 (1.15)**	**93.12 (1.14)**
Minimum	$r/4$	1	**94.92 (0.27)**	94.60 (0.54)	94.76 (0.58)	94.56 (0.54)	**94.64 (0.63)**	**94.56 (0.76)**	**94.44 (0.81)**	**92.72 (1.62)**	83.76 (2.17)
		2	**94.84 (0.30)**	**95.16 (0.44)**	**95.08 (0.38)**	94.84 (0.51)	**95.04 (0.51)**	**94.92 (0.84)**	**94.96 (0.76)**	**94.24 (0.57)**	**91.04 (1.58)**
		3	**96.04 (0.13)**	**96.08 (0.17)**	**96.04 (0.30)**	**96.16 (0.34)**	**95.88 (0.50)**	**95.56 (0.61)**	**95.20 (0.65)**	**94.92 (0.53)**	**94.12 (0.84)**
		4	**96.04 (0.13)**	**96.08 (0.25)**	**96.24 (0.34)**	**96.24 (0.43)**	**95.96 (0.64)**	**95.84 (0.54)**	**95.72 (0.65)**	**95.00 (0.93)**	**94.64 (0.74)**
		5	**96.04 (0.40)**	**96.16 (0.28)**	**96.48 (0.37)**	**96.20 (0.51)**	**96.20 (0.63)**	**95.40 (0.78)**	**95.44 (0.89)**	**95.00 (0.76)**	**94.80 (0.71)**
		6	**95.72 (0.38)**	**95.60 (0.27)**	**95.56 (0.44)**	**95.16 (0.67)**	**95.24 (0.40)**	**95.32 (0.38)**	**95.12 (0.45)**	**94.84 (1.02)**	**94.72 (1.08)**
		7	94.60 (0.43)	**94.76 (0.40)**	**95.08 (0.60)**	94.52 (0.65)	**94.92 (0.89)**	**94.80 (0.92)**	**94.40 (0.46)**	**94.28 (1.21)**	**93.92 (1.26)**
Q_1	$r/4$	1	**94.88 (0.25)**	**94.76 (0.35)**	94.48 (0.32)	94.44 (0.51)	93.96 (0.51)	91.88 (0.73)	86.08 (1.99)	77.24 (2.17)	54.68 (3.69)
		2	**94.88 (0.37)**	**94.72 (0.32)**	94.72 (0.56)	**94.92 (0.27)**	**94.40 (0.63)**	93.12 (0.96)	85.80 (2.08)	65.04 (2.27)	37.68 (2.60)
		3	**95.20 (0.33)**	**94.84 (0.55)**	94.80 (0.50)	94.84 (0.48)	**94.16 (0.66)**	**94.20 (0.43)**	85.84 (2.22)	59.24 (4.65)	30.84 (2.92)
		4	**95.96 (0.23)**	**95.56 (0.55)**	**95.12 (0.53)**	**95.12 (0.65)**	**94.88 (0.41)**	**94.44 (0.64)**	87.08 (1.62)	51.00 (3.28)	27.40 (1.69)
		5	**96.36 (0.35)**	**96.32 (0.45)**	**95.68 (0.37)**	**95.72 (0.38)**	**95.28 (0.49)**	**94.28 (0.46)**	87.84 (1.24)	45.72 (2.52)	25.80 (1.62)
		6	**97.12 (0.17)**	**97.08 (0.19)**	**96.16 (0.28)**	**95.20 (0.42)**	**95.36 (0.57)**	**94.36 (0.64)**	89.44 (1.91)	40.28 (3.62)	25.88 (0.94)
		7	**97.20 (0.00)**	**97.16 (0.13)**	**96.92 (0.27)**	**96.08 (0.53)**	**95.28 (0.65)**	**94.20 (0.66)**	90.36 (1.18)	38.56 (3.15)	26.08 (1.32)
Median	$r/4$	1	**94.76 (0.13)**	94.60 (0.43)	93.88 (0.91)	92.68 (1.13)	88.36 (1.90)	79.40 (3.41)	66.40 (2.40)	47.40 (4.01)	29.84 (3.57)
		2	94.68 (0.19)	94.64 (0.28)	94.36 (0.48)	92.52 (0.78)	82.68 (1.64)	64.00 (3.35)	45.12 (3.26)	30.48 (2.38)	25.24 (1.21)
		3	94.72 (0.25)	94.52 (0.19)	94.56 (0.43)	93.84 (0.80)	77.52 (2.32)	52.52 (2.77)	35.16 (2.71)	27.28 (0.82)	25.92 (0.90)
		4	**94.76 (0.23)**	94.52 (0.19)	94.64 (0.51)	94.12 (0.73)	73.88 (1.78)	43.48 (3.14)	32.68 (1.33)	27.76 (0.89)	26.16 (1.25)
		5	**95.24 (0.30)**	94.56 (0.34)	94.24 (0.47)	94.48 (0.59)	71.84 (3.09)	38.08 (3.49)	30.92 (1.05)	27.68 (1.25)	25.56 (0.91)
		6	**95.56 (0.35)**	**94.88 (0.41)**	94.44 (0.64)	94.32 (0.41)	71.88 (2.41)	33.28 (2.30)	30.08 (1.01)	27.76 (0.93)	25.33 (1.16)
		7	**95.32 (0.33)**	**94.88 (0.53)**	94.52 (0.57)	94.64 (0.57)	68.56 (2.69)	30.08 (2.78)	30.84 (0.93)	27.68 (0.73)	25.56 (0.91)

