Computational Statistics (2024) 39:2677–2701
https://doi.org/10.1007/s00180-023-01400-1

ORIGINAL PAPER

Classifying for images based on the extracted probability density function and the quasi Bayesian method

Hieu Huynh-Van[1,2,3] · **Tuan Le-Hoang**[2,4] · **Tai Vo-Van**[5]

Received: 25 January 2023 / Accepted: 6 August 2023 / Published online: 31 August 2023

Abstract

This study presents a novel algorithm for image classification based on a quasi-Bayesian approach and the extraction of probability density functions (PDFs). First, representative PDFs are extracted from each image using its features. Next, a measure is developed to evaluate the similarity between the extracted PDFs. Finally, an algorithm is established for determining prior probabilities using fuzzy clustering techniques. By combining these improvements, we develop a more efficient algorithm for classifying image data. An image is assigned to a specific group if it has the highest value of prior probability and a similar level to that group. We explain the proposed algorithm step-by-step with a numerical example and clearly demonstrate its convergence. When applied to multiple image datasets, the proposed algorithm has shown stability and efficiency, outperforming many other statistical and machine learning methods. Additionally, we have developed a Matlab procedure to apply the proposed algorithm to real image datasets. These applications demonstrate the potential of research in various fields related to the digital revolution and artificial intelligence.

Keywords Classification problem · Empirical error · Prior probability · Probability density function

1 Introduction

Classification involves assigning an element to an appropriate group based on observed variables. It is currently a significant development direction in multidimensional statistics and data science, with applications in many fields, attracting the interest of statisticians and information technology professionals (Pham-Gia et al. 2000; Nhu et al. 2020; Vovan et al. 2021). However, the classification problem remains unsolved. Image classification, specifically, involves determining the

Hieu Huynh-Van, Tuan Le-Hoang and Tai Vo-Van have contributed equally to this work.

Extended author information available on the last page of the article

label for an image based on a built classification model from a training set. This is typically achieved through two steps: extracting features from the image and creating a specific classification model using these extracted features.

The critical first step in solving the classification problem for images is feature extraction (Laleh et al. 2019). Feature extraction is considered effective if the extracted values help distinguish images from each other and reduce the time required for classification (Celebi and Alpkocak 2000; Park et al. 2004). Normally, images are extracted based on color, texture, and shape features (Vovan et al. 2017; Nguyentrang and Vovan 2017; VijayaLakshmi and Mohan 2016). Although there has been much discussions on this issue, no method is considered optimal for all cases. Once the features are extracted, a representative element must be chosen for the image to apply to the classification problem. This choice is significant because image recognition is performed based on these representative elements. Currently, three main objects are used to represent an image: a numerical matrix, a PDF, and an interval (Vovan et al. 2019, 2021). For example, Garg and Dhiman (2021) used texture features to extract a matrix for each image, while Che-Ngoc et al. (2022) used the maximum function of PDFs built from texture features to classify a person's face. Moreover, in recent years, many authors have used the interval element to represent an image, such as Phamtoan and Vovan (2020), Phamtoan et al. (2022), and Lethikim et al. (2023). Each type of extraction has its advantages for specific cases depending on the image's features, and there is no extraction type that is optimal for all cases.

Regarding the classification problem, there are several commonly used methods today, including traditional statistical methods such as Fisher (Fisher 1938), Logistic regression (Kung et al. 2010; Scott 2015), Naive Bayes (based on discrete variables) (Nhu et al. 2020), and improved Bayes (based on continuous variables) (Nguyentrang and Vovan 2017). Fisher's method can classify two or more groups, but their covariance matrices must be the same, limiting its practical applications where actual data rarely satisfies this condition (Vovan 2016). Logistic regression is a popular method but only effective when the data have a good separation of groups and the dependent variable is binary (Yuan et al. 2012). The improved Bayesian method does not require the condition of the data and can classify many groups (Vovan 2016; Nguyentrang and Vovan 2017), making it theoretically advantageous. However, it has some disadvantages in real-world applications at present (Vovan et al. 2019; Vovan 2018; Vovan et al. 2017). The improved Bayesian method is a popular classification method due to its ability to classify many groups and the absence of data requirements. However, it has several disadvantages that limit its practical application. Firstly, the determination of prior probability is often subjective and not always suitable for all datasets. Several statistical approaches have been proposed, including uniform distribution, sample rate, and the Laplace method. Additionally, Nguyentrang and Vovan (2017) proposed an algorithm to determine the prior probability based on fuzzy cluster analysis. Nonetheless, these methods are only suitable for specific datasets. Secondly, estimating the representative PDF for each image and group is still a challenging task, despite many improvements. Finally, the computational complexity, such as finding the maximum function and integrating it in

multi-dimensional space, poses a significant barrier to the practical application of this method (Vovan and Pham-Gia 2010; Vovan 2018).

There are various classification methods available with machine learning and deep learning techniques. Quadratic Discriminant Analysis (QDA) is an extension of the Fisher method, where each group is allowed to have its own covariance matrix, unlike Fisher, which assumes a common covariance matrix for all groups (Chen et al. 2016). XGBoost is a popular supervised learning algorithm that uses sequentially-built decision trees to provide accurate results and a highly scalable training method that prevents overfitting (Behera et al. 2021). However, XGBoost is not suitable for data that has significant overlap between groups. Support Vector Machines (SVM) are effective classification tools for high-dimensional data but can be unstable and have high error rates for datasets containing irregular elements (Huang et al. 2018; Shawe 2000). Artificial Neural Network (ANN) is a powerful method that can detect complex nonlinear relationships between variables, but it is prone to overfitting and requires high computational cost. Additionally, the initial parameter settings depend on the researcher's experience (Neto et al. 2021). The Random Tree method is a supervised machine learning classifier that creates a group of decision trees and uses the most frequent tree outputs as the final classification. This method requires significant training time for a larger number of trees. Bagged classifiers are ensemble meta-estimators that aggregate forecasts from classifiers trained on random subsets of the original dataset. However, like the Random Tree method, Bagged classifiers also require significant training time (Dietterich 2000). Adaptive Boosting is a statistical classification meta-algorithm that can be used with other learning algorithms to enhance performance (Wyner et al. 2017). K-nearest neighbor (kNN) is a complex method that can achieve high classification performance. However, kNN heavily relies on selecting parameters and distance types for each dataset, and computing distances between the element to be classified and all parts of the training set can be time-consuming (Imandoust and Bolandraftar 2013). Subspace-kNN combines predictions from multiple decision trees trained on different subsets. However, it also has the same computational disadvantage as kNN (Gou et al. 2012).

Some other methods were used for image classification. Azimbagirad (2020) proposes the use of the Tsallis-Entropy segmentation method to enhance the accuracy and performance of classifying MRI brain images. Furthermore, Azimbagirad and Junior (2021) continued to use the Tsallis-entropy method to estimate the parameters in the Gaussian model and apply it to the brain structure classification problem in medicine. Renukadevi et al. (2022) utilizes Fourier and wavelet transforms to classify brain images based on time-extracted frequencies. These methods have yielded good results for specific cases, but they have not explored other applications. To classify image objects, most of the above methods require extracting features into numerical matrices, which can be time-consuming and limiting for real-world applications in areas such as automation and artificial intelligence. Additionally, their classification accuracy has limitations in many cases. In recent years, some authors have proposed representing image features as two-dimensional intervals for clustering and classification (Huynh-Van et al. 2023; Lethikim et al. 2022; Phamtoan and Vovan 2023). While these methods

have shown promising results for specific datasets, they are often outperformed by deep learning algorithms. However, researchers have not yet explored the concept of utilizing PDFs to represent images for classification.

This study proposes a novel image classification method based on the extracted features with the following contributions:

(i) Developing a new measure to evaluate the similarity of PDFs. This measure is used as an effective criterion for classifying the images.
(ii) Proposing a method that utilizes a PDF estimated from the extracted grayscale to represent each image. This method can effectively reduce computational costs and the false classification rate.
(iii) Building an algorithm to determine the prior probability based on fuzzy relationships between the classified element and groups.
(iv) Establishing a new classification rule, where an image is assigned to a group if it has the highest values of both prior probability and similarity to that group.

