

Article

Mahalanobis-Constrained Hard-Clustering Algorithm

Vedran Novoselac 

Mechanical Engineering Faculty in Slavonski Brod, University of Slavonski Brod, Trg I. B. Mažuranić 2,
35000 Slavonski Brod, Croatia; vnovoselac@unisb.hr

Abstract

The paper proposes a modification of the standard k -means algorithm that removes low-probability tail points of Gaussian components using the p -quantile of the chi-squared distribution as a trimming threshold based on the squared Mahalanobis distance. The procedure is evaluated on synthetic noisy datasets and compared with well-known state-of-the-art clustering methods. Applications in circular image pattern recognition demonstrate effective detection of spherical patterns in highly noisy environments. Pattern detection is driven by an optimization using a constructed spherical clustering-validity index, and the method's practical potential is confirmed by comparisons with state-of-the-art circular detection techniques.

Keywords: cluster analysis; data analysis; image processing; pattern recognition

MSC: 62H30; 94A16; 62H35; 68T10

1. Introduction

It is well established that iterative methods find extensive application across a wide range of disciplines—such as cluster analysis, pattern recognition, and image processing—and thus contribute to many aspects of interdisciplinary research [1–4]. Consequently, well-known iterative algorithms are often used and modified to identify clusters in datasets, for example the standard k -means algorithm and the Expectation–Maximization (EM) algorithm [5–19]. These methods and their modifications are widely employed to achieve more reliable data modelling and to address domain-specific problems. Furthermore, the most state-of-the-art deep-learning methods—particularly convolutional neural networks (CNN) and long short-term memory networks (LSTM)—together with modern optimizers provide useful approaches for task-specific feature extraction and model tuning [20]. For unsupervised clustering in vision, methods that jointly learn embeddings and cluster assignments have proven effective, either by optimizing deep embeddings [21] or by alternating k -means on deep features with supervised training on the resulting pseudo-labels [22].

For this purpose, the k -means clustering method is modified in this work by incorporating the Mahalanobis distance; this modification and related Mahalanobis-based variants are commonly used to explore and evaluate different data models and research areas [6,8,9,16–19]. The proposed method was studied and tested on artificial generated dataset with Gaussian distributed clusters which are contained in high level noisy environment. Thus, the proposed method is based on a procedure that excludes data points that are located in the low-probability tails of the Gaussian components, leveraging the well-known fact that the Mahalanobis distance follows a chi-squared distribution [23]. It has been shown that the proposed method presents an effective clustering method in noisy environments where outliers constitute a large proportion of the data or where the data structure is inherently prone to



Academic Editor: Ali Al Bataineh

Received: 30 May 2026

Revised: 21 July 2026

Accepted: 24 July 2026

Published: 28 July 2026

Copyright: © 2026 by the author.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and

conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

contamination [24,25]. The presented method was compared with well-known state-of-the-art clustering methods [1,3–5,11], where the performance is evaluated by repeated Monte Carlo simulations driven by various measurement metrics [1,5,11].

The paper concludes with an application of the proposed method to image processing pattern recognition, a scientific field that considers wide range of central-based modifications and their applications [6,12,14,26]. Specifically, the proposed method is applied to detect spherical patterns in different scene test images. The task is performed on pre-processed, segmented images, which are divided into subregions—advancing area in image processing [16,18,27,28]. Results are obtained by optimizing the p -quantile of the chi-squared distribution, which regulates the trimming procedure. The optimization is performed by observing the spherical clustering validity index, which provides a measure for evaluating final clustering results [1,4,10,17]. The proposed method is evaluated on well-known state-of-the-art circle detection models [7,15,29,30], which are evaluated by the proposed spherical cluster validity measure in order to emphasize the effect of highly noisy image structures. Finally, the paper ends with a discussion that considers further possibilities and directions for research on the proposed method, followed by a conclusion.

2. Mahalanobis-Constrained Hard-Clustering Algorithm

This section presents the development of the proposed method, obtained by modifying the k -means procedure with center-based constraints that exploit the fact that the squared Mahalanobis distance follows a chi-square distribution [10,24,31], where numerous related Mahalanobis-based developments have been investigated in recent years across various application domains [6,8–10,16–19]. At the end of the section, numerical experiments are presented to evaluate the proposed method: specifically, clustering performance is assessed on artificial datasets generated under various noisy conditions.

2.1. The Proposed Method

To develop a center-based trimming constraint for the iterative procedure, the (squared) Mahalanobis distance

$$d_M(x; \mu, \Sigma) = (x - \mu)^\top \Sigma^{-1} (x - \mu),$$

is considered, where $\mu \in \mathbb{R}^n$ denotes the mean vector and $\Sigma \in \mathbb{R}^{n \times n}$ is the covariance matrix. Assuming that the cluster data are Gaussian, the squared Mahalanobis distance is known to follow a chi-squared distribution with n degrees of freedom. Hence, for $p \in [0, 1]$ the following constraint is imposed:

$$d_M(x; \mu, \Sigma) \leq \chi_n^2(p), \quad (1)$$

where $\chi_n^2(p)$ denotes the p -quantile value of the chi-squared distribution with n degrees of freedom. Thus, at the current iteration, the trimming is performed by rejecting the $(1 - p)100\%$ of observations lying in the low-probability tails.

In order to demonstrate the performance of the proposed method and present the corresponding algorithm, an illustrative example is presented in Figure 1, where an artificial dataset $X = \{x_i \in \mathbb{R}^n : i = 1, \dots, m\}$ is shown together with the contour lines of the iterative step. So, each step is generated by k centroids $\{\mu_1^{(t)}, \dots, \mu_k^{(t)}\}$ and covariances $\{\Sigma_1^{(t)}, \dots, \Sigma_k^{(t)}\}$, which eventually define clusters $\{\pi_1^{(t)}, \dots, \pi_k^{(t)}\}$. The black marked dotted line denotes the borderlines $\|x - \mu_j^{(t)}\|_2 = \|x - \mu_l^{(t)}\|_2, j \neq l$, suggesting the performance of the standard k -means algorithm. By the blue marked dashed lines the contours of the ellipsoid $d_M(x; \mu_j^{(t)}, \Sigma_j^{(t)}) = \chi_n^2(p)$ is denoted, which includes the constraint (1) that is generated by the each corresponding cluster centroid and covariance. In that sense, the trimming procedure is performed, which involves all the data contained in an ellipsoid region that is generated

by some predefined value $p \in [0, 1]$, and excludes all observations that exceed the p -quantile value. This performance is conducted by defining the trimming indicator as

$$\delta_{ij}^{(t)} = \begin{cases} 1, & d_M(x_i; \mu_j^{(t)}, \Sigma_j^{(t)}) \leq \chi_n^2(p); \\ 0, & \text{otherwise,} \end{cases} \quad (2)$$

which considers whether the observations are excluded from the calculation of the current iteration step or not, where it denoted by the red dashed lines that the observation x_i is not associated with any cluster, i.e., $\delta_{ij}^{(t)} = 0, \forall j$, and thus it can be declared as an outlier. Consequently, the clusters can be defined as

$$\pi_j^{(t)} = \{x_i: \|x_i - \mu_j^{(t)}\|_2 = \delta_{ij}^{(t)} \cdot \min_{1 \leq l \leq k} \|x_i - \mu_l^{(t)}\|_2\}, \quad i = 1, \dots, m.$$

Eventually, the next iteration step can be performed by generating new centroids and covariances, i.e.,

$$\mu_j^{(t+1)} = \frac{1}{|\pi_j^{(t)}|} \sum_{x_i \in \pi_j^{(t)}} x_i;$$

$$\Sigma_j^{(t+1)} = \frac{1}{|\pi_j^{(t)}| - 1} \sum_{x_i \in \pi_j^{(t)}} (x_i - \mu_j^{(t+1)})^\top (x_i - \mu_j^{(t+1)}),$$

where the entire procedure is repeated until convergence of the centroids and covariances. This performance is evaluated by considering the iteration error step $\Delta^{(t)}$ with a small predefined stoppage criterion $\varepsilon > 0$, where the centroid errors are measured using the L_2 -norm (Euclidean norm), and the covariances using the Frobenius matrix norm, i.e.,

$$\Delta^{(t+1)} = \left(\sum_{j=1}^k (\|\mu_j^{(t)} - \mu_j^{(t+1)}\|_2^2 + \|\Sigma_j^{(t)} - \Sigma_j^{(t+1)}\|_F^2) \right)^{\frac{1}{2}}.$$

Finally, the proposed method can be summarized by Algorithm 1.

