

Article

Image Feature Extraction Using Symbolic Data of Cumulative Distribution Functions

Sri Winarni ^{1,*}, Sapto Wahyu Indratno ², Restu Arisanti ¹ and Resa Septiani Pontoh ¹¹ Department of Statistics, Universitas Padjadjaran, Jl. Raya Bandung Sumedang km 21 Jatinangor, Sumedang 45363, Indonesia; r.arisanti@unpad.ac.id (R.A.); resa.septiani@unpad.ac.id (R.S.P.)² Statistics Research Group, Faculty of Mathematics and Natural Sciences, Institut Teknologi Bandung, Jl. Ganesha No. 10, Bandung 40132, Indonesia; saptowi@itb.ac.id

* Correspondence: sri.winarni@unpad.ac.id

Abstract: Symbolic data analysis is an emerging field in statistics with great potential to become a standard inferential technique. This research introduces a new approach to image feature extraction using the empirical cumulative distribution function (ECDF) and distribution function of distribution values (DFDV) as symbolic data. The main objective is to reduce the dimension of huge pixel data by organizing them into more coherent pixel-intensity distributions. We propose a partitioning method with different breakpoints to capture pixel intensity variations effectively. This results in an ECDF representing the proportion of pixel intensities and a DFDV representing the probability distribution at specific points. The novelty of this approach lies in using ECDF and DFDV as symbolic features, thus summarizing the data and providing a more informative representation of the pixel value distribution, facilitating image classification analysis based on intensity distribution. The experimental results underscore the potential of this method in distinguishing image characteristics among existing image classes. Image features extracted using this approach promise image classification analysis with more informative image representations. In addition, theoretical insights into the properties of DFDV distribution functions are gained.

Keywords: image features; symbolic data; empirical cumulative distribution function; distribution of distribution values; image classification

MSC: 62P99

Citation: Winarni, S.; Indratno, S.W.; Arisanti, R.; Pontoh, R.S. Image Feature Extraction Using Symbolic Data of Cumulative Distribution Functions. *Mathematics* **2024**, *12*, 2089. <https://doi.org/10.3390/math12132089>

Academic Editor: Andrea Scozzari

Received: 14 June 2024

Revised: 27 June 2024

Accepted: 28 June 2024

Published: 3 July 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In contemporary image processing, extracting image features holds significant importance and is required in a wide range of applications [1], including pattern recognition [2], object detection [3–6], and other fields. In an era of abundant visual data, image features are crucial for presenting relevant and structured information from complex images [7–12]. These features are numerical representations that describe various essential aspects of an image, ranging from pixel values to texture and shape. While image feature extraction is necessary for image analysis, using pixel values as the main feature can be challenging. One of the main issues is the high dimensionality of the features, as well as their excessive sensitivity to small changes in the image. In image classification, using pixel values as random variables often results in an extensive representation of random variables, leading to computational complexity and high memory requirements [13]. Additionally, pixel values are sensitive to changes in scale, rotation, and illumination, where minor changes in the image can result in classification errors [14]. Pixel values are also sensitive to lighting variations, making it difficult for classification models to distinguish the same object under different lighting conditions [15]. To overcome these problems, an alternative approach is needed that can provide a more effective solution.

This research proposes using symbolic data as no longer a single value as classical data. Symbolic data are data that are represented in the forms of symbols, denoting categories, intervals, distributions, and the like. Symbolic data summarizes or describes an extensive database to make it easier to analyze and still contains relevant information from the original database [16,17]. The symbolic data used in this study include the empirical cumulative distribution function (ECDF), which provides an overview of the relative distribution of pixel values in an image. The concept of ECDF has not been widely used as a feature in image classification research. In another field, ECDF is used to classify regions based on global atmospheric temperature and humidity profiles [18,19]. The subsequent symbolic data presented in this study are the distribution function of distribution values (DFDV), referring to the distribution of ECDF values at specific points. The subsequent section will elaborate on this concept. The concept of distribution functions as image features is new in image classification.