The proposed method is described step by step using a specific image set and can be efficiently implemented through the established Matlab procedure. Additionally, the complex proofs for the algorithm's convergence have been solved. Moreover, this study demonstrates the advantages of the proposed algorithm in comparison to existing algorithms using many real image datasets.

The remaining sections of the article are organized as follows. Section 2 discusses some challenges related to the proposed algorithm and presents methods for estimating image features. Section 3 presents the proposed algorithm in detail, including a numerical example. This section also provides proof of the convergence of the proposed algorithm. Section 4 applies and compares the proposed algorithm to multiple image sets with diverse characteristics. Finally, the conclusion summarizes the main findings of the study.

2 Some related issues to the proposed algorithm

2.1 Similarity of probability density functions

Let $G = \{g_1, g_2, ..., g_k\}$ be PDFs in $R^n, n \geq 1, k \geq 2$, and $g_{\max}(x) = \max\{G(x)\}$. Then, we have the following definitions:

Definition 1 Similar level of cluster (SLC) G is defined by Eq. (1):

$$S(g_1, g_2, ..., g_k) = \frac{k}{k-1}\left(1 - \frac{1}{k}\int_{R^n} g_{\max}(x)dx\right). \quad (1)$$

When $k = 1, S(g_1)$ is defined as 1.

We have

$$g_i(x) \le \max\left\{g_1(x), g_2(x), ..., g_k(x)\right\} \le \sum_{i=1}^{k} g_i(x).$$

Because $\int_{R^n} g_i(x)dx = 1$,

$$1 \le \int_{R^n} g_{\max}(x)dx \le k \text{ or } 0 \le 1 - \frac{1}{k}\int_{R^n} g_{\max}(x)dx \le 1 - \frac{1}{k}.$$

From (1), multiply both sides of the above inequality by $\frac{k}{k-1}$, we can deduce the result as follows:

$$0 \le S(g_1, g_2, ..., g_k) \le 1. \tag{2}$$

The function $S(.)$ measures the overlap between PDFs and is normalized on the range [0,1]. A larger value of $S(.)$ indicates a greater degree of overlap between the PDFs. Specifically, when the PDFs are separate, $S(.) = 0$, and when they completely overlap, $S(.) = 1$. Therefore, $S(.)$ can be used as a measure to evaluate the dissimilarity of PDFs.

Previous studies, such as those by Vovan and Pham-Gia (2010) and Nguyentrang and Vovan (2017) have proposed using cluster width to measure the similarity of PDFs. However, cluster width depends on the number of PDFs in clusters, limiting its use for comparing clusters with different numbers of elements. In contrast, we believe that $S(.)$ can overcome this drawback because it can evaluate the similarity of clusters with different numbers of PDFs.

Definition 2 The representative PDF of cluster G is defined by Eq. (3).

$$g_G = \frac{\sum_{i=1}^{k} (\mu_{Gi})^2 g_i}{\sum_{i=1}^{k} (\mu_{Gi})^2}, \tag{3}$$

where μ_{Gi} is the probability of placing g_i in the cluster G.

From Eq. (3), we can quickly check that g_G is non-negative function, and $\int g_G(x)dx = 1$. It also means that g_G is a PDF.

2.2 Estimate the probability density function

It states that there are many methods available for estimating PDF, including both parametric and non-parametric approaches (Dempster et al. 1977; Nguyentrang and Vovan 2017; Che-Ngoc et al. 2022). This study utilizes the kernel function method, a non-parametric technique that is currently advantageous for estimating PDF.

Let $\left\{x_1, x_2, ..., x_n\right\}$ be n-dimensional discrete data. The estimated PDF by the kernel method has the following form:

$$\widehat{f}(x) = \frac{1}{Nh_1h_2...h_n} \sum_{i=1}^{N} \prod_{j=1}^{n} K_j\left(\frac{x_i - x_{ij}}{h_j}\right), \tag{4}$$

where h_j is the smooth parameter of the jth variable, $K_j(.)$ is the kernel function of the jth variable with

$$\begin{cases} K(x) \geq 0 \\ \int K(x)dx = 1. \end{cases}$$

There are many kernel functions and smooth parameters proposed to apply in Eq. (4). According to the literature, there are no kernel functions and smooth parameter to optimize for all data (Nguyentrang and Vovan 2017). In this study, we use the normal kernel function (Terrell and Scott 1992): $K(x) = \frac{1}{\sqrt{2\pi}}\exp(-\frac{x^2}{2})$, and the smooth parameter widely used in many studies (Nguyentrang and Vovan 2017; Vovan et al. 2017; Vovan et al. 2022).

2.3 The prior probability

In the classification problem by the Bayesian method, finding the prior probability is the critical problem. For discrete elements, although many authors are interested in this problem, such as Pham-Gia et al. (2000), Miller (2001), finding the appropriate prior probability for the specific cases is still an unsolved problem. At present, there are the following popular methods for determining prior probability:

i) Uniform distribution:

$$q_1 = q_2 = ... = q_k = \frac{1}{k}, \tag{5}$$

ii) Sample rate:

$$q_i = \frac{n_i}{N}, \tag{6}$$

iii) Laplace method:

$$q_i = \frac{n_i + n/k}{N + n}, \tag{7}$$

where n_i is the number of elements in w_i, n is the number of dimensions, k is the number of clusters, and N is the number of elements for all groups.

It can be seen that the method (i) is difficult to convince as a good one because it does not base on any information. The methods (ii) and (iii) are only based on the sample set without considering the relationship between the classified element and the groups. Based on fuzzy clustering technique (Maronna et al. 2016), Nguyentrang

and Vovan (2017) proposed the new method to determine the prior probability. However, it is only applied for discrete elements.

3 The proposed algorithm

3.1 The representative PDF for image

A pixel is the smallest unit of a digital image or graphic that can be displayed and represented on a digital display device.

RGB color scales refer to the Red-Green-Blue color model, which is a commonly used system for representing and displaying colors on electronic devices such as computers and televisions. In this model, colors are created by combining varying intensities of red, green, and blue light. Each color channel typically ranges from 0 to 255, representing the intensity of that color component. By adjusting the values of the three channels, we can be achieved $256 \times 256 \times 256 = 16,777,216$ different colors.

A gray image, also known as a grayscale image or black-and-white image, is an image where each pixel is represented by a single intensity value. Unlike color images that have multiple color channels (such as RGB), a grayscale image has only one channel. The intensity value of each pixel in a gray image typically ranges from 0 (black) to 255 (white), with intermediate values representing different shades of gray. In a gray image, there is no color information, and the image appears in varying shades of gray depending on the intensity values of the pixels.

In this study, color images will be converted to grayscale to reduce computational complexity while preserving recognition characteristics. Each grayscale image with dimensions of $m \times n$ pixels will be feature-extracted into a numerical matrix of size $m \times n$. This matrix will be reshaped into a column of data and a kernel function method will be used to estimate it as a representative PDF for the image. An illustration of this process is shown in Fig. 1.

3.2 The algorithm

Let $w_1, w_2, \ldots, w_k$ be k groups so that the ith group has n_i images, $n_1 + n_2 + \ldots + n_k = N$. Let I_0 be the image that we need to classify. Then, the proposed algorithm has the following steps:

- **Step 1.** Extract the grayscale of images in k groups and the classified image. Each image is transformed into a column vector with a number of elements equal to the size of the image multiplied by its pixel resolution. For instance, an image with a resolution level of $m \times n$ should be transformed into a column vector with $m \times n$ elements.
- **Step 2.** Estimate the PDF for each group by the kernel function method according to Eq. (4). Let g_j, $j = 1, 2, \ldots, N$ and g_0 be the set of PDFs extracted for groups and I_0, respectively.