Remark 1. By considering the trimming indicator (2), it can be that if $p \rightarrow 1$, every observation will be included in the modeling process because $\chi_n^2(p) \rightarrow \infty$. In that case, all data will be included, and the proposed method will behave identically to the standard k -means algorithm.

Remark 2. The computational complexity of Algorithm 1 is

$$k \mathcal{O}(m + mn^2 + n^3),$$

since the proposed method involves computations of k components with corresponding expectations ($\mathcal{O}(m)$), covariance matrices of size $n \times n$ ($\mathcal{O}(mn^2)$), and their inverses ($\mathcal{O}(n^3)$).

2.2. Numerical Experiments

In this subsection, the proposed method is examined. First, it is tested and evaluated on an artificial dataset containing an outlier (i.e., in a noisy environment) to demonstrate its capabilities. Afterwards, it is compared with the well-known state-of-the-art clustering methods.

Algorithm 1 The Proposed Method

Input: $X = \{x_i \in \mathbb{R}^n : i = 1, \dots, m\}$, $\{\mu_1^{(0)}, \dots, \mu_k^{(0)}\}$, $\{\Sigma_1^{(0)}, \dots, \Sigma_k^{(0)}\}$, $p \in [0, 1]$, $t = 0$, $\varepsilon > 0$;

1: *loop*:

2: Calculate $\delta_{ij}^{(t)}, \forall i \in \{1, \dots, m\}, \forall j \in \{1, \dots, k\}$:

$$\delta_{ij}^{(t)} = \begin{cases} 1, & d_M(x_i; \mu_j^{(t)}, \Sigma_j^{(t)}) \leq \chi_n^2(p); \\ 0, & \text{otherwise;} \end{cases}$$

3: Determinate clusters $\pi_j^{(t)}, \forall j \in \{1, \dots, k\}$:

$$\pi_j^{(t)} = \{x_i : \|x_i - \mu_j^{(t)}\|_2 = \delta_{ij}^{(t)} \cdot \min_{1 \leq l \leq k} \|x_i - \mu_l^{(t)}\|_2\}, \quad i = 1, \dots, m;$$

4: Calculate $\forall j \in \{1, \dots, k\}$ new centroid and covariance matrix:

$$\mu_j^{(t+1)} = \frac{1}{|\pi_j^{(t)}|} \sum_{x_i \in \pi_j^{(t)}} x_i;$$

$$\Sigma_j^{(t+1)} = \frac{1}{|\pi_j^{(t)}| - 1} \sum_{x_i \in \pi_j^{(t)}} (x_i - \mu_j^{(t+1)})^\top (x_i - \mu_j^{(t+1)});$$

5: Calculate error iteration step:

$$\Delta^{(t+1)} = \left(\sum_{j=1}^k (\|\mu_j^{(t)} - \mu_j^{(t+1)}\|_2^2 + \|\Sigma_j^{(t)} - \Sigma_j^{(t+1)}\|_F^2) \right)^{\frac{1}{2}};$$

$$t = t + 1;$$

6: *iterate until* $\Delta^{(t)} < \varepsilon$;

Output: $\{\mu_1^{(t)}, \dots, \mu_k^{(t)}\}, \{\Sigma_1^{(t)}, \dots, \Sigma_k^{(t)}\}$;

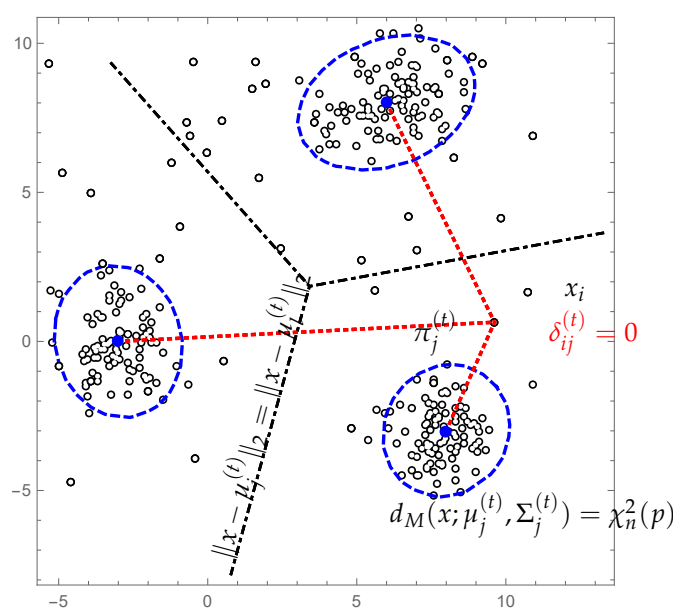


Figure 1. Schematic of the Proposed Method.

2.2.1. Evaluation on Noisy Data

To evaluate the proposed method, artificial datasets that include outliers are considered. Outliers (noisy data) are atypical observations that can distort fitted models and

undermine statistical inference. They may arise from different generating mechanisms, measurement or data-entry errors, or inherent noisy structure, and can therefore exert disproportionate influence and lead to misleading conclusions. Detecting and assessing such observations prior to model building allows analysts to decide whether to accommodate, downweight, or exclude them, and to choose modeling strategies that limit or reduce their impact [10,17,24,25,31,32].

The following Figure 2 illustrates the proposed method on datasets whose clusters π_j^* follow Gaussian distributions $\mathcal{N}(\mu_j^*, \Sigma_j^*)$ and are embedded in an artificially generated high-level noisy structured environment. In the presented figure, the black dashed ellipses $d_M(x; \mu_j^*, \Sigma_j^*) = \chi_n^2(0.95)$ denote the contours of the original Gaussian-distributed clusters, the blue ellipse $d_M(x; \mu_j^{(0)}, \Sigma_j^{(0)}) = \chi_n^2(0.95)$ denotes the input settings, and the red ellipse $d_M(x; \mu_j^{(t)}, \Sigma_j^{(t)}) = \chi_n^2(0.95)$ denotes the final output of the Algorithm 1. The results in Figure 2a,b demonstrate the performance of Algorithm 1 for $p = 0.95$, which achieves good results, while the results in Figure 2c,d ($p = 0.90$) show degraded performance, as the trimming removes too much data from the low-probability tails and can therefore produce poorer outcomes.

In Figure 3, the performance of the final Algorithm 1 output is presented via the Sum of Squared Errors (SSE) loss function

$$\text{SSE} = \left(\sum_{j=1}^k (\|\mu_j^{(t)} - \mu_j^*\|_2^2 + \|\Sigma_j^{(t)} - \Sigma_j^*\|_F^2) \right)^{\frac{1}{2}}. \quad (3)$$

Results reveal that the p -quantile value depends on the input settings and data structure what can cause sensitivity near the minimum value of the observed SSE loss function. The situation corresponding to the example in Figure 2, when the noise level reaches approximately 80%, reveals sensitivity around the minimum value and causes a significant jump in the SSE. In another case, when the noise level reaches approximately 30%, the p -quantile demonstrates some robustness around the minimal value for the corresponding input setting, despite different noise density structures and input locations.

In Figure 4a,b, the iteration errors for the examples shown in Figure 2 are presented, which show the convergence properties of Algorithm 1 for each case. Furthermore, the iteration error of Algorithm 1 for $p = 0.65$ is presented for both examples, since excessive trimming leads to degenerate conditional covariance matrices and consequently causes the immediate stoppage of Algorithm 1.

2.2.2. Comparison with State-of-the-Art Clustering Methods

This section presents a comparison of the proposed method with well-known state-of-the-art clustering techniques and some of their robust modifications. To that end, several established methods are evaluated: standard k -means, EM algorithm, and DB-SCAN [1,3,4]. Robust adaptations are also examined, including trimmed k -means [5] and a robust EM that models clusters with the Student t -distribution; the t -distribution's heavy tails increase resistance to outliers [11]. Given the complexity of datasets that exhibit specific structure or contain outliers, this comparison is conducted to demonstrate the advantage of the proposed method over techniques that perform similarly on well-grouped, structureless datasets with low noise.