*__Note:__ The bold values indicate a statistically significant difference in identification accuracy rates between the algorithm with and without the denoising spatial kernel.

5.3. *Salt-and-pepper noise*

Building upon the findings from the pepper and salt noise analyses, this section extends the evaluation to salt-and-pepper noise, providing a comprehensive assessment of the FLDA-based ballistic identification algorithm. Table 4 presents identification accuracy rates for images processed utilizing an **ROI** sized at $\left(r/2\right)^2$ and $\left(r/4\right)^2$ The analysis reaffirms the algorithm's robustness, achieving identification rates exceeding 90% across varying noise levels. Specifically, identification surpasses this threshold at $\theta \leq 0.4$ for $\left(r/2\right)^2$ and $\theta \leq 0.6$ for $\left(r/4\right)^2$, further validates the superiority of the smaller **ROI** size in maintaining identification accuracy under noise contamination.

Moreover, as observed in the pepper and salt noise analyses, performance improvements are evident when appropriate denoising spatial kernels are employed. Table 4 highlights that noise levels ranging from $\theta \leq 0.6$ to $\theta \leq 0.8$, the application of denoising techniques significantly enhance identification accuracy. Among the evaluated approaches, the median denoising spatial kernels with $\varsigma = 5$ proves to be the most effective, offering a favorable balance between computational efficiency and noise suppression. Notably, this kernel exhibits a shorter execution time compared to kernels with $\varsigma = 6$ and $\varsigma = 7$, establishing it as the optimal universal denoising technique for mitigating salt-and-pepper noise in this study.

Table 4: Enhancing identification accuracy: Impact of denoising spatial kernels and segmentation window size on salt-and-pepper noise levels