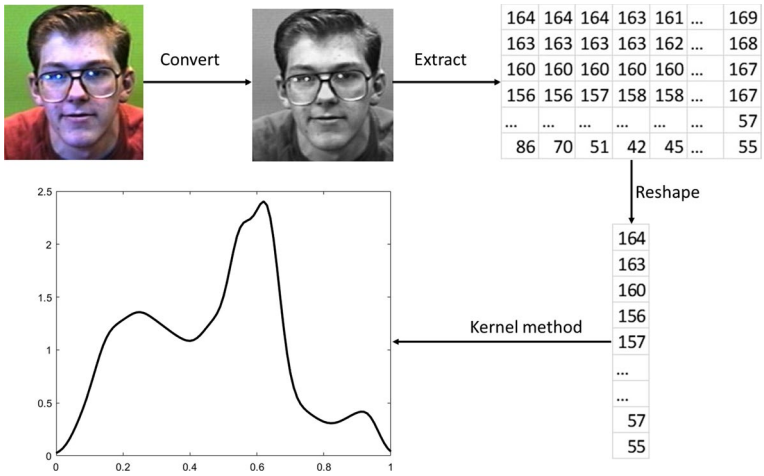


Fig. 1 Illustration for extracting representative PDF for an image

- **Step 3.** Set up the initial probability matrix $U^{(0)} \in M$(M is fuzzy partition space) with k rows and $N+1$ columns. This matrix can be chosen randomly. However, for the convenience in application, this study sets up the initial matrix as follows:

$$U^{(0)} = \left[\mu_{ij}^{(0)}\right]_{k\times(N+1)} = \begin{bmatrix} \mu_{11}^{(0)} & \mu_{12}^{(0)} & \cdots & \mu_{1N}^{(0)} & \frac{1}{k} \\ \mu_{21}^{(0)} & \mu_{22}^{(0)} & \cdots & \mu_{2N}^{(0)} & \frac{1}{k} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \mu_{k1}^{(0)} & \mu_{k2}^{(0)} & \cdots & \mu_{kN}^{(0)} & \frac{1}{k} \end{bmatrix}. \tag{8}$$

The matrix set up by (8) has the first N columns as non-fuzzy partitions for the elements to groups $w_1, w_2, ..., w_k$. Specifically, $\mu_{ij}^{(0)} = 1$ when the jth element belongs to the ith cluster, and $\mu_{ij}^{(0)} = 0$ when the jth element does not belong to the ith cluster. The last column $(N+1)$ is the initial probability to assign I_0 to groups $w_i, i = 1, 2, ..., k$. In this step, the same prior probability of I_0 is given in the first iteration. It means that the first prior probability of I_0 to the groups is $1/k$. Find the representative PDF $g_{v_i}^{(0)}$ for each group $w_i, i = 1, 2, ..., k$ by Eq. (3).

- **Step 4.** Update the new probability matrix $U^{(t)}$, where its elements are given by (9).

$$\mu_{ij}^{(t)} = \begin{cases} \frac{1}{\sum_{c=1}^{k} \left[(1-S(g_{v_i}^{(t)}, g_j))/(1-S(g_{v_i}^{(t)}, g_c))\right]^2} & \text{if } S(g_{v_i}^{(t)}, g_j) < 1 \\ 0 & \text{otherwise,} \end{cases} \tag{9}$$

where $i = 1, 2, \cdots k, j = 1, 2, \cdots, N$, $V^{(t)} = \{g_{v_1}^{(t)}, g_{v_2}^{(t)}, ..., g_{v_k}^{(t)}\}$ is the set of the representative elements of k groups (g_{v_i} is calculated by (3)), and $S(g_{v_i}^{(t)}, g_j), i = 1, 2, ..., k; j = 1, 2, ..., N$ is calculated by (1).

- **Step 5.** Calculate the following value:

$$\|J^{(t+1)}(U,V) - J^{(t)}(U,V)\|, \tag{10}$$

with $J^{(t)}(U,V) = \sum_{i=1}^{k}\sum_{j=1}^{N}[\mu_{ij}^{(t)}]^2.S^2(g_j, g_{v_i}^{(t)})$.

- **Step 6.** Repeat Step 3 and Step 4 t times until the following condition is satisfied:

$$\|J^{(t+1)}(U,V) - J^{(t)}(U,V)\| < \varepsilon, \tag{11}$$

where ε is an arbitrarily small positive number. It measures the different of $J^{(t)}$ through two consecutive iterations. In this study, ε is chosen as 0.0001 for the examples and applications. When the algorithm ends, we will obtain the matrix of size $k \times (N+1)$, where the sum of each column always equals 1. The last column in this matrix is used as the priori probability to assign I_0 into the groups:

$$q = \begin{bmatrix} q_1 \\ q_2 \\ \cdots \\ q_k \end{bmatrix} = \begin{bmatrix} \mu_{1(N+1)}^{(t)} \\ \mu_{1(N+1)}^{(t)} \\ \cdots\cdots\cdots \\ \mu_{k(N+1)}^{(t)} \end{bmatrix}.$$

- **Step 7.** Classify g_0 to the groups based on the following principle: g_0 will be classified to group w_m if the following condition is satisfied:

$$\max\left\{q_i.S\left(g_0, g_{v_i}\right)\right\} = q_m.S\left(g_0, g_{v_m}\right); i, m = 1, 2, ..., k. \tag{12}$$

The flowchart of the proposed algorithm is given in Fig. 2.

We have the following comments about the proposed algorithm:

(i) Step 3, Step 4, Step 5 and Step 6 find the prior probability for the classified image I_0. It is based on the fuzzy relationship between the classified element with the groups. In our opinion, it is more suitable than the traditional methods presented in Section 2.3.

(ii) The proposed method is considered to be an improvement of the traditional Bayesian classifier for the discrete element. Image I_0 will be classified to Group w_i if it has the maximum value of the prior probability and the similarity level ($S(.)$) to this group.

(iii) To apply the proposed algorithm, we have developed a program in Matlab software. This program can effectively process for real images. When provided with a training dataset, it converts all the images to grayscale for feature extraction. It then estimates the PDF for each image and finds the representative PDF for groups. For a classified image, the program automatically estimates the PDF based on grayscale, finds the prior probability, computes the SLC with the groups, and applies the proposed classification rule. The examples and applications in this study are implemented by the established Matlab program.

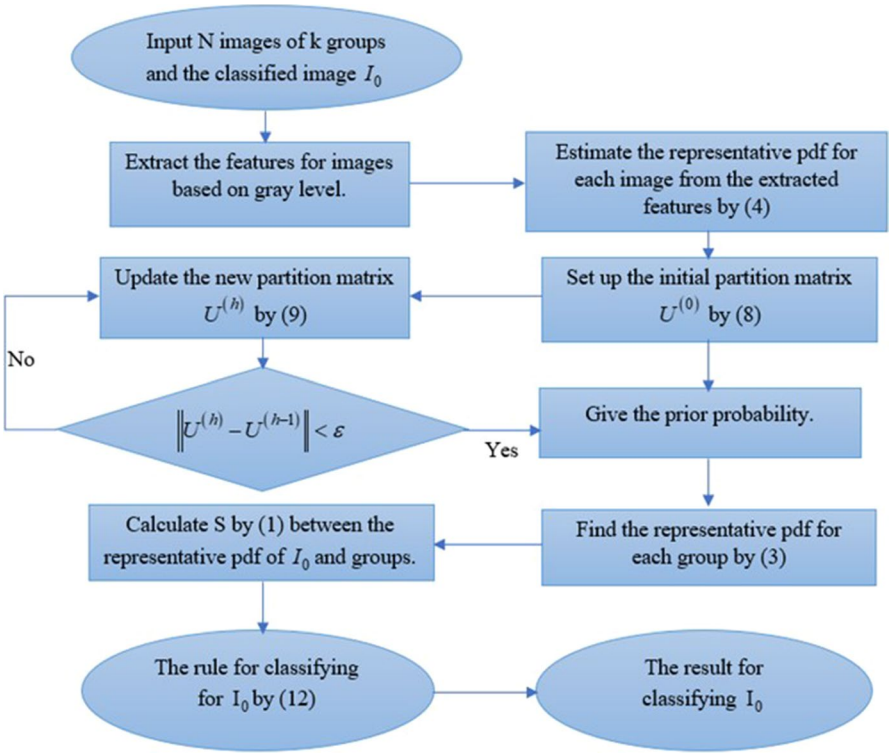


Fig. 2 The flowchart of the proposed algorithm

3.3 The convergence

The convergence of the proposed algorithm is shown from Step 3 to Step 6. It is proven by the following theorem:

Theorem 1 *Let* $V^{(t)} = \{g_{v_1}^{(t)}, g_{v_2}^{(t)}, ..., g_{v_k}^{(t)}\}$ *and* $U^{(t)} = \left[\mu_{ij}^{(t)}\right]_{k\times N}$ *denote the prototype elements and the probability matrix at iteration t of the proposed algorithm, respectively. We define* $J^{(t)}(U,V) = \sum_{i=1}^{k}\sum_{j=1}^{N}[\mu_{ij}^{(t)}]^2.S^2(g_j, g_{v_i}^{(t)})$, *where* $S^2(g_j, g_{v_i}^{(t)})$ *is the squared distance between* g_j *and* $g_{v_i}^{(t)}$. *It can be shown that* $J^{(t)}$ *will converge after several iterations from Step* 3 *to Step* 6 *of the proposed algorithm.*

Proof For solving this problem, we will prove three following results:

(i) Let $\varphi_1(U) = J(V,U)$ with the fixed V. When $S(g_j, g_{v_i}^{(t)}) > 0, \forall i = 1,2,\ldots,k; \forall j = 1,2,\ldots,N$, $\varphi_1(U)$ reaches its minimum value if and only if U is computed by (9).

(ii) Let $\varphi_2(V) = J(V, U)$ with the fixed U. When $S(g_j, g_{v_i}^{(t)}) > 0, \forall i = 1, 2, \ldots, k; \forall k = 1, 2, \ldots, N$, $\varphi_2(V)$ reaches its minimum value if and only if V is computed by (3).

(iii) From (i) and (ii), we have the sequence $\left\{J^{(t)}\right\}_{t=1}^{\infty}$ does not decrease and is bounded, so it converges.

* For (i), set $\mu_{ij} = \xi_{ij}^2$, $\Xi = \left[\xi_{ij}^2\right]_{k\times N}$ and $\phi(\Xi) = J(V, \Xi)$ is the probability matrix and the optimized objective function, respectively. Then, it is known that minimizing the function $\varphi(U)$ over M is a Kuhn-Tucker problem with the inequality and equality constraints as follows:

$$\mu_{ij} \in [0, 1], 0 < \sum_{j=1}^{N} \mu_{ij} < 1, \sum_{i=1}^{k} \mu_{ij} = 1. \tag{13}$$

By releasing the inequality constraints in (13), we can solve the above problem by the Lagrange multiplier method. Let $\alpha = \left(\alpha_1, \alpha_2, \ldots, \alpha_N\right)$ as Lagrange multipliers, then the Lagrange function is defined by

$$\phi(\Xi, \alpha) = \sum_{i=1}^{k}\sum_{j=1}^{N} \xi_{ij}^2 [S(g_j, g_{v_i}^{(0)}]^2 + \sum_{j=1}^{N} \alpha_j \left(\sum_{i=1}^{k} \xi_{ij}^2 - 1 \right).$$

To find the breakpoint of the function ϕ, we solve the following system of equations:

$$\frac{\partial \phi}{\partial \alpha_j} = \sum_{i=1}^{k} \xi_{ij}^2 - 1 = 0; j = 1, 2, \ldots, N, \tag{14}$$

$$\frac{\partial \phi}{\partial \xi_{lq}} = 2\xi_{lq} S(gv_l, g_q)^2 = 0; l = 1, 2, \ldots, k; \; q = 1, 2, \ldots, N. \tag{15}$$

From(15), we obtain

$$\xi_{lq}^2 = \mu_{lq} = \left(\frac{\alpha_q}{2S(g_{v_l}, g_q)^2} \right)^{-1}. \tag{16}$$

Take the sum (16) by l, and apply (14), we get

$$-\alpha_q = \frac{1}{\sum_{l=1}^{k} \left(\frac{1}{2S(g_{v_l}, g_q)^2} \right)}. \tag{17}$$

Substituting (17) into (16), we get U calculated by (3) which is the breakpoint of the function ϕ.

Continuing to consider the second partial partial derivatives of the Lagrange function, we have

$$\frac{\partial^2 \phi}{\partial \xi_{rs} \partial \xi_{lq}} = \begin{cases} 4\xi_{lq}^2 S^2(f_{v_l}, f_q) & \text{if} \quad r = l, s = q, \\ 0 & \text{for the opposite cases.} \end{cases}$$

Using (9) and (17) to compute the non-zero elements in the Hessian matrix, H_ϕ, we have

$$\lambda_q = \frac{\partial^2 \phi}{\partial \xi_{lq}^2} = 8\left(\sum_{j=1}^{k} S^2(fv_j, f_q) \right).$$

Because $m > 1$ and $S(f_{v_j}, f_q) > 0, \forall j, q$, $\lambda_q > 0, \forall q$.

Then, we have

$$H_\phi = \begin{pmatrix} \frac{\partial^2 \phi}{\partial \xi_{11}^2} & \cdots & \frac{\partial^2 \phi}{\partial \xi_{11} \partial \xi_{1N}} & \cdots & \frac{\partial^2 \phi}{\partial \xi_{11} \partial \xi_{N1}} & \cdots & \frac{\partial^2 \phi}{\partial \xi_{11} \partial \xi_{NN}} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \frac{\partial^2 \phi}{\partial \xi_{1N} \partial \xi_{11}} & \cdots & \frac{\partial^2 \phi}{\partial \xi_{1N}^2} & \cdots & \frac{\partial^2 \phi}{\partial \xi_{1N} \partial \xi_{N1}} & \cdots & \frac{\partial^2 \phi}{\partial \xi_{1N} \partial \xi_{NN}} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \frac{\partial^2 \phi}{\partial \xi_{N1} \partial \xi_{11}} & \cdots & \frac{\partial^2 \phi}{\partial \xi_{N1} \partial \xi_{1N}} & \cdots & \frac{\partial^2 \phi}{\partial \xi_{N1}^2} & \cdots & \frac{\partial^2 \phi}{\partial \xi_{N1} \partial \xi_{NN}} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \frac{\partial^2 \phi}{\partial \xi_{NN} \partial \xi_{11}} & \cdots & \frac{\partial^2 \phi}{\partial \xi_{NN} \partial \xi_{1N}} & \cdots & \frac{\partial^2 \phi}{\partial \xi_{NN} \partial \xi_{N1}} & \cdots & \frac{\partial^2 \phi}{\partial \xi_{NN}^2} \end{pmatrix}$$

$$= \begin{pmatrix} \lambda_1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & \lambda_N & \cdots & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 0 & \cdots & \lambda_1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & \lambda_N \end{pmatrix}$$

positive determination. Therefore, U calculated is the solution of the easing optimization problem under consideration. Also, applying $0 < \mu_{ij} < 1, \forall i, j$, we have U computed is also solution of the original minimization problem.

* For (ii)

For each particular point x, let $v_i = f_{v_i}(x)$, $y_j = f_j(x)$ and v be the vectors which contains the v_i. Consider the objective function $\Psi(v) = G(v, U) = \sum_{i=1}^{k} \sum_{j=1}^{N} \mu_{ij}^2 \left(v_i - y_j\right)^2$. We have

$$\frac{\partial \Psi}{\partial v_j} = 2 \sum_{j=1}^{N} \mu_{ij}^2 \left(v_i - y_j\right).$$

Solving the equation $\frac{\partial \Psi}{\partial v_j} = 0$, we have the necessary condition for Ψ to have a minimum of

$$v_j = \frac{\sum_{j=1}^{N} \mu_{ij}^2 y_j}{\sum_{j=1}^{N} \mu_{ij}^2}.$$

Continuing to consider the second partial partial derivatives and using (13), we have

$$\frac{\partial^2 \Psi}{\partial v_r \partial v_s} = \begin{cases} 2\sum_{j=1}^{N} \mu_{rj}^2 > 0 & \text{if} \quad r = s, \\ 0 & \text{for the opposite cases.} \end{cases}$$

$\Rightarrow H_\Psi$ is positive definite. Thus, for each particular point x, $\Psi(v)$ reaches its minimum if and only if

$$g_{v_i}(x) = \frac{\sum_{j=1}^{N} \mu_{ij}^2 g_j(x)}{\sum_{j=1}^{N} \mu_{ij}^2} = g_{v_i}^*(x)$$

or

$$\sum_{i=1}^{k} \sum_{j=1}^{N} \mu_{ij}^2 \left(g_{v_i}^*(x) - g_j(x) \right)^2 = \min \sum_{i=1}^{k} \sum_{j=1}^{N} \mu_{ij}^2 \left(g_{v_i}(x) - g_j(x) \right)^2 \tag{18}$$

Integrating (18), we have (ii).