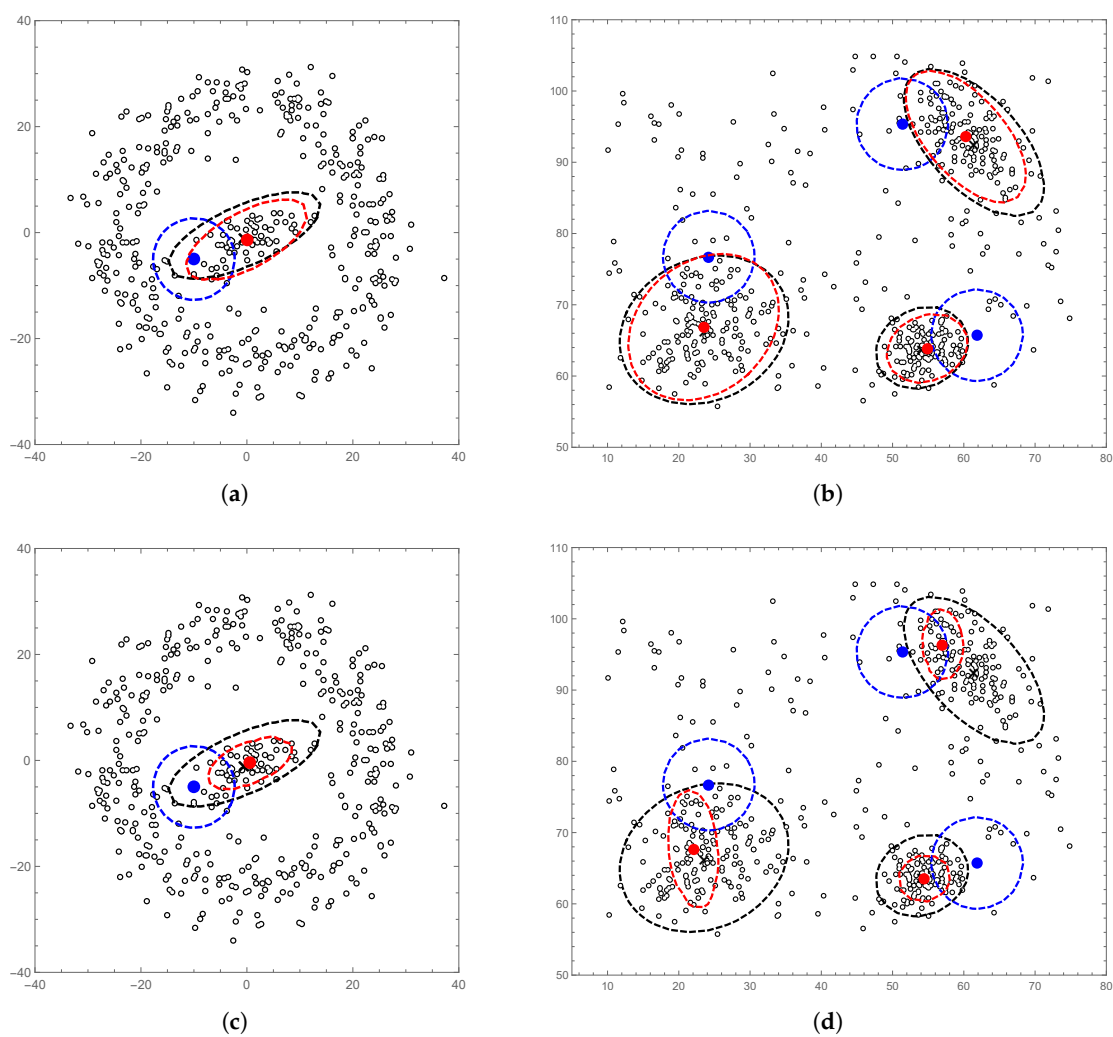


Figure 2. Numerical examples: (a) Noise level $\approx 80\%$ ($p = 0.95$). (b) Noisy level $\approx 30\%$ ($p = 0.95$). (c) Noise level $\approx 80\%$ ($p = 0.90$). (d) Noisy level $\approx 30\%$ ($p = 0.90$).

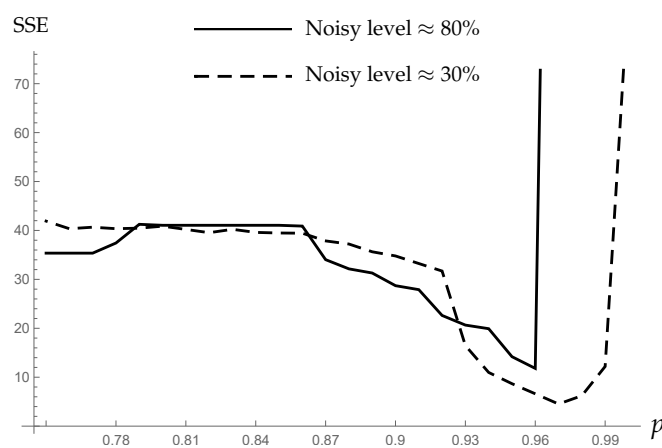


Figure 3. SSE via p -quantile.

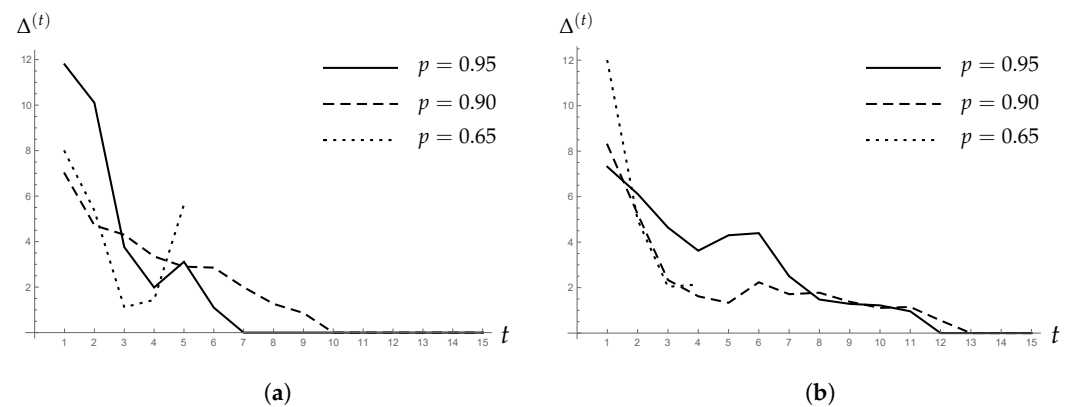


Figure 4. Iteration error: (a) Noise level $\approx 80\%$. (b) Noisy level $\approx 30\%$.

Figure 5 presents the final results of the presented clustering methods for the case of Gaussian-distributed clusters $\mathcal{N}(\mu_j^*, \Sigma_j^*)$ that are located in an artificial noisy structured environment with noise level $\approx 60\%$. The black dashed ellipses denote $d_M(x; \mu_j^*, \Sigma_j^*) = \chi_n^2(0.95)$, while the red dashed ellipses $d_M(x; \mu_j^{(t)}, \Sigma_j^{(t)}) = \chi_n^2(0.95)$ denote the final output of the corresponding method. In this case, the means and covariance matrix are generated by the final output cluster $\pi_j^{(t)}$, such that $\mu_j^{(t)} = \text{Mean}(\pi_j^{(t)})$ and $\Sigma_j^{(t)} = \text{Cov}(\pi_j^{(t)})$. Each method's input parameters (e.g., cluster centres and covariances) are initialized close to the true cluster parameters to enable a fair comparison of final results and to facilitate the SSE loss function (3). In this scenario, Figure 5a,b present results of the standard and the trimmed k -means versions, respectively. The inputs are set near the centroids $\{\mu_1^*, \dots, \mu_k^*\}$ for both methods, while the trimming parameter that minimizes Δ^* for the trimmed version is set to 63%. Obviously, the results show that the trimmed k -means greatly outperform the standard version. Figure 5c,d present the standard and robust variation of the EM method, which both considered centroids input near $\{\mu_1^*, \dots, \mu_k^*\}$ and covariances $\{\Sigma_1^*, \dots, \Sigma_k^*\}$, where equal a priori probabilities for each cluster are considered. Due to the high noise level, both methods were shown to perform poorly. Figure 5e presents the results of the DBSCAN clustering method, which has shown very good performance for a minimum distance of 1.8 and a minimum number of points of 5, demonstrating that the target clusters can be tracked (DBSCAN also detects a larger number of smaller clusters in this situation). Finally, Figure 5f presents the proposed method, which has been run for the input near the original means and covariances, where results are obtained by considering $p = 0.98$ (see Figure 3).