Using ECDF for symbolic data as image features offers several advantages: first, more efficient processing. Symbolic data in the form of ECDF provide information related to the distribution of pixel values and the patterns formed between pixel values. This provides more detailed information about the characteristics of the image because, in addition to the pixel value information, there is also information about the distribution of pixel values in the image. Second, this approach benefits extensive image data by reducing dimensionality and saving memory. By summarizing the pixel value distribution information into a few change values, ECDF enables data dimensionality reduction without sacrificing important information. This helps overcome the considerable memory challenges often associated with extensive image data. Third, using ECDF provides better tolerance to exposure, rotation, and scale variations in the image. If these aspects change, the distribution of pixel values will remain relatively constant. This allows the model to recognize objects or patterns better despite variations in the image. Therefore, using distribution functions as image features can be an attractive alternative. Further exploration of empirical cumulative distribution function-based approaches has the potential to improve image classification quality.

This research aims to investigate a new method for image feature extraction using symbolic data by depicting pixel value distributions through the ECDF. This initial study aims to use the extracted features for classification analysis. This research contributes to image feature extraction by introducing the use of ECDF and DFDV as symbolic data. The ECDF and DFDV methods convert high-dimensional pixel data into structured formats, which summarize pixel intensity distributions across an image. This approach seeks to improve the robustness and effectiveness of feature representation, enhancing image classification accuracy and efficiency. This research enhances the symbolic data analysis approach by using ECDF and DFDV to represent pixel intensity data. This breakthrough opens up opportunities for applying these techniques to a range of image processing and computer vision tasks. Preliminary applications in image classification [20,21] have shown promise.

The structure of this paper consists of several primary segments. The Section 1 is the introduction, which furnishes context regarding the significance of image feature extraction in contemporary image processing. Next is the Section 2, which explains in detail the use of ECDF in image feature extraction and the methods used in this study. The Section 3 presents the experimental results, where the data and findings from this research are presented and analyzed. The discussion follows, where the findings are further discussed and their implications explored. For the conclusion, the paper summarizes the fundamental discoveries and outlines potential avenues for future research.

2. Cumulative Distribution Function for Image Feature Extraction

This research presents a method for extracting image features by using a distribution function to uncover significant patterns and characteristics of the image's pixel intensity distribution. This section examines the application of ECDF for feature extraction, starting

with the depiction of pixel intensity values and progressing to more sophisticated methods such as symbolic data and DFDV. We utilize the Kolmogorov–Smirnov test to validate the appropriateness of the employed distribution model, thereby enhancing the analysis in the field of digital image processing.

2.1. Image Features Represented by Pixel Intensity Values

An image consists of several pixels represented in a matrix with a size equal to the size of the image. The size depends on the number of rows and columns of pixels in the image. If the row size equals the column size, the pixel value matrix representation will be an $m \times m$ square, where m is the number of rows and columns. The pixel value matrix consists of value elements that reflect the brightness of colors. In color images, these values are represented by red, green, and blue, each with a brightness level of $0, 1, \dots, 255$. In gray images, the color brightness level reflects the gray level with a value of $0, 1, \dots, 255$. A value of 0 represents black, and 255 represents white. Values between 0 and 255 produce various levels of gray; the higher the pixel value, the lighter the pixel's gray [22].

The representation of an $m \times m$ image in a matrix of pixel values can be written as matrix $A_{m \times m}$. The matrix element is denoted by $x_{b,c}$, which is the image pixel value in the row- b of the column- c , with $b = 1, 2, \dots, m$ and $c = 1, 2, \dots, m$. In image processing analysis, to facilitate data representation and analysis, the matrix of pixel values needs to be converted into a vector form, referred to as vectorization. Two approaches can be used: the row-by-row approach and the column-by-column approach. In the row-by-row approach, each row of the pixel value matrix is taken sequentially and then combined into a vector [23]. One pixel value matrix will become one data vector. If row-by-row vectorization is performed on the matrix $A_{m \times m}$, the vector $vec(A)$ will be formed, which is illustrated in Figure 1.