* For (iii)

For all j, since g_j is continuous, $g_{v_i}^{(t)}$, $S^2(g_j, g_{v_i}^{(t)})$, and $J(V^{(t)}, U^{(t)})$ are also continuous functions for all i, k, t. Also, since $g_{v_i}^{(t)}$ is the weighted average of probability density functions, $g_{v_i}^{(t)}$ belongs to a convex hull H compact $\forall i, t$. From (13), we also have $U^{(t)}$ in the compact space M. Because J is continuous on $H \times M$ compact, J is compact. Furthermore, in the condition $S(g_j, g_{v_i}^{(t)}) > 0, \forall i, k, t$, i) and ii) show $J^{(t+1)} \leq J^{(t)}, \forall t$. The sequence $\left\{ J^{(t)} \right\}_{t=1}^{\infty}$ does not decrease and is bounded, so it converges. □

3.4 The numerical example

This example classifies four butterfly species to demonstrate the steps of the proposed algorithm in detail. The image set is taken from the website https://www.kaggle.com/datasets and consists of 96 images of 4 species: Adonis, Chestnut, Clouded sulphur, and Cabbage. There are 24 images for each species. The image samples of the 4 species are shown in Fig. 3.

In this dataset, we randomly select about 90% of images in each group (86 images) as the training set, and about 10% remaining images as the test set (10 images). The detail for the number of images from each species in the training and test sets are given in Table 1.

The steps of the proposed algorithm are presented as follows:

(a) Adonis (b) Chestnut (c) Clouded sulphur (d) Cabbage

Fig. 3 The sample images of 4 butterfly species

Table 1 The number of training and test sets for 4 butterfly species

Butterfly species	Group	Train set	Test set	Images for the test set
Adonis	w_1	21	3	I_1, I_2, I_3
Chestnut	w_2	22	2	I_4, I_5
Clouded sulphur	w_3	22	2	I_6, I_7
Cabbage	w_4	21	3	I_8, I_9, I_{10}

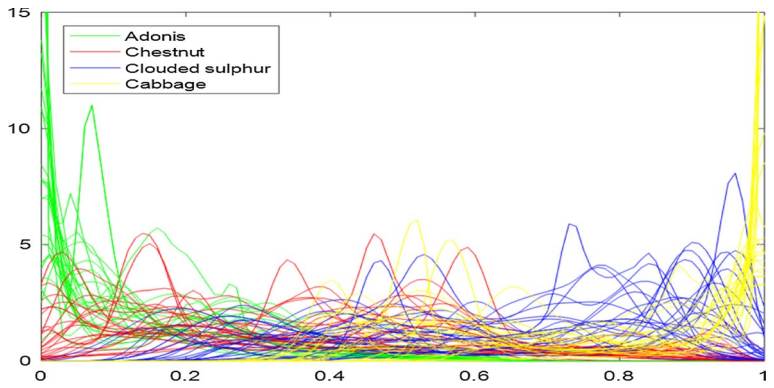


Fig. 4 The estimated PDFs of 96 butterfly images

- Step 1. Extracting the gray level of each image into a column. Since each image has the resolution level of 50176, it is extracted into the column vector with 50176 elements.
- Step 2. Estimating the PDFs from the data extracted in Step 1, we have the 96 PDFs shown in Fig. 4.
- Step 3. Suppose that we need to classify the image of Chestnut ($I_0 = I_5$ in w_2) in the test set (see Fig. 3d). First, we set up the initial partition matrix $U^{(0)}$ of size 4×93 based on the classification information of the butterfly groups from the train set:

$$U^{(0)} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & ... & 0.25 \\ 0 & 0 & 0 & 0 & 0 & 0 & ... & 0.25 \\ 0 & 0 & 1 & 1 & 0 & 1 & ... & 0.25 \\ 1 & 0 & 0 & 0 & 1 & 0 & ... & 0.25 \end{bmatrix}.$$

In the above matrix, we have: - From Column 1 to Column 86: $\mu_{ij}^{(0)} = 1,\ i = 1, 2, 3, 4; j = 1, 2, ..., 86$ if the *jth* image belongs to group w_i, and $\mu_{ij}^{(0)} = 0$ for otherwise. - The last column has four rows with the same value of 0.25.

- Perform Step 4, Step 5 and Step 6, after 17 iterations, the algorithm stops. At that time, the last column of matrix $U^{(17)}$ has the following result:

$$q = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} = \begin{bmatrix} \mu_{1(93)}^{(17)} \\ \mu_{2(93)}^{(17)} \\ \mu_{3(93)}^{(17)} \\ \mu_{4(93)}^{(17)} \end{bmatrix} = \begin{bmatrix} 0.2427 \\ 0.6984 \\ 0.0333 \\ 0.0256 \end{bmatrix}.$$

It means that the prior probability to put the image I_0 into four groups w_1, w_2, w_3, and w_4 are 0.2427, 0.6984, 0.0333 and 0.0256, respectively. Illustration for process to find the prior probability of I_0 is shown in Fig. 5.

- Step 7. Classify I_0 to the groups: We have

$$S(g_0, g_{v_1}) = 0.9042, S(g_0, g_{v_2}) = 0.8563,$$
$$S(g_0, g_{v_3}) = 0.8563, S(g_0, g_{v_4}) = 0.8524.$$

Therfore,

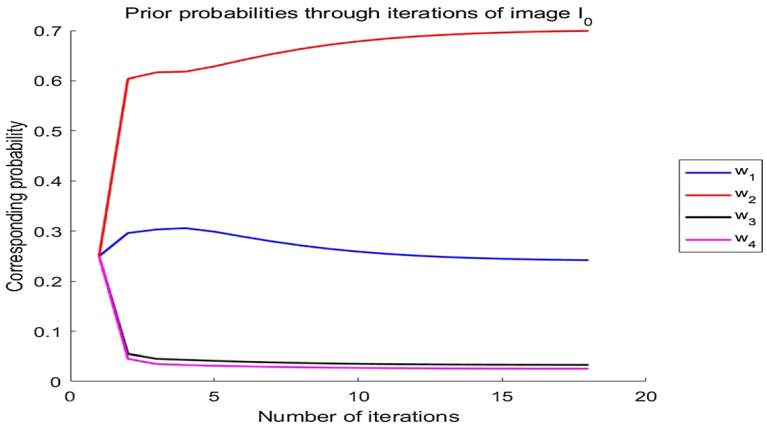


Fig. 5 The iterations for finding the prior probability of I_0

Table 2 The prior probability of 10 elements in the test set

Image	Adonis	Chestnut	Clouded surphul	Cabbage
I_1	0.9076	0.0666	0.0140	0.0118
I_2	0.9934	0.0049	0.0140	0.0118
I_3	0.9657	0.0217	0.0066	0.0060
I_4	0.1437	0.4658	0.2288	0.1617
I_5	0.2427	0.6987	0.0333	0.0256
I_6	0.1109	0.1663	0.4113	0.3114
I_7	0.0764	0.1592	0.5370	0.2275
I_8	0.0307	0.0450	0.0890	0.8353
I_9	0.0265	0.0455	0.0974	0.8307
I_{10}	0.0012	0.0031	0.0079	0.9878

Table 3 The SLC of PDFs extracted in test set and the representative PDFs for 4 species

Image	Adonis	Chestnut	Clouded surphul	Cabbage
I_1	0.8718	0.8562	0.8562	0.8524
I_2	0.8747	0.8559	0.8559	0.8520
I_3	0.8723	0.8523	0.8523	0.8483
I_4	0.8832	0.8563	0.8563	0.8524
I_5	0.8864	0.8554	0.8554	0.8515
I_6	0.8977	0.8538	0.8538	0.8499
I_7	0.9042	0.8563	0.8563	0.8524
I_8	0.9118	0.8549	0.8549	0.8510
I_9	0.9121	0.8563	0.8563	0.8000
I_{10}	0.9121	0.8549	0.8549	0.8508

$$q_1.S(g_0, g_{v_1}) = 0.2194, q_2.S(g_0, g_{v_2}) = 0.5980,$$
$$q_3.S(g_0, g_{v_3}) = 0.0285, q_4.S(g_0, g_{v_4}) = 0.0218.$$

Since $\max\{q_i.g_i(x)\} = q_4.S(g_0, g_{v_4})$, we put I_0 into w_2. Thus, this butterfly image has been classified into its correct species.