Optimal results for the proposed method are obtained by running repeated Monte Carlo simulations in which random initializations are generated across the search region and cluster-validation indices are used to select the best solution [1,3,4]. Figure 6 shows simulations performed on various clustering criteria for $p = 0.98$, which has demonstrated the good performance discussed in the previous section (see Figure 3). Each run is performed with $k = 3$, with starting centroids $\mu_j^{(0)} \in X$ randomly generated over the observed dataset, and covariances $\Sigma_j^{(0)} = \sigma_n I_n$ taken as predefined scaled identity matrices with scaling randomly drawn from $\sigma_n \in [5, 10]$, and also including good separability of the centroids such that $\|\mu_i - \mu_j\|_2 > \sigma_i + \sigma_j$. Under these conditions, starting positions were chosen to avoid empty-case initializations and bead configurations that produce an ill-conditioned bead matrix, which would otherwise cause immediate termination of Algorithm 1.

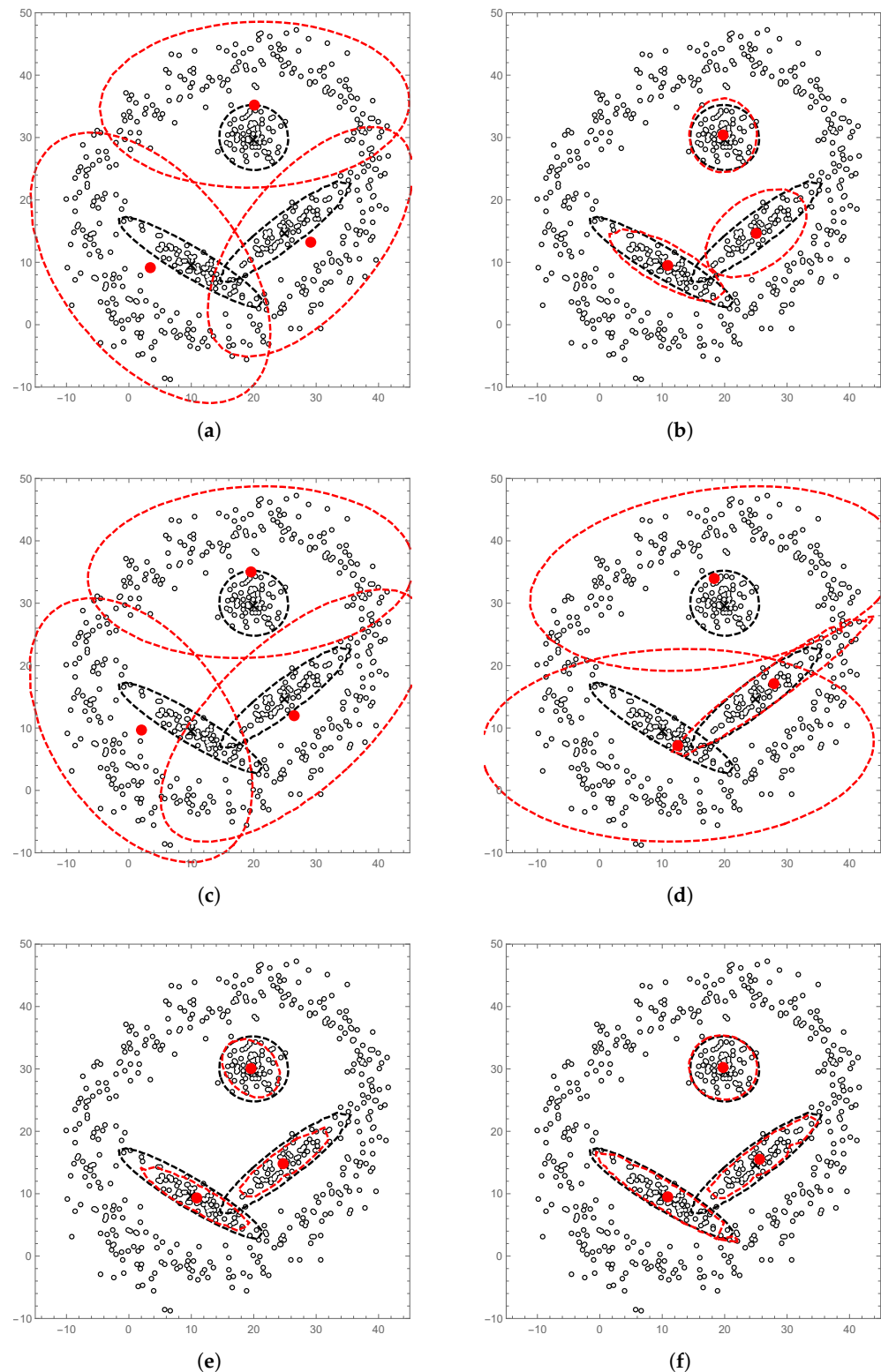


Figure 5. Clustering results: (a) k -means. (b) Trimmed k -means. (c) EM algorithm. (d) Robust EM algorithm. (e) DBSCAN. (f) The proposed method.

So, Figure 6a considers the performance of Algorithm 1 by evaluating the outlier precision (OutPrec), which measures the correctness of detected outliers as the number of correctly detected outliers divided by all detected outliers. Figure 6b evaluates the performance by considering outlier recall (OutRec), which measures the proportion of actual outliers that were detected, defined as the number of correctly detected outliers

divided by all true outliers. Figure 6c shows the Adjusted Rand Index (ARI), which measures similarity between two clusterings and ranges from -1 to 1 : 1 indicates perfect agreement, 0 indicates agreement comparable to random, and negative values indicate less agreement than expected by chance [33]. Figure 6d shows the average Normalized Mutual Information (NMI), which quantifies agreement between two clusterings by normalizing their mutual information by the label entropies [34]. Figure 6e presents the Root Mean Squared Error (RMSE) simulation, which measures the square root of the average squared differences between predicted and actual values:

$$\text{RMSE} = \left(\frac{1}{k(n + n^2)} \sum_{j=1}^k (\|\mu_j^{(t)} - \mu_j^*\|_2^2 + \|\Sigma_j^{(t)} - \Sigma_j^*\|_F^2) \right)^{1/2}.$$

Figure 6f shows performance measured by the Covariance Error (CovErr), defined as the difference between the estimated covariance matrix and the true covariance:

$$\text{CovErr} = \sum_{j=1}^k \|\Sigma_j^{(t)} - \Sigma_j^*\|_F.$$

Figure 6g shows simulations using the Sum of Squared Errors (SSE) (3), which serves as the primary optimization metric in the evaluation. Finally, Figure 6h reports running-time performance; runtimes were recorded in the Wolfram Cloud (Wolfram Research, Champaign, IL, USA).

The final results for the methods in Figure 5 are listed in Table 1, with comparative methods obtained by fine-tuning parameters and selecting the solution with the minimal SSE and the proposed method obtained via the repeated Monte Carlo simulations shown in Figure 6. OutPrec is computed only for methods that handle outliers, with Algorithm 1 achieving the best performance at 97% precision and trimmed k -means at 96%. OutRec shows the best performance in DBSCAN, achieving 99% (though DBSCAN lacks OutPrec), while Algorithm 1 and trimmed k -means perform slightly worse. All methods show good ARI results, with DBSCAN and Algorithm 1 achieving the highest ARI of 0.97. Also, all methods perform well for NMI, with DBSCAN and Algorithm 1 achieving the highest NMI of 0.96. In RMSE, CovErr, and SSE cases, the proposed method has achieved the best results. The proposed And finally, the running-time performance shows that standard k -means and its trimmed variant slightly outperform Algorithm 1, while EM and its robust modification converge more slowly.

Table 1. Final results.