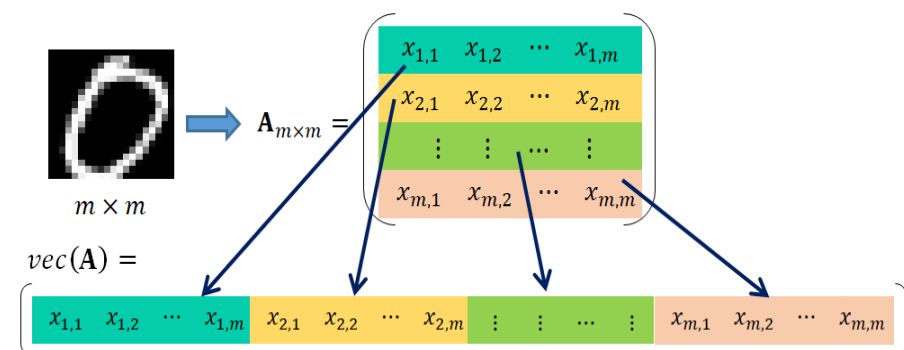


Figure 1. Illustration of vector formation $vec(A)$ from pixel value matrix $A_{m \times m}$ using row-by-row vectorization.

The result of vectorization is the vector $vec(A)$ with the vector elements represented as x_i , which is the value of the pixel i , where $i = 1, 2, \dots, n$ and $n = m \times m$ is the number of pixels in the image. If there are as many as N images, then matrices $A_1, A_2, \dots, A_N$ will represent image j , with $j = 1, 2, \dots, N$. The vector representation for each image is $vec_n(A_j)$, with $n = m \times m$. Thus, a data matrix X of size $N \times n$ is formed with matrix elements denoted by x_{ji} , which is the value of the pixel i in the image j , with $i = 1, 2, \dots, n$ and $j = 1, 2, \dots, N$. The $X_{N \times n}$ matrix and its elements can be written as follows:

$$X_{N \times n} = \begin{bmatrix} vec_n(A_1) \\ vec_n(A_2) \\ vec_n(A_3) \\ \vdots \\ vec_n(A_N) \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & x_{13} & x_{14} & x_{15} & \cdots & x_{1n} \\ x_{21} & x_{22} & x_{23} & x_{24} & x_{25} & \cdots & x_{2n} \\ x_{31} & x_{32} & x_{33} & x_{34} & x_{35} & \cdots & x_{3n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{N1} & x_{N2} & x_{N3} & x_{N4} & x_{N5} & \cdots & x_{Nn} \end{bmatrix} \quad (1)$$

Element x_{11} is the first-pixel value of the first image; element x_{12} is the second-pixel value of the first image, and so on [24]. This $X_{N \times n}$ data structure is a classic data structure that represents pixel values as single values with a considerable size.

2.2. The Empirical Cumulative Distribution Function (ECDF)

Suppose X_j is a random variable of pixel values in the image j , with $j = 1, 2, \dots, N$. The ECDF of the pixel value in image j is denoted by $\mathcal{F}_j(x)$, where x is the realization of the pixel value, which is $0, 1, 2, \dots, 255$, defined as follows:

$$\mathcal{F}_j(x) = \frac{1}{n} \sum_{i=1}^n 1\{x_{ji} \leq x\} \quad (2)$$

where n is the number of pixels, and $1\{x_{ji} \leq x\}$ is an indicator function that takes value 1 if $x_{ji} \leq x$ and 0 otherwise; it can be written as follows:

$$1\{x_{ji} \leq x\} = \begin{cases} 1 & \text{if } x_{ji} \leq x \\ 0 & \text{else} \end{cases} \quad (3)$$

The function $\mathcal{F}_j(x)$ is the proportion of the total observations with values less than or equal to x [25].

2.3. Symbolic Data

In this research, the classical data structure in Equation (1) is converted into symbolic data in the form of ECDF $\mathcal{F}_j(x)$ with $j = 1, 2, \dots, N$ using Equation (2). The symbolic data structure for the ECDF of pixel value S_j is defined as follows:

$$X_{N \times n} = \begin{bmatrix} x_{11} & x_{12} & x_{13} & x_{14} & x_{15} & \cdots & x_{1n} \\ x_{21} & x_{22} & x_{23} & x_{24} & x_{25} & \cdots & x_{2n} \\ x_{31} & x_{32} & x_{33} & x_{34} & x_{35} & \cdots & x_{3n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{N1} & x_{N2} & x_{N3} & x_{N4} & x_{N5} & \cdots & x_{Nn} \end{bmatrix} \rightarrow S_j = \begin{bmatrix} \mathcal{F}_1(x) \\ \mathcal{F}_2(x) \\ \mathcal{F}_3(x) \\ \vdots \\ \mathcal{F}_N(x) \end{bmatrix} \quad (4)$$

The matrix element $X_{N \times n}$ can be summarized into symbolic data S_j whose element $\mathcal{F}_j(x)$ is the ECDF of the image j pixel value, with $j = 1, 2, \dots, N$ given in Equations (2) and (3) [17]. The formation of the ECDF of the pixel values of an image can be done by using the elements in the row of the matrix $X_{N \times n}$ or the elements of the symbolic data S_j in Equation (4). The formation of the ECDF of the pixel values is illustrated in Figure 2.