Doing the same for Step 3, Step 4, Step 5 and Step 6 of the remaining images in the test set, we get the prior probabilities of the ten images of the test set. This is presented in Table 2.

The SLC of PDFs extracted from 10 images in the test set and the representative PDFs for four groups is given in Table 3.

From the results of Tables 2, 3, and applying Step 7 of the proposed algorithm, we can easily classify the images in the test set. For example, we obtain as follows:

$*$ I_1, I_2, I_3 are classified w_1.
$*$ I_4, I_5 are classified w_2.
$*$ I_6, I_7 are classified w_3.
$*$ I_8, I_9, I_{10} are classified w_4.

It means that all ten butterfly images are correctly classified into their species.

We compare the proposed algorithm with popular methods such as Linear Discriminant Analysis (LDA), Support Vector Machines (SVM), Naive Bayes, QDA, XGBoost, kNN, Subspace-kNN, Subspace-Discriminant, Random Tree, AdaBoost, and Bagged.

Furthermore, the proposed algorithm is compared with the improved Bayesian method of Vovan et al. (2019): for this method, we consider many cases of the prior probability, such as the uniform distribution (Bayes-U), the sample ratio (Bayes-T), and Laplace method (Bayes-L). In addition, the methods of Huynh-Van et al. (2023) and Lethikim et al. (2022) are also compared. The number of correctly classified elements (CCE), the number of wrongly classified elements (WCE), and the rate of correct classification (RCC) of the algorithms are presented in Table 4.

Table 4 demonstrates that the proposed algorithm achieved 100% accuracy in classifying all elements of the test set, while the other methods had false classification rates ranging from 10% to 40%.

Table 4 The result of algorithms for the test set of butterfly data

Methods	CCE	WCE	RCC (%)
LDA	7	3	70
Logistic	6	4	60
SVM	8	2	80
Naive Bayes	6	4	60
QDA	7	3	70
XGBoost	7	3	70
kNN	9	1	90
Subspace–Discriminant	8	2	80
Random Tree	7	3	70
AdaBoost	8	2	80
Bagged	8	2	80
Bayes-U	6	4	60
Bayes-T	6	4	60
Bayes-L	6	4	60
Huynh-Van et al. (2023)	8	2	80
Lethikim et al. (2022)	7	3	70
Proposed	10	0	100

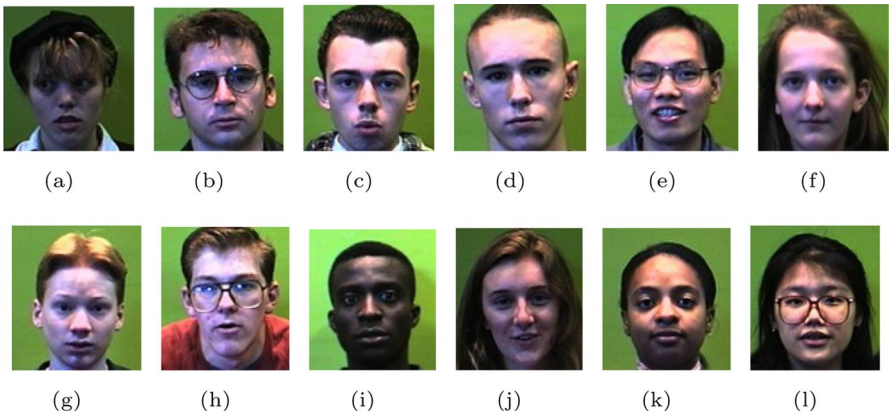


Fig. 6 12 human faces in considered 240 images

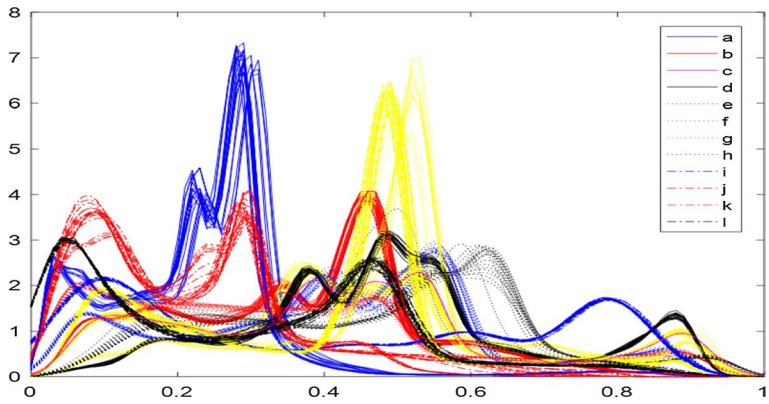


Fig. 7 The 240 PDFs extracted from 240 images of 12 human faces

4 Application

4.1 Face recognition

In this section, we analyze 240 images of human faces that are grouped into 12 categories, with 20 images per category. These images were sourced from the website https://cswww.essex.ac.uk/mv/allfaces/faces95.html. Figure 6 displays the 12 faces that are represented in the 240 images.

To evaluate the performance of our approach, we divided the set of 240 images into training and test sets with an 80% to 20% ratio. The training set consists of 192 images, while the test set comprises 48 images. Next, we extracted the features of all 240 images and estimated the representative PDFs for each image. The results are displayed in Fig. 7.

Figure 7 illustrates that the PDFs of the 12 groups exhibit significant overlap, posing a challenge for the efficiency of classification methods.

Performing Step 3 of the proposed method, we obtain the prior probability for 48 images of the test set. Next, the *SLC* of classified PDFs and representative PDFs for 12 groups are computed. Applying the proposed classification algorithm for 12 images in the test set, we get the correct classification for all. Comparing the proposed algorithm with other methods, we obtain Table 5.

Table 5 indicates that all methods achieved a high correct classification rate for this dataset (over 93%). However, the proposed method had the best performance, accurately classifying all images into their correct groups.

4.2 Identification of skin cancer

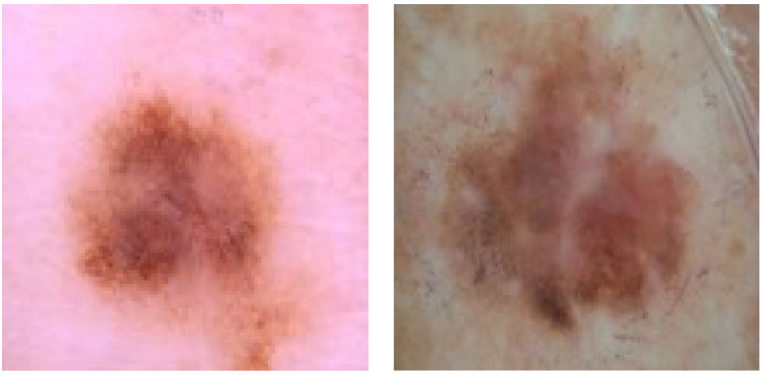
Cancer is currently a dangerous and frightening disease that is difficult to detect and treat. Skin cancer, in particular, is common and challenging to differentiate. It appears as moles or unusual rashes on the skin, making it difficult to distinguish between normal skin lesions and cancer with the naked eye. To address this problem, we used a dataset from https://www.kaggle.com/datasets. The dataset is divided into two groups: benign tumors and melanoma, with 160 and 90 images, respectively. Figure 8 shows some sample images from these two groups.

Extract the features for images, and estimate their PDFs, we obtain 250 PDFs shown in Fig. 9.