Method	k -Means	Trimmed k -Means	EM	Robust EM	DBSCAN	Algorithm 1
Measure						
OutPrec	-	96%	-	-	91%	97%
OutRec	-	97%	-	-	99%	98%
ARI	0.95	0.94	0.86	0.93	0.97	0.97
NMI	0.93	0.93	0.86	0.92	0.96	0.96
RMSE	8.04	1.08	9.05	11.74	1.05	0.31
CovErr	144.47	19.42	162.12	211.30	18.92	5.49
SSE	144.78	19.46	162.38	211.41	18.97	5.66
Runtime (sec)	0.15	0.17	31.48	36.17	2.20	0.46

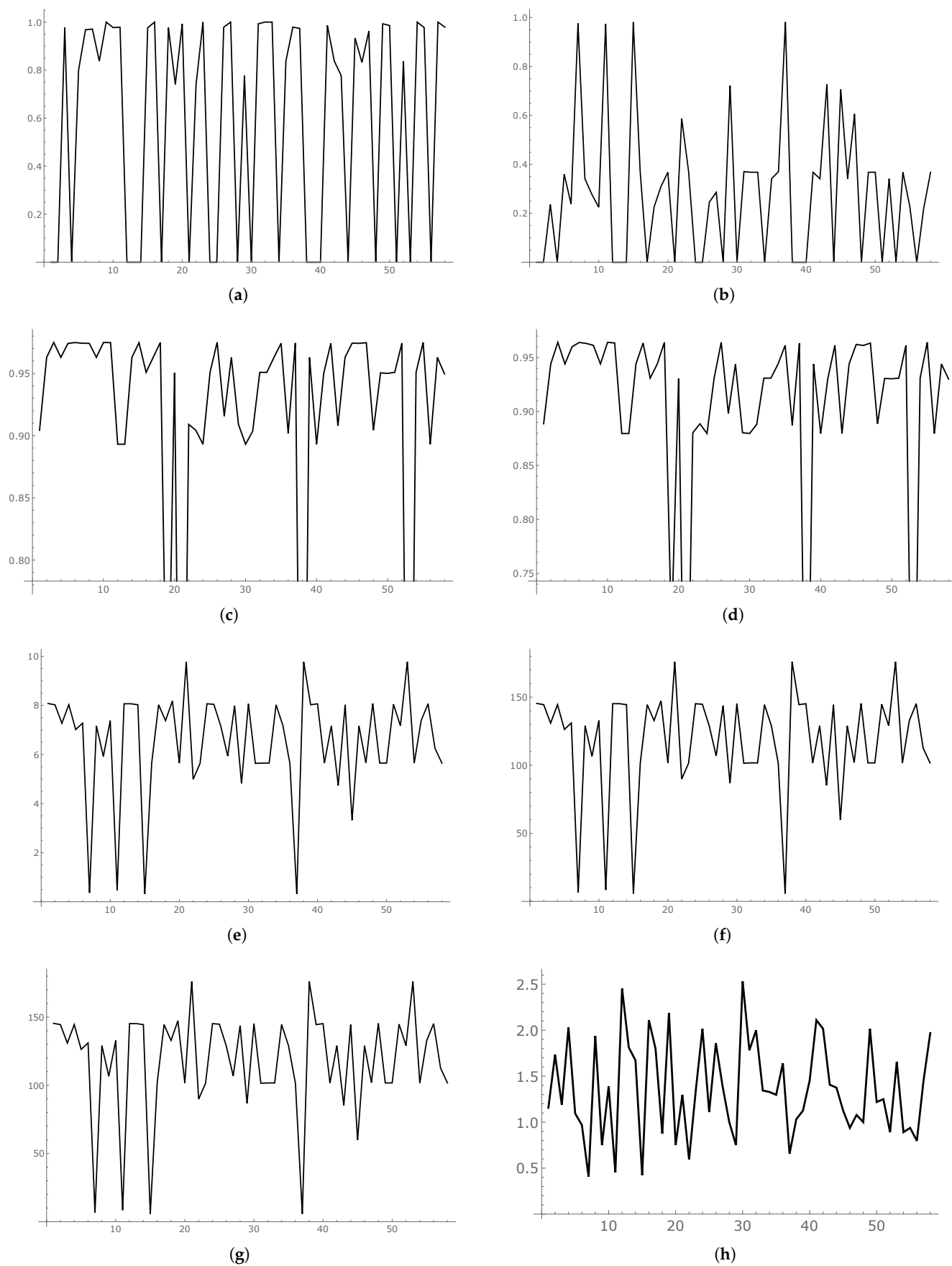


Figure 6. Repeated Monte Carlo simulations for Algorithm 1 ($k = 3$, $p = 0.98$): (a) OutPrec. (b) OutRec. (c) ARI. (d) NMI. (e) RMSE. (f) CovErr. (g) SSE. (h) Runtime.

3. Application: Recognition of Circular Patterns in Images

This section discusses the application of the proposed method to image processing, particularly pattern recognition [2]. So, Figure 7 presents an image of size $M \times N$ represented in matrix form as $X = [x_{ij}]_{1 \leq i \leq M, 1 \leq j \leq N}$, where each component x_{ij} denotes the color intensity of the corresponding pixel. One of the best-known representations of pixel color intensities is the RGB color space: a color image is treated as a vector-valued image signal in which each pixel is a three-component vector (Red, Green, Blue). In this scenario the color intensities are defined by $x_{ij} = (R_{ij}, G_{ij}, B_{ij})$, with $R_{ij}, G_{ij}, B_{ij} \in \{0, 1, \dots, 255\}$.

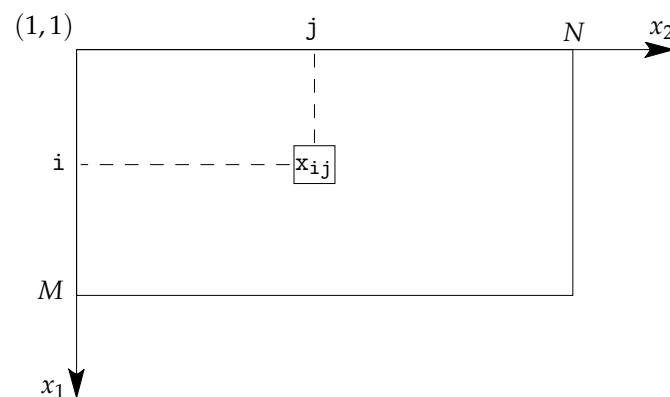


Figure 7. Image representation.

The proposed method is illustrated for spherical pattern detection in a variety of industrial-scene RGB images, showing different types of holes and circular objects such as bearings and rivets. Figure 8 presents some of these image types, which are tested to detect those patterns.

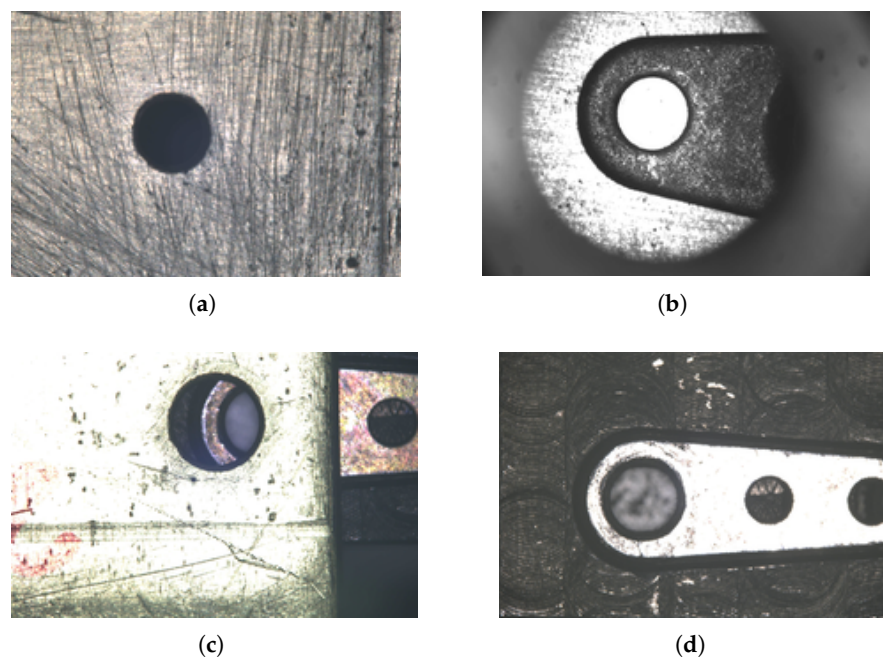


Figure 8. Test images : (a) Hole (150×220). (b) Bearing (150×220). (c) Bearing and hole (150×230). (d) Bearing and holes (150×220).