Kernel	r	ς	**Identification accuracy rates (%) at noise level of θ (standard deviation (%))**								
			0.1	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**	**0.8**	**0.9**
Without kernel	$r/2$	-	94.36 (0.97)	93.16 (0.79)	92.44 (1.42)	90.40 (1.57)	85.40 (2.50)	77.20 (2.60)	69.48 (3.05)	55.04 (2.26)	41.56 (3.91)
Median	$r/2$	1	**94.64 (0.21)**	**94.56 (0.51)**	**94.40 (0.68)**	**94.08 (0.77)**	**91.40 (1.05)**	84.68 (1.78)	72.48 (3.87)	58.56 (1.84)	40.48 (3.26)
		2	**94.52 (0.19)**	**94.60 (0.28)**	**94.28 (0.46)**	**94.28 (0.38)**	**94.40 (0.57)**	**92.84 (0.74)**	81.64 (2.49)	63.64 (1.83)	41.28 (2.10)
		3	**94.12 (0.33)**	**94.40 (0.38)**	**94.36 (0.55)**	**94.36 (0.48)**	**94.40 (0.42)**	**93.92 (0.92)**	88.92 (1.52)	66.92 (2.44)	42.24 (3.06)
		4	**93.84 (0.34)**	**93.96 (0.44)**	**94.32 (0.49)**	**94.32 (0.41)**	**94.36 (0.55)**	**94.00 (0.53)**	**92.28 (0.80)**	75.80 (2.55)	43.20 (1.87)
		5	**93.76 (0.34)**	**93.68 (0.32)**	**94.40 (0.46)**	**94.24 (0.54)**	**94.48 (0.62)**	**93.84 (1.05)**	**93.52 (0.75)**	83.40 (1.52)	46.12 (2.32)
		6	**93.84 (0.47)**	**94.32 (0.65)**	**94.44 (0.44)**	**93.80 (0.39)**	**94.28 (0.68)**	**93.96 (0.61)**	**93.36 (0.66)**	88.40 (2.30)	48.20 (2.10)
		7	**94.40 (0.63)**	**94.00 (0.53)**	**94.36 (0.85)**	**94.12 (0.33)**	**94.64 (0.51)**	**94.08 (0.59)**	**93.24 (0.83)**	**90.80 (1.80)**	51.76 (2.55)
Without kernel	$r/4$	-	94.44 (0.30)	94.20 (0.43)	94.32 (0.80)	93.36 (0.80)	92.96 (1.10)	90.84 (0.95)	87.40 (1.94)	77.64 (2.87)	54.36 (3.08)
Median	$r/4$	1	94.52 (0.33)	**94.64 (0.34)**	94.60 (0.28)	**94.40 (0.38)**	**93.96 (0.74)**	**93.48 (1.07)**	90.00 (1.52)	79.00 (2.20)	53.36 (4.00)
		2	94.52 (0.19)	**94.64 (0.34)**	94.64 (0.39)	**94.28 (0.42)**	**94.52 (0.57)**	**94.16 (0.98)**	**92.08 (0.53)**	84.20 (3.13)	57.48 (1.96)
		3	**94.76 (0.30)**	**94.84 (0.30)**	94.64 (0.51)	**94.60 (0.63)**	**94.92 (0.46)**	**94.76 (0.72)**	**93.88 (0.53)**	88.20 (1.52)	59.32 (2.71)
		4	**95.20 (0.50)**	**94.96 (0.47)**	94.52 (0.46)	**94.96 (0.34)**	**94.88 (0.17)**	**94.72 (0.41)**	**94.40 (1.03)**	89.88 (1.49)	60.96 (2.98)
		5	**95.56 (0.35)**	**95.68 (1.37)**	**95.20 (0.53)**	**94.84 (0.69)**	**94.68 (0.63)**	**94.80 (0.68)**	**94.40 (0.73)**	**92.20 (1.86)**	63.12 (2.57)
		6	**95.76 (0.28)**	**95.84 (0.34)**	**95.52 (0.41)**	**95.20 (0.42)**	**95.24 (0.48)**	**95.16 (0.40)**	**94.76 (0.74)**	**92.96 (0.89)**	66.28 (3.60)
		7	**95.56 (0.51)**	**95.40 (0.69)**	**95.48 (0.27)**	**95.12 (0.49)**	**95.08 (0.57)**	**95.28 (1.11)**	**95.08 (0.38)**	**94.12 (1.32)**	67.76 (3.08)

***Note**: The bold values indicate a statistically significant difference in identification accuracy rates between the algorithm with and without denoising spatial kernel.

6. Conclusion and Recommendations for Future Work

This study provides a comprehensive evaluation of statistical computing techniques for forensic firearm identification, specifically assessing the robustness of the FLDA-based ballistic identification algorithm against fixed-value impulse noise. Through the simulation of pepper, salt, and salt-and-pepper noise at varying contamination levels (10% to 90%), this research systematically examines the algorithm's performance when paired with various denoising spatial kernels. The findings reveal that the algorithm maintains high identification accuracy rates $(\geq 90\%)$ for pepper and salt noise when associated with maximum-ranked and minimum-ranked ordered denoising spatial kernels, respectively. For salt-and-pepper noise, a median denoising spatial kernel $\left(\varsigma^2 = 5^2\right)$ proves universally superior, striking a balance between accuracy and computation efficiency. This study further confirms that a segmentation window size of $\left(r/4\right)^2$ enhances identification accuracy, aligning with prior literature. Notably, identification performance deteriorates at noise levels exceeding $\theta = 0.6$, necessitating the utilization of denoising spatial kernels to preserve forensic reliability.