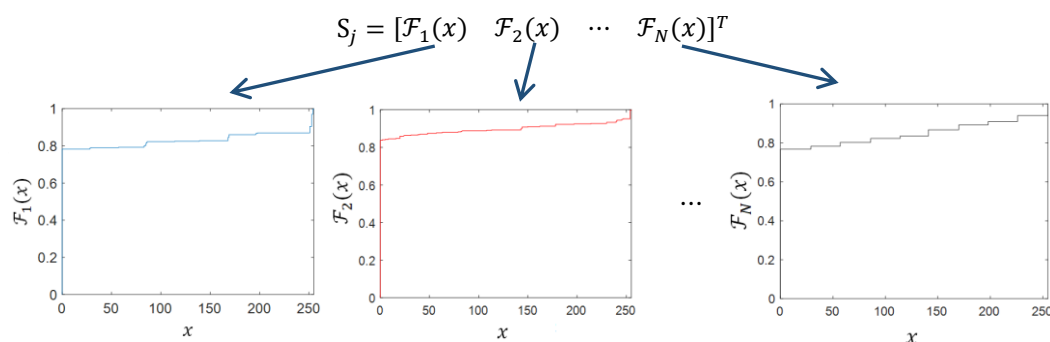


Figure 2. Illustration of ECDF formation for each image. The element S_j can be written as $\mathcal{F}_j(x)$ with $j = 1, 2, \dots, N$ and $x = 0, 1, 2, \dots, 255$ are the realized pixel values.

The ECDF function formed in Figure 2 represents the probability that the pixel value in an image is less than or equal to a certain value. For example, $\mathcal{F}_1(83) = 0.8$ means that in image 1, the probability that the pixel intensity value is less than or equal to 83 is 0.8. In

other words, 80% of all pixels in the image have values less than or equal to 83. This is a characteristic of imagery using pixel values that are less than or equal to x .

Datasets often used in image classification analysis consist of multiple image classes. If the image classes are denoted in $k = 0, 1, 2, \dots, v - 1$, where v is the number of image classes, and the number of images in class- k is denoted by N_k , then the symbolic data S_j in Equation (4) can be written as follows:

$$K_k = \begin{bmatrix} \mathcal{K}_{0j} \\ \mathcal{K}_{1j} \\ \mathcal{K}_{2j} \\ \vdots \\ \mathcal{K}_{(v-1)j} \end{bmatrix}, \text{ with } \mathcal{K}_{kj} = \begin{bmatrix} \mathcal{F}_{k1}(x) \\ \mathcal{F}_{k2}(x) \\ \mathcal{F}_{k3}(x) \\ \vdots \\ \mathcal{F}_{kN_k}(x) \end{bmatrix} \quad (5)$$

with $k = 0, 1, 2, \dots, v - 1$; $\sum_{k=0}^{v-1} N_k = N$. The symbolic data K_k with element $\mathcal{K}_{kj}$ is the ECDF of the class image k . The number of images of all N_k classes equals the number of images in the dataset. These forms of symbolic data are much more compact than classical data, as they are no longer presented as single values but as symbolic distribution functions for each image class [17].

The element $\mathcal{F}_{kj}(x)$ of symbolic data $\mathcal{K}_{kj}$ is the ECDF value of the image class k with realized pixel values $x \in \{0, 1, 2, \dots, 255\}$. If it is assumed that all 256 pixel values appear, then the element $\mathcal{F}_{kj}(x)$ can be written as follows:

$$\mathcal{F}_{kj}(x) = [\mathcal{F}_{kj}(0) \quad \mathcal{F}_{kj}(1) \quad \mathcal{F}_{kj}(2) \quad \mathcal{F}_{kj}(3) \quad \dots \quad \mathcal{F}_{kj}(255)] \quad (6)$$

This shows that classical data $X_{N \times n}$ can be summarized by symbolic data S_j with a maximum of 256. The sizes of these data will be smaller than the original data, especially if not all realized pixel values appear.