The experiments were conducted on RGB color-segmented images obtained by using the standard k -means algorithm [2,27,28]. The application of a cluster partitioning scenario by the k -means at the RGB color intensities set $RGB = \{x_{ij} \mid i = 1, \dots, M \ \& \ j = 1, \dots, N\}$,

divides the observed RGB into the k disjoint set $\{\Pi_1, \dots, \Pi_k\}$ such that $RGB = \bigcup_{t=1}^k \Pi_t$. Consequently, each partition Π_t generates a corresponding subregion $R_t = \{(i, j) \mid x_{ij} \in \Pi_t\}$ of the spatial region of the observed image X , and thus can be observed and tested in order to detect a specific pattern which they possess. In Figure 9, the image subregion sets $R^* = R_t$, $t \in \{1, \dots, k\}$, are shown in the image spatial domain for the corresponding examples presented in Figure 8. The subregions R^* are obtained by applying k -means with $k = 2$, where it is assumed, without the loss of generality, that R^* contains observed spherical patterns. Finally, the proposed method is presented by applying the Algorithm 1 on the set $X = R^*$, where the starting initial Mahalanobis contours $d_M(x; \mu_j^{(0)}, \Sigma_j^{(0)}) = \chi_n^2(p)$ are denoted by blue, while the final contours $d_M(x; \mu_j^{(t)}, \Sigma_j^{(t)}) = \chi_n^2(p)$ are colored red (for each example $p = 0.88$ is observed).

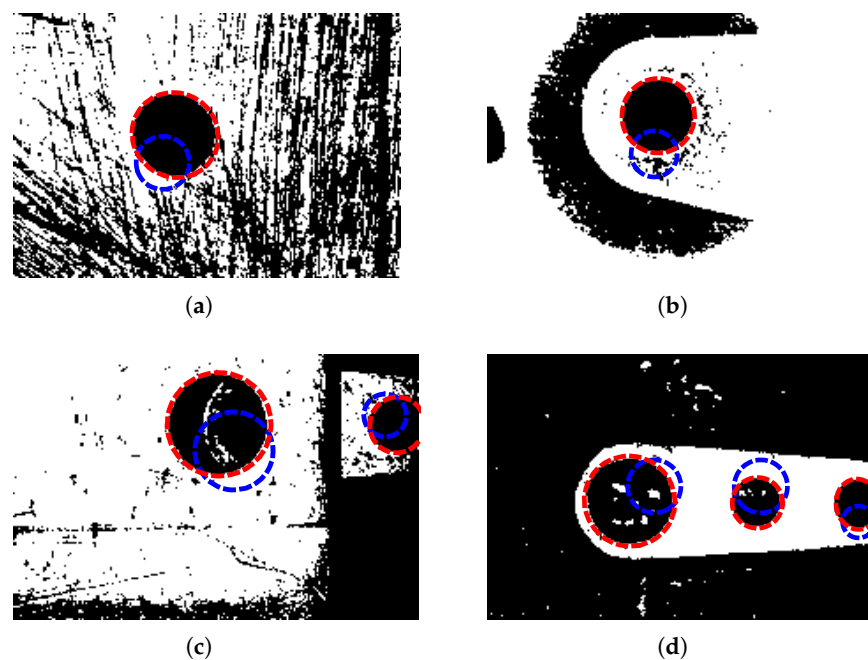


Figure 9. Image segmentation and pattern recognition results: (a) Hole (150×220). (b) Bearing (150×220). (c) Bearing and hole (150×230). (d) Bearing and holes (150×220).

To choose the appropriate p -quantile and thus optimize the rejection procedure while avoiding unnecessary gaps, the cluster validity index Ind_j is developed for each final cluster $\pi_j^{(t)}$. It is modeled to incorporate the condition number κ_2 of the corresponding covariance matrix $\Sigma_j^{(t)}$ and a measure of how well the data support that component. So, the spherical detection index is defined as

$$\text{Ind}_j(p) = \frac{1}{\kappa_2(\Sigma_j^{(t)})} \cdot \frac{|\pi_j^{(t)}|}{\#\{(i, j) \mid d_M((i, j); \mu_j^{(t)}, \Sigma_j^{(t)}) \leq \chi_n^2(p)\}}$$

where $\kappa_2(\Sigma_j^{(t)})$ presents the condition number of a covariance matrix in the sense of L_2 -norm, i.e.,

$$\kappa_2(\Sigma_j^{(t)}) = \frac{\lambda_j^{\max}}{\lambda_j^{\min}},$$

where $\lambda_j^{\min}, \lambda_j^{\max}$ are minimal and maximal singular values of $\Sigma_j^{(t)}$. Finally, the index that includes the minimum of the overall component indices is defined as

$$\text{Ind}(p) = \min_{j=1,\dots,k} \text{Ind}_j(p), \quad (4)$$

where the goal is to achieve the maximal value that indicates the most spherical and supported components. In Figure 10, the measurements via p -quantile value are conducted for each corresponding specification that is presented in Figure 9, where it is established that the most appropriate p -quantile value reaches its maximum for the observed index Ind around $p \approx 0.88$.

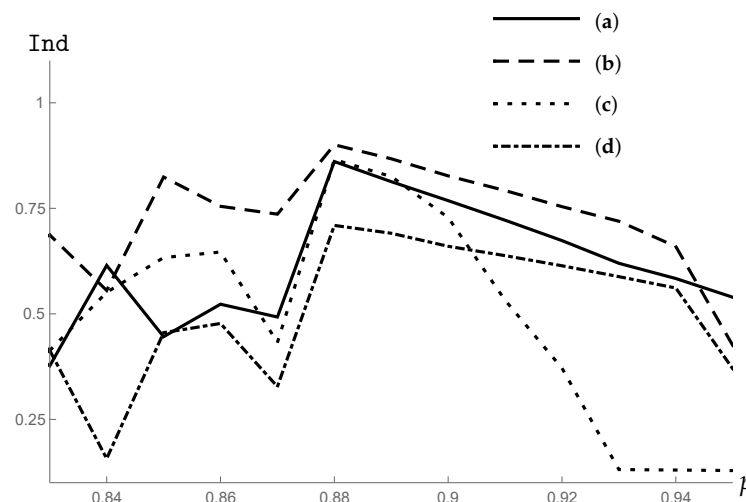


Figure 10. Index via p -quantile value.

Comparison with State-of-the-Art Circle Detectors

In this subsection, the evaluation is presented using the proposed method (Algorithm 1, $p = 0.88$) and well-known state-of-the-art circle detection methods. The proposed method is compared with the robust version of the Center-Based Circle Detection method [7,15], the Hough Circle Detection method [30], and the modified RANSAC Circle Detection method [29]. Each method was tested on the image examples shown in Figure 8 and evaluated using the defined spherical validity index Ind (4) for each corresponding image. Each compared method was assessed on the image edge-detector domain [2], and the final output was taken as the maximal value of Ind in order to detect spherical objects which are located within highly noisy structures. For that purpose, the RANSAC Circle Detector is modified to find circles that maximize Ind. In this situation, the circles are randomly generated along the edge domain until the best performance of Ind is established.

The overall results are presented in the following figures, where Figure 11 illustrates the effectiveness of the observed method for the test image Figure 8a. Each method achieved good performance, where RANSAC slightly outperforms Hough Circle Detector and the Proposed Method, except the robust variation of the Center-Based Circle Detector, which failed due to high noise contamination. Similar results are obtained for the test image Figure 8a, which are presented in Figure 12. Figure 13c shows that the Hough Circle Detector fails to detect an isolated spherical object, while the Proposed Method and RANSAC modification establish good performance of detection, as is presented in Figures 13a and 13d, respectively. Finally, Figure 14 presents similar results as the previous case, where the Hough Circle Detector has shown a lack of detection of a skew isolated circle, and the best performance of the modified RANSAC method. In this situation,

the visual performance of RANSAC (Figure 14d) does not match the high cluster validity index $\text{Ind} = 0.92$ due to the lack of proper initialization, which leads to overlapping.

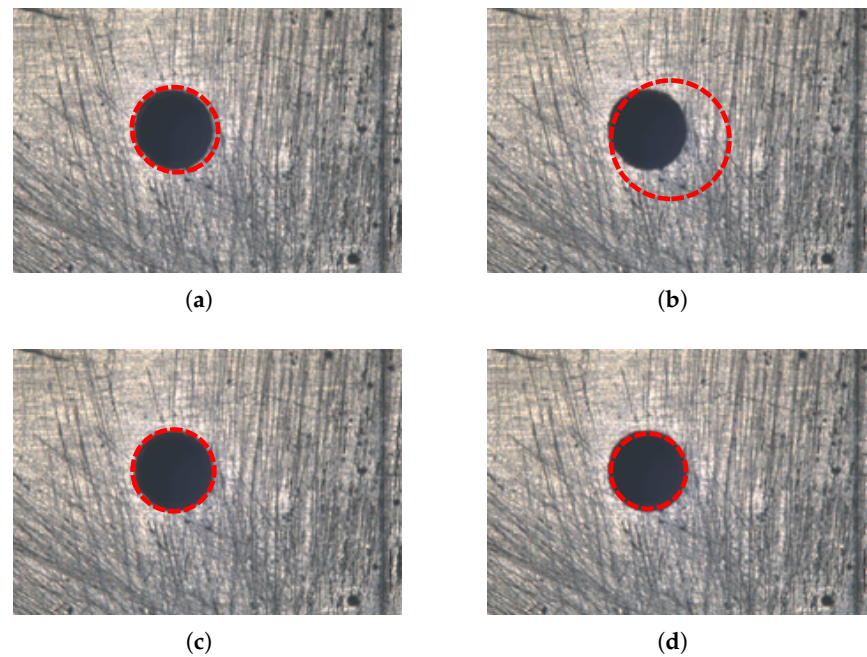


Figure 11. Final results for test image Hole (150×220): (a) Proposed Method ($\text{Ind} = 0.86$). (b) Center-Based Circle Detection ($\text{Ind} = 0.58$). (c) Hough Circle Detection ($\text{Ind} = 0.98$). (d) RANSAC Circle Detection ($\text{Ind} = 1$).