These results highlight the potential of FLDA-based statistical computing techniques in forensic applications by reducing reliance on physical verification and accelerating forensic investigations. This study aligns with the United Nations Sustainable Development Goals (SDGs) by advancing Artificial Intelligence (AI)-driven forensic methodologies (SDG9: Industry, Innovation, and Infrastructure) and strengthening forensic accuracy in criminal investigations (SDG16: Peace, Justice, and Strong Institutions), ultimately contributing to national security and judicial efficiency. While this study establishes a statistical framework for evaluating noise resilience in forensic firearm identification, future research could explore expanding the dataset to mitigate overfitting and underfitting. Additionally, integrating the FLDA-based approach into mobile forensic applications could further enhance accessibility and operational efficiency in real-time forensic investigations.

Acknowledgments

The authors express their sincere gratitude to the Ministry of Higher Education (MOHE) for financial support through the MyBrain15 program scholarship and the Fundamental Research Grant Scheme (FRGS) (Grant No.: FRGS/1/2019/STG06/UPM/02/7 and UKM-ST-06-FRGS0183-2010). Special appreciation is extended to Prof. Dr. Nor Azura Md. Ghani for facilitating data collection utilized in this study. The authors also thank the Forensic Laboratory of the Royal Malaysia Police (RMP) for providing equipment and technical assistance during the data collection process.

References

Alanazi T.M., Berriri K., Albekairi M., Atitallah A.B., Sahbani A. & Kaaniche K. 2023. New real-time high-density impulsive noise removal method applied to medical images. *Diagnostics* **13**(10): 1709.

Albano A. 1974. Representation of digitized contours in terms of conic arcs and straight-line segments. *Computer Graphics and Image Processing* **3**(1): 23–33.

Canny J. 1986. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **8**(6): 679–698.

Chuan Z.L. 2014. Statistical firearm identification for forensic ballistics. PhD thesis. School of Mathematical Sciences, Universiti Kebangsaan Malaysia.

Chuan Z.L., Ghani N.A.M., Liong C.-Y. & Jemain A.A. 2013a. Automatic anchor point detection approach for firearms firing pin impression. *Sains Malaysiana* **42**(9): 1339–1344.

Chuan Z.L., Jemain A.A., Liong C.-Y. & Ghani N.A.M. 2013b. Effectiveness of cross entropy and Tsallis entropy thresholding for automatic forensic ballistics identification system. *Journal of Quality Measurement and Analysis* **9**(1): 33–46.

Chuan Z.L., Jemain A.A., Liong C.-Y., Ghani N.A.M. & Tan L.K. 2017. A robust firearm identification algorithm of forensic ballistics specimens. *Journal of Physics: Conference Series* **890**: 012126.

Chuan Z.L., Wei D.C.T., Yan C.L.W., Nasser M.F.A., Ghani N.A.M., Jemain A.A. & Liong C.-Y. 2023. A comparative study of two-dimensional statistical moment invariants features in formulating an automated probabilistic machine learning identification algorithm for forensic application. *Malaysian Journal of Fundamental and Applied Sciences* **19**(4): 525–538.

Fisher R.A. 1936. The use of multiple measurements in taxonomic problems. *Annals of Eugenics* **7**(2): 179–188.

Ghani N.A.M., Liong C.-Y. & Jemain, A.A. 2009a. Extraction and selection of basic statistical features for forensic ballistic specimen identification. *Sains Malaysiana* **38**(2): 249–260.

Ghani N.A.M., Liong C.-Y. & Jemain A.A. 2009b. Extraction and selection of geometric moment features for forensic ballistic specimen identification. *Matematika* **25**(1): 15–30.

Ghani N.A.M., Liong C.-Y. & Jemain A.A. 2010. Analysis of geometric moments as features for firearm identification. *Forensic Science International* **198**(1–3): 143–149.

Ghani N.A.M., Liong C.-Y. & Jemain, A.A. 2018. Neurocomputing approach for firearm identification. *Pertanika Journal of Science and Technology* **26**(1): 341–352.

Heard B.J. 2008. *Handbook Of Firearms and Ballistics: Examining and Interpreting Forensic Evidence*. 2nd Ed. Chichester, England: John Wiley & Sons.

Kamaruddin S.B.A., Ghani N.A.M., Liong C.-Y. & Jemain A.A. 2012. Firearm classification using neural networks on ring of firing pin impression images. *ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal* **1**(3): 27–35.