This study takes 200 images for training set and the remaining 50 images for the test set. The result of the proposed algorithm is compared with others in Table 6.

Table 5 The result of algorithms for the test set of human face data

Methods	CCE	RCC (%)
LDA	47	97.92
Logistic	45	93.75
SVM	47	97.92
QDA	46	95.83
XGBoost	45	93.75
kNN	47	97.92
Subspace -kNN	46	95.83
Subspace - Discriminant	45	93.75
Random Tree	45	93.75
AdaBoost	47	97.92
Bagged	45	93.75
Bayes-U	45	93.75
Bayes-T	45	93.75
Bayes-L	45	93.75
Huynh-Van et al. (2023)	47	97.92
Lethikim et al. (2022)	46	95.83
Proposed	48	100.0



(a) Non-Cancer (b) Cancer

Fig. 8 Sample of two image groups for skin cancer and non-cancer data

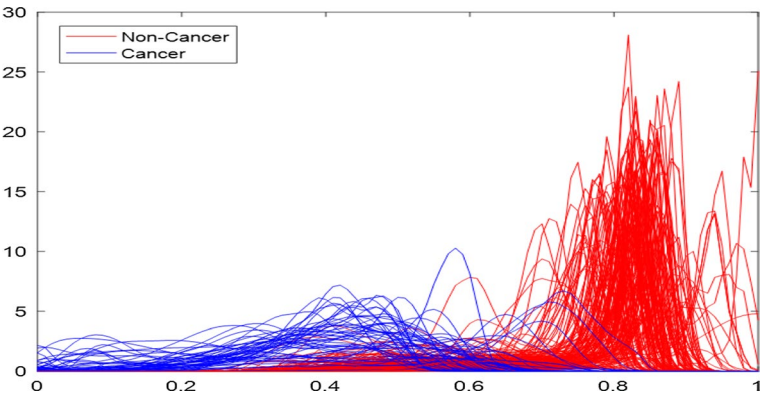


Fig. 9 The graph shows the estimated PDFs for 250 images

Table 6 indicates that the proposed algorithm outperforms other popular methods when the value of RCC is 100%. This demonstrates the superiority of our algorithm and its ability to achieve the best results.

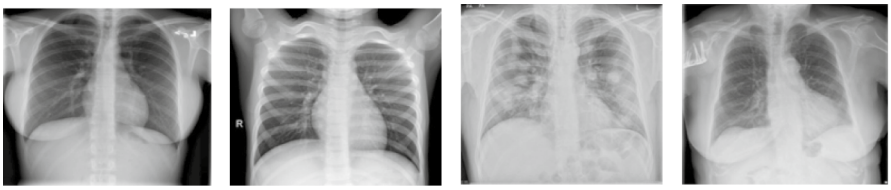
4.3 Recognizing the lung image for people infected with Covid-19

The goal of this study is to develop a method for recognizing lung images of people infected with Covid-19, which could potentially provide a faster and less expensive approach for detecting positive cases compared to current testing methods.

We obtained the dataset from https://www.kaggle.com/datasets, which includes 30,000 lung images, with 20,000 images infected with Covid-19 and 10,000 images not infected with Covid-19. For the training set, we randomly selected 29,000 images and reserved the remaining 1,000 images for the test set. Figure 10

Table 6 The result of algorithms for the test set of skin cancer and non-cancer data

Methods	NEC	RCC (%)
LDA	43	86
Logistic	43	86
SVM	44	88
QDA	44	88
XGBoost	45	90
kNN	48	96
Subspace -kNN	46	92
Subspace - Discriminant	45	90
Random Tree	46	92
AdaBoost	47	94
Bagged	47	94
Bayes-U	46	92
Bayes-T	48	94
Bayes-L	46	92
Huynh-Van et al. (2023)	48	94
Lethikim et al. (2022)	48	94
Proposed	50	100



(a) Not infected (b) Not infected (c) Infected (d) Infected

Fig. 10 Sample image of two groups infected and not infected with Covid-19

displays sample images from the two groups. Successful classification of these images could have significant implications for the diagnosis and treatment of Covid-19.

Extract the features and estimate the PDFs of 30000 images, we have Fig. 11.

Classifying for 1000 images of the test set, we obtain the result of methods in Table 7.

Figure 11 shows that the PDFs of the two groups have significant overlap, making it challenging to achieve accurate classification results using traditional methods. However, our proposed algorithm has achieved outstanding results, with a high correct classification rate of 92%.

Through three applications on datasets with different characteristics, the proposed classification algorithm consistently delivers good and stable results, surpassing many existing methods. In our opinion, the main reasons for these results are as follows:

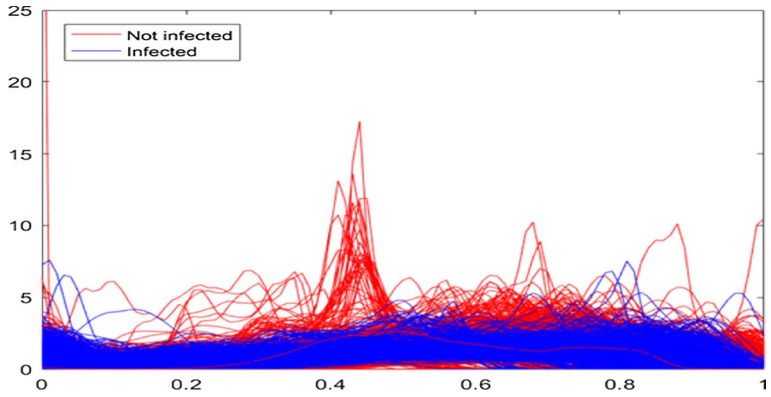


Fig. 11 The PDFs extracted from 2000 images

Table 7 The result of algorithms for the test set of infected and non-infectedr data with Covid-19

Methods	CCE	RCC (%)
LDA	861	86.1
Logistic	765	76.5
SVM	833	83.3
QDA	845	84.5
XGBoost	882	88.2
kNN	894	89.4
Subspace -kNN	861	86.1
Subspace - Discriminant	847	84.7
Random Tree	853	85.3
AdaBoost	827	82.7
Bagged	852	85.2
Bayes-U	778	77.8
Bayes-T	778	77.8
Bayes-L	779	77.9
Huynh-Van et al. (2023)	862	86.2
Lethikim et al. (2022)	879	87.9
Proposed	921	92.1

(1) The images are well distinguished with the proposed representative PDF, which is estimated from the data of extracted grayscale features.
(2) The prior probabilities are determined for each element instead of each group, unlike previous methods. Furthermore, relying on fuzzy relationships between the classified element and groups to find the prior probability is reasonable.
(3) The proposed classification rule is appropriate as it is based on the highest values of prior probability and similarity of the classified element with the groups using the established SLC measure.

5 Conclusion

To the best of our knowledge, the idea of classifying images based on extracted probability density functions (PDFs) is a novel approach that has not yet received much attention from statisticians and information technologists. This study proposes a new classification algorithm for image data based on representative PDFs, which improves upon the traditional Bayesian classifier.

Our proposed algorithm introduces a measure for evaluating the similarity between PDFs, which is used to build the algorithm and determine the prior probabilities. These probabilities are dependent on the training set and the relationships between the classified element and known groups. Additionally, the study addresses the problem of feature extraction from images into PDFs based on grayscale. An image is classified into a specific group if it has the highest prior probability and a similar level to that group. We demonstrate the proposed method step-by-step with a numerical example and effectively implement it using an established Matlab procedure. We apply the proposed method to four specific image sets that have significant applications in medicine. The examples and applications, with different characteristics in terms of number, domains, and features, demonstrate the reasonableness and stability of the proposed method, as well as its promising potential for real-world applications. In the future, we plan to apply the proposed approach to many other real-world problems.

Acknowledgements This research is funded by Vietnam National University Ho Chi Minh City (VNU-HCM) under Grant Number C2023-26-12.

Data availability The datasets analysed during this study are openly available from the public data in the website www.kaggle.com/datasets and www.essex.ac.uk/mv/allfaces/faces95.html.

Declarations

Conflict of interest No potential conflict of interest was reported by the authors.