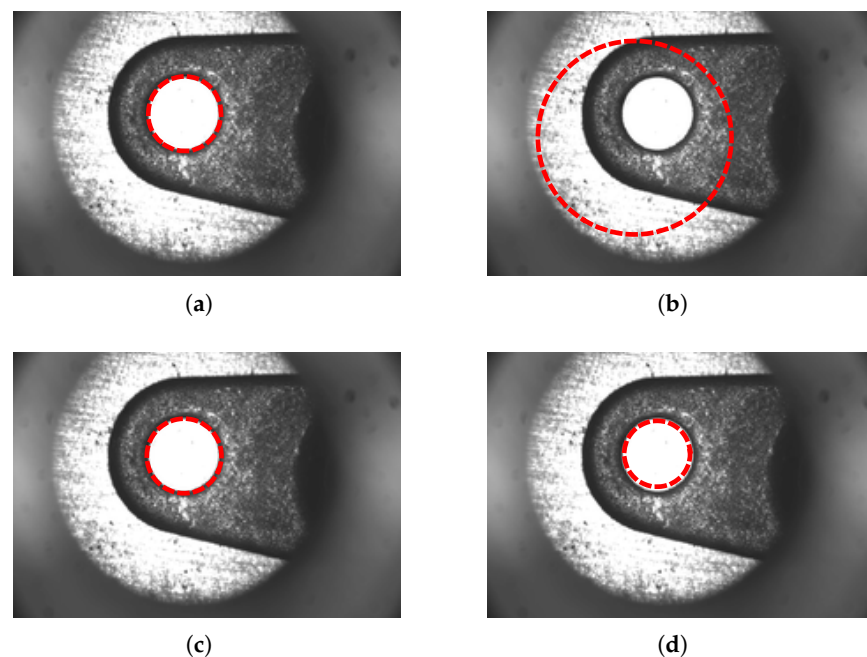


Figure 12. Final results for test image Bearing (150×220): (a) Proposed Method ($\text{Ind} = 0.86$). (b) Center-Based Circle Detection ($\text{Ind} = 0.52$). (c) Hough Circle Detection ($\text{Ind} = 0.93$). (d) RANSAC Circle Detection ($\text{Ind} = 1$).

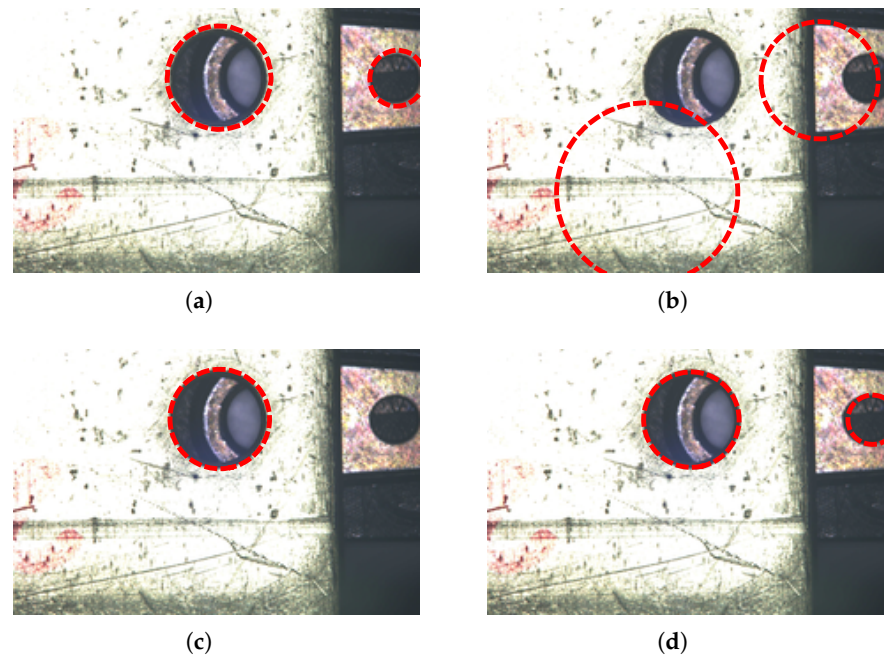


Figure 13. Final results for test image Bearing and hole (150×230): (a) Proposed Method (Ind = 0.86). (b) Center-Based Circle Detection (Ind = 0.13). (c) Hough Circle Detection (Ind = 0.94). (d) RANSAC Circle Detection (Ind = 0.94).

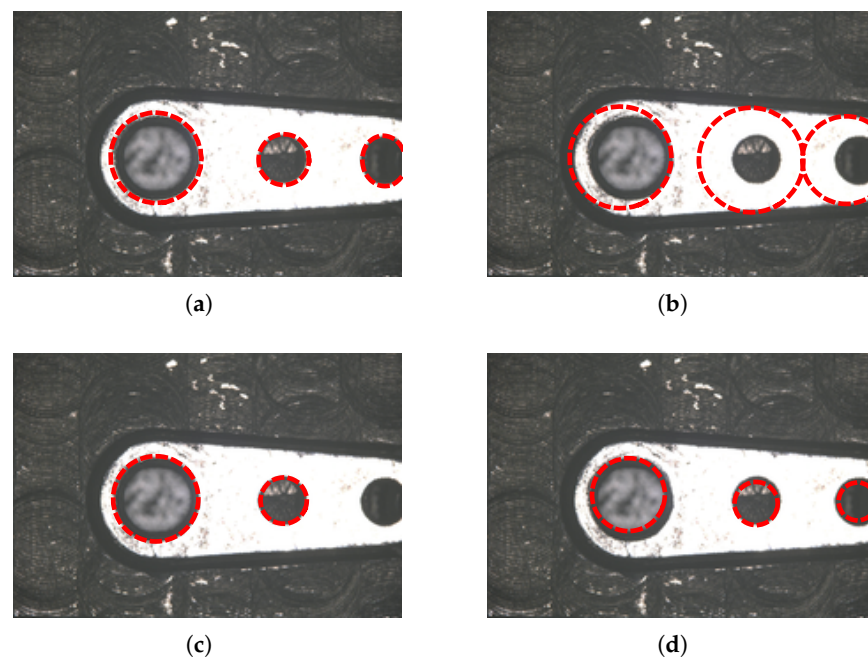


Figure 14. Final results for test image Bearing and holes (150×220): (a) Proposed Method (Ind = 0.71). (b) Center-Based Circle Detection (Ind = 0.21). (c) Hough Circle Detection (Ind = 0.94). (d) RANSAC Circle Detection (Ind = 0.92).

4. Discussion

This paper presents a modification of the standard iterative k -means clustering method. The Mahalanobis-Constrained Hard-Clustering Algorithm is constructed to omit data from the tails of the Gaussian low probability that is achieved by the fact that Mahalanobis distance follows a chi-squared distribution; adjusting the p -quantile controls exclusion of points in the component tails. In this respect, the proposed approach departs from

common alternatives that rely on robust estimators for parameter estimation within the iterative procedure, enabling effective performance in highly noisy data or noisy-structure scenarios where other methods may fail. Further development is possible by incorporating robust estimators and designing alternative models and constraints, which would expand the range of applications and research directions [21,22,24,25,31]. Additionally, modeling various types of indices supports further study of issues such as determining the optimal number of components or clusters, which presents a particularly challenging problem in noisy data [3,10].