Leng J. & Huang Z. 2012. On analysis of circle moments and texture features for cartridge images recognition. *Expert Systems with Applications* **39**(2): 2092–2101.

Li J., Ablan C., Wu R., Guan S. & Yao J. 2021. Preprocessing method comparisons for VGG16 fast-RCNN pistol detection. *EPiC Series in Computing* **76**: 39–48.

Liang C.Z., Sern A.L.B., Cheng T.C., Luen D.L.K., Japashov N. & Hiae T.E. 2024. Empowering industry 5.0: Nurturing STEM tertiary education and careers through Additional Mathematics. In Al–Humairi S.N.S. (ed.). *Utilizing Renewable Energy, Technology, and Education for Industry 5.0*: 124–155. United States: IGI Global.

Liong C.-Y., Ghani N.A.M., Kamaruddin S.B.A. & Jemain A.A. 2012. Firearm classification based on numerical features of the firing pin impression. *Procedia Computer Science* **13**: 144–151.

Liong C.-Y., Ghani N.A.M., Kamaruddin S.B.A. & Jemain A.A. 2020. Conceptual design of firearm identification mobile application (FIMA). *AIP Conference Proceedings* **2266**(1): 090014.

Marr D. & Hildreth E. 1980. Theory of edge detection. *Proceedings of the Royal Society B: Biological Sciences* **207**(1167): 187–217.

Mirza B., Li X., Lauwers K., Reddy B., Muller A., Wozniak C. & Djali S. 2023. A clinical site workload prediction model with machine learning lifecycle. *Healthcare Analytics* **3**: 100159.

Moura L. & Kitney R. 1991. A direct method for least-squares circle fitting. *Computer Physics Communications* **64**(1): 57–63.

Nichols R.G. 1997. Firearm and toolmark identification criteria: A review of the literature. *Journal of Forensic Sciences* **42**(3): 466–474.

Okoye K., Nganji J.T., Escamilla J. & Hosseini S. 2024. Machine learning model (RG-DMML) and ensemble algorithm for prediction of students' retention and graduation in education. *Computers and Education: Artificial Intelligence* **6**: 100205.

Razak N.A., Liong C.-Y., Jemain A.A., Ghani N.A.M., Zakaria S. & Sulaiman H. 2017. Firing pin impression segmentation using Canny edge detection operator and Hough Transform. *Journal of Telecommunication, Electronic and Computer Engineering* **9**(1): 23–26.

Sood S., Singh H., Malarvel M. & Ahuja R. 2021. Significance and limitations of deep neural networks for image classification and object detection. *Proceedings of the 2021 2nd International Conference on Smart Electronics and Communication*, pp. 1453–1460.

Thumwarin P. 2008. An automatic system for firearm identification. *Proceedings of the 2008 International Symposium on Communications and Information Technologies*, pp. 100–103.

Toktas A., Erkan U., Ustun D. & Wang X. 2023. Parameter optimization of chaotic system using Pareto-based triple objective artificial bee colony algorithm. *Neural Computing and Applications* **35**: 13207–13223.

Xin L-P., Zhou J. & Rong G. 2000. A cartridge identification system for firearm authentication. *Proceedings of the 5th International Conference on Signal Processing*, pp. 1405–1408.

Yunus A. & Loo J. 2024. London street crime analysis and prediction using crowdsourced dataset. *Journal of Computational Mathematics and Data Science* **10**: 100089.

Zhou J., Xin L.-P., Gao D.-S., Zhang C.-S. & Zhang D. 2001. Automated cartridge identification for firearm authentication. *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 749–754.

Centre for Mathematical Sciences
Universiti Malaysia Pahang Al-Sultan Abdullah
Lebuh Persiaran Tun Khalil Yaakob
26300 Kuantan
Pahang, MALAYSIA
E-mail: chuanzl@umpsa.edu.my[*], *limbingsern123@gmail.com, zecheng.t@gmail.com*

Department of Mathematical Sciences
Faculty of Science & Technology
Universiti Kebangsaan Malaysia
43600 UKM Bangi
Selangor, MALAYSIA
E-mail: azizj@ukm.edu.my, lg@ukm.edu.my

Received: 17 April 2025
Accepted: 18 August 2025

[*]Corresponding author