References

Azimbagirad M, Fabrício HS, Antonio CSF, Junior LOM (2020) Tsallis-entropy segmentation through MRF and Alzheimer anatomic reference for brain magnetic resonance parcellation. Magn Resonance Imaging 65:136–145

Azimbagirad M, Junior LOM (2021) Tsallis generalized entropy for Gaussian mixture model parameter estimation on brain segmentation application. Neurosci Inf 1(1):100002

Behera DK, Das M, Swetanisha S (2022) Follower link prediction using the XGBoost classification model with multiple graph features. Wirel Pers Commun 127:695–714

Celebi E, Alpkocak A (2000) Clustering of texture features for content-based image retrieval. In: International conference on advances in information systems, Springer, pp. 216–225

Chen Y, Liu C, Chou K, Wang S (2016) Real-time and low-memory multi face detection system design based on Naive Bayes classifier using FPGA. In International automatic control conference (CACS), Berlin, pp. 7-12

Che-Ngoc H, Nguyen-Trang T, Nguyen-Bao T, Vo-Van T (2022) A new approach for face detection using the maximum function of probability density functions. Ann Oper Res 312:99–119

Dempster AP, Laird NM, Rubin DB (1977) Maximum likelihood from incomplete data via the EM algorithm. J R Stat Soc Ser B 39(1):1–22

Dietterich T (2000) An experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting, and randomization. Mach Learn 40(2):139–157
Fisher RA (1938) The statistical utilization of multiple measurements. Ann Eugen 8(4):376–386
Garg M, Dhiman G (2021) A novel content-based image retrieval approach for classification using GLCM features and texture fused LBP variants. Neural Comput Appl 33:1311–1328
Gou J, Du L, Zhang Y, Xiong T (2012) A new distance-weighted k-nearest neighbor classifier. J Inf Comput Sci 9(6):1429–1436
Huang S, Cai N, Pacheco PP, Narrandes S, Wang Y, Xu W (2018) Applications of support vector machine (SVM) learning in cancer genomics. Cancer Genom Proteom 15(1):41–51
Huynh-Van H, Le-Hoang T, Thai-Minh T, Nguyen-Dinh H, Vo-Van T (2023) Classifying the lung images for people infected with COVID-19 based on the extracted feature interval. Comput Methods Biomech Biomed Eng Imaging Vis 11(3):856–865
Imandoust SB, Bolandraftar M (2013) Application of k-nearest neighbor (KNN) approach for predicting economic events: theoretical background. Int J Eng Res Appl 3(5):605–610
Kung Y, Cheng-Chung W, Shih-Yun H, Shiu-Shia WL, Chung WY (2010) Application of logistic regression analysis of home mortgage loan prepayment and default risk. ICIC Express Lett 4(2):325–331
Laleh M, Shervan FE (2019) Texture image analysis and texture classification methods: a review. Int J Image Process Pattern Recogn 2(1):1–29
Lethikim N, Nguyentrang T, Vovan T (2022) A new image classification method using interval texture feature and improved Bayesian classifier. Multimed Tools Appl 81:36473–36488
Lethikim N, Lehoang T, Vovan T (2023) Automatic clustering algorithm for interval data based on overlap distance. Commun Stat Simul Comput 52(5):2194–2209
Maronna R, Charu C, Aggarwal K, Chandan KR (2016) Data clustering: algorithms and applications. Stat Pap 57:565–566
Miller G, Inkret WC, Little TT, Martz HF, Schillaci ME (2001) Bayesian prior probability distributions for internal dosimetry. Radiat Prot Dosimet 94(4):347–52
Neto JG, Ozorio LV, De Abreu TCC, Dos Santos BF, Pradelle F (2021) Modeling of biogas production from food, fruits and vegetables wastes using artificial neural network (ANN). Fuel 285:119081
Nguyentrang T, Vovan T (2017) A new approach for determining the prior probabilities in the classification problem by Bayesian method. Adv Data Anal Classif 11:629–643
Nhu VH, Zandi D, Shahabi H, Chapi K, Shirzadi A, Al-Ansari N, Singh SK, Dou J, Nguyen H (2020) Comparison of support vector machine, Bayesian logistic regression, and alternating decision tree algorithms for shallow landslide susceptibility mapping along a mountainous road in the west of Iran. Appl Sci 10(15):5047
Pham-Gia T, Turkkan N, Tai VV (2000) Statistical discrimination analysis using the maximum function. Commun Stat Simul Comput 37(2):320–336
Phamtoan D, Vovan T (2021) Automatic fuzzy genetic algorithm in clustering for images based on the extracted intervals. Multimed Tools Appl 80:35193–35215
Phamtoan D, Nguyenhuu K, Vovan T (2022) Fuzzy clustering algorithm for outlier interval data based on the robust exponent distance. Appl Intell 52:6276–6291
Phamtoan D, Vovan T (2023) The fuzzy cluster analysis for interval value using genetic algorithm and its application in image recognition. Comput Stat 38:25–51
Park SB, Lee JW, Kim SK (2004) Content-based image classification using a neural network. Pattern Recogn Lett 25(3):287–300
Renukadevi T, Saraswathi K, Prabu P, Venkatachalam K (2022) Brain image classification using time frequency extraction with histogram intensity similarity. Comput Syst Sci Eng 41(2):460–645
Scott DW (2015) Multivariate density estimation: theory, practice, and visualization. Wiley, London
Shawe TJ, Cristianini N (2000) An introduction to support vector machines and other kernel-based learning methods. Cambridge University Press, Cambridge
Terrell GR, Scott DW (1992) Variable kernel density estimation. Ann Stat 20(3):1236–1265
VijayaLakshmi B, Mohan V (2016) Kernel-based PSO and FRVM: An automatic plant leaf type detection using texture, shape, and color features. Comput Electron Agric 125:99–112
Vovan T, Pham-Gia T (2010) Clustering probability distributions. J Appl Stat 37(11):1891–1910
Vovan T (2016) L^1-distance and classification problem by Bayesian method. J Appl Stat 44(3):385–401
Vovan T, Chengoc H, Nguyentrang T (2017) Textural features selection for image classification by Bayesian method. In: 2017 13th international conference on natural computation, fuzzy systems and knowledge discovery (ICNC-FSKD), pp. 733-139

Vovan T (2018) Some results of classification problem by Bayesian method and application in credit operation. Stat Theory Relat Fields 2(2):150–157
Vovan T, Tranphuoc L, Chengoc H (2019) Classifying two populations by Bayesian method and applications. Commun Math Stat 7(2):141–161
Vovan T, Lethikim N, Nguyentrang T (2021) An efficient robust automatic clustering algorithm for interval data. Commun Stat Simul Comput. https://doi.org/10.1080/03610918.2021.1965165
Vovan T, Chengoc H, Ledai N, Nguyentrang T (2022) A new strategy for short-term stock investment using Bayesian approach. Comput Econ 59:887–911
Wyner AJ, Olson M, Bleich J, Mease D (2017) Explaining the success of AdaBoost and random forests as interpolating classifiers. J Mach Learn Res 18(48):1–33
Yuan W, Xiaoqian J, Jihoon K, Lucila OM (2012) Grid binary logistic regression glore: building shared models without sharing data. J Am Med Inf Assoc 19(5):758–764

Authors and Affiliations

Hieu Huynh-Van[1,2,3] · Tuan Le-Hoang[2,4] · Tai Vo-Van[5]

✉ Tai Vo-Van
vvtai@ctu.edu.vn

Hieu Huynh-Van
hvhieu.sdh221@hcmut.edu.vn

Tuan Le-Hoang
tuanlh@uit.edu.vn

[1] Faculty of Applied Science, Ho Chi Minh City University of Technology (HCMUT), Ho Chi Minh City, Vietnam

[2] Vietnam National University Ho Chi Minh City, Ho Chi Minh City, Vietnam

[3] Faculty of Fundamental Science, Industrial University of Ho Chi Minh City, Ho Chi Minh City, Vietnam

[4] Department of Mathematics and Physics, University of Information Technology, Ho Chi Minh City, Vietnam

[5] College of Natural Science, Can Tho University, Can Tho City, Vietnam