5. Conclusions

Building on the well-known k -means algorithm and Mahalanobis-distance-based variants, we developed and examined a center-based iterative method constrained by the Mahalanobis distance. Numerical experiments show that the proposed Mahalanobis-Constrained Hard-Clustering Algorithm effectively handles highly noisy data and a range of noisy data-structure models. Its application to detecting spherical patterns in multi-scene test images confirmed the method's effectiveness at identifying circular patterns under severe noise. This work used the Wolfram Cloud (Wolfram Research, Champaign, IL, USA). Available at: <https://www.wolframcloud.com/>.

Funding: This research paper was funded by the University of Slavonski Brod through the institutional research project Smartmod4IND45, financed by the European Union—NextGenerationEU. The views and opinions expressed in this paper are those of the author and do not necessarily reflect the official position of the European Union or the European Commission. Neither the European Union nor the European Commission can be held responsible for them.

Data Availability Statement: Test image datasets publicly available at <https://datasetninja.com/holes-object-detection> (accessed on 8 May 2026) were used. All analyzed data originate from that source and can be accessed at the stated address subject to the source's terms of use. No new data were generated beyond the referenced test images.

Acknowledgments: During the preparation of this manuscript, the author used GPT-5-MINI for purposes of generating text and locating relevant information. The author has reviewed and edited the output and take full responsibility for the content of this publication.

Conflicts of Interest: The author declares no conflicts of interest.

References

1. Gan, G.; Ma, C.; Wu, J. *Data Clustering, Theory, Algorithms and Applications*; SIAM: Philadelphia, PA, USA, 2007.
2. Gonzalez, R.C.; Woods, R.E. *Digital Image Processing*, 4th ed.; Pearson: New York, NY, USA, 2018.
3. Scitovski, R.; Sabo, K.; Martínez-Álvarez, F.; Ungar, Š. *Cluster Analysis and Applications*; Springer: Cham, Switzerland, 2021.
4. Theodoridis, S.; Koutroumbas, K. *Pattern Recognition*; Academic Press: Cambridge, MA, USA, 2009.
5. Cuesta-Albertos, J.A.; Gordaliza, A.; Matrán, C. Trimmed k -means: An attempt to robustify quantizers. *Ann. Stat.* **1997**, *25*, 553–576. [[CrossRef](#)]
6. Grbić, R.; Grahovac, D.; Scitovski, R. A method for solving the multiple ellipses detection problem. *Pattern Recognit.* **2016**, *60*, 824–834. [[CrossRef](#)]
7. Marošević, T. Data clustering for circle detection. *Croat. Oper. Res. Rev.* **2014**, *5*, 15–24. [[CrossRef](#)]
8. Marošević, T.; Scitovski, R. Multiple ellipse fitting by center-based clustering. *Croat. Oper. Res. Rev.* **2015**, *6*, 43–53. [[CrossRef](#)]
9. Morales-Esteban, A.; Scitovski, R.; Sabo, K.; Grahovac, D.; Ungar, Š. Earthquake analysis of clusters of the most appropriate partition. *J. Seismol.* **2024**, *28*, 1373–1388. [[CrossRef](#)]
10. Novoselac, V. Investigation of the optimal number of clusters by the adaptive EM algorithm. *Croat. Oper. Res. Rev.* **2019**, *10*, 1–12. [[CrossRef](#)]
11. Peel, D.; McLachlan, G. J. McRobust mixture modelling using the t -distribution. *Stat. Comput.* **2000**, *10*, 339–348. [[CrossRef](#)]
12. Sabo, K.; Grahovac, D.; Scitovski, R. Incremental method for multiple line detection problem—Iterative reweighted approach. *Math. Comput. Simul.* **2020**, *178*, 588–602. [[CrossRef](#)]

13. Sabo, K.; Scitovski, R.; Ungar, Š. Multiple spheres detection problem—Center based clustering approach. *Pattern Recognit. Lett.* **2023**, *176*, 34–41. [\[CrossRef\]](#)
14. Sabo, K.; Scitovski, R.; Ungar, Š.; Tomljanović, Z. A method for searching for a globally optimal k-partition of higher-dimensional datasets. *J. Glob. Optim.* **2024**, *89*, 633–653. [\[CrossRef\]](#)
15. Scitovski, R.; Marošević, T. Multiple circle detection based on center-based clustering. *Pattern Recognit. Lett.* **2014**, *52*, 9–16. [\[CrossRef\]](#)
16. Scitovski, R.; Sabo, K.; Grahovac, D.; Martínez-Álvarez, F.; Ungar, Š. A partitioning incremental algorithm using adaptive Mahalanobis fuzzy clustering and identifying the most appropriate partition. *Pattern Anal. Appl.* **2025**, *28*, 3. [\[CrossRef\]](#)
17. Scitovski, R.; Sabo, K.; Grahovac, D.; Ungar, Š. Minimal distance index—A new clustering performance metrics. *Inf. Sci.* **2023**, *640*, 119046. [\[CrossRef\]](#)
18. Scitovski, R.; Sabo, K.; Nikić, P.; Majstorović, S. A new efficient method for solving the multiple ellipse detection problem. *Expert Syst. Appl.* **2023**, *222*, 119853. [\[CrossRef\]](#)
19. Scitovski, R.; Scitovski, S. A fast partitioning algorithm and its application to earthquake investigation. *Comput. Geosci.* **2013**, *59*, 124–131. [\[CrossRef\]](#)
20. Alakkari, K.; Abotaleb, M.; El-kenawy, E.-S.M.; Mishra, P. Advanced Deep Statistical Learning Approach for Forecasting Global Economic Policy Uncertainty and Volatility. *Comput. Econ.* **2026**. [\[CrossRef\]](#)
21. Xie, J.; Girshick, R.; Farhadi, A. Unsupervised deep embedding for clustering analysis. In Proceedings of the International Conference on Machine Learning, New York, NY, USA, 19–24 June 2016; pp. 478–487. [\[CrossRef\]](#)
22. Caron, M.; Bojanowski, P.; Joulin, A.; Douze, M. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018; Springer: Berlin/Heidelberg, Germany, 2018; pp. 132–149. [\[CrossRef\]](#)
23. Johnson, R.A.; Wichern, D.W. *Applied Multivariate Statistical Analysis*, 6th ed.; Pearson Prentice Hall: Upper Saddle River, NJ, USA, 2007.
24. Filzmoser, P.; Gregorich, M. Multivariate Outlier Detection in Applied Data Analysis: Global, Local, Compositional and Cellwise Outliers. *Math. Geosci.* **2020**, *52*, 1049–1066. [\[CrossRef\]](#)
25. Rousseeuw, P. J.; Leroy, A.M. *Robust Regression and Outlier Detection*; John Wiley & Sons: New York, NY, USA, 2003.
26. Scitovski, R.; Radojičić, U.; Sabo, K. A fast and efficient method for solving the multiple line detection problem. *Rad Hrvat. Akad. Znan. Umjet.* **2019**, *23*, 123–140. [\[CrossRef\]](#)
27. Krpić, Z.; Martinović, G.; Vazler, I. Data clustering: Applications in engineering. *Croat. Oper. Res. Rev.* **2010**, *1*, 180–189.
28. Taler, P.; Sabo, K. Color image segmentation based on intensity and hue clustering—A comparison of LS and LAD approaches. *Croat. Oper. Res. Rev.* **2014**, *5*, 375–385. [\[CrossRef\]](#)
29. Fischler, M.A.; Bolles, R.C. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM* **1981**, *24*, 381–395. [\[CrossRef\]](#)
30. Yuen, H.K.; Princen, J.; Illingworth, J.; Kittler, J. Comparative study of Hough Transform methods for circle finding. *Image Vis. Comput.* **1990**, *8*, 71–77. [\[CrossRef\]](#)
31. Mayrhofer, M.; Radojičić, U.; Filzmoser, P. Robust Covariance Estimation and Explainable Outlier Detection for Matrix-Valued Data. *Technometrics* **2025**, *67*, 516–530. [\[CrossRef\]](#)
32. Tafro, A.; Seršić, D. Gaussian Mixture Estimation from Lower-Dimensional Data with Application to PET Imaging. *Mathematics* **2024**, *12*, 764. [\[CrossRef\]](#)
33. Hubert, L.; Arabie, P. Comparing partitions. *J. Classif.* **1985**, *2*, 193–218. [\[CrossRef\]](#)
34. Strehl, A.; Ghosh, J. Cluster ensembles—A knowledge reuse framework for combining multiple partitions. *J. Mach. Learn. Res.* **2002**, *3*, 583–617. [\[CrossRef\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.