2.4. Distribution Function of Distribution Values (DFDV)

In this research, an image will generate an ECDF. For example, if there are N images, the set of ECDFs for those images will be generated. At a certain point, say T_1 and T_2 , the ECDF values of each function can be collected, and a distribution function of distribution values (DFDV) is formed, representing the distribution functions of pixel ECDF values that describe the usage characteristics of pixel values at points T_1 and T_2 . The usage characteristics of these pixel values can be different in different image classes. The selection of different points will also give different characteristics. Furthermore, points T_1 and T_2 are distinguishing points. The DFDV formed will be as many as the distinguishing points taken [26].

Two symbolic datasets were used in this research. The first is the symbolic dataset for the ECDF of the image class k , defined as follows:

$$\mathcal{K}_{kj} = [\mathcal{F}_{k1}(x) \quad \mathcal{F}_{k2}(x) \quad \mathcal{F}_{k3}(x) \quad \dots \quad \mathcal{F}_{kN_k}(x)]^T \quad (7)$$

where $\mathcal{K}_{kj}$ denotes the symbolic data for the class image k , with $k = 0, 1, 2, \dots, v - 1$, and N_k is the number of images in the class k . The element $\mathcal{F}_{k1}(x)$ is the ECDF of class k of the first image. Then, $\mathcal{F}_{k2}(x)$ denotes the ECDF of class k of the second image, and so on, until $\mathcal{F}_{kN_k}(x)$ is the ECDF of class k of the image N_k . The element $\mathcal{F}_{kj}(x)$ can be written following Equation (6).

The symbolic data, $\mathcal{K}_{kj}$, in Equation (5) contain the ECDF value $\mathcal{F}_{kj}(x)$. This symbolic data summarize the classic data of pixel values into symbolic data of distribution functions with much smaller data sizes. The second symbolic dataset denotes DFDV, which summarize the symbolic data $\mathcal{K}_{kj}$ into G_{kt} , denoting the symbolic data of the DFDV at image class k of the discrimination point, with $k = 0, 1, 2, \dots, v - 1$ and $t = 1, 2$. The symbolic data, G_{kt} , are defined as follows:

$$G_{kt} = [G_{kT_1}(y) \quad G_{kT_2}(y)]^T \quad (8)$$

where $G_{kT_1}(y)$ is the DFDV of image class k of discrimination point T_1 , and $G_{kT_2}(y)$ is the DFDV of image class k of discrimination point T_2 . The DFDV is the CDF of the ECDF values, where y is the $\mathcal{F}_{kj}(T)$ value, so $y \in [0, 1]$. G_{kt} symbolic data are much smaller than classical data.

The DFDV is the probability that a distribution will have a value smaller than or equal to a certain value at a certain point in the distribution. Suppose X_j is a random variable of pixel values in the image j with $j = 1, 2, \dots, N$, and N are many images. Random variable X_1 is the pixel value in image 1, random variable X_2 is the pixel value in image 2, and so on. The ECDF function of N images can be written as $\mathcal{F}_1(x), \mathcal{F}_2(x), \dots, \mathcal{F}_N(x)$. Let $\mathfrak{F} = \{\mathcal{F}_1(x), \mathcal{F}_2(x), \dots, \mathcal{F}_N(x)\}$ be the set of ECDF $\mathcal{F}_j(x)$ with $j = 1, 2, \dots, N$.

Consider a particular point, say $T \in \{0, 1, 2, \dots, 255\}$, and a probability value of $y \in [0, 1]$. Then, the DFDV at point T can be interpreted as the probability that $\mathfrak{F}$ will have a cumulative distribution value less than or equal to y at that point T . The DFDV function at point T , denoted by $G_T(y)$, is a function from $[0, 1]$ to $[0, 1]$ defined as follows:

$$G_T(y) = P(\{\mathcal{F}_j(x) \in \mathfrak{F} | \mathcal{F}_j(T) \leq y\}) \quad (9)$$

with $\mathcal{F}_j(T)$ representing the ECDF value of $\mathcal{F}_j(x)$ at point T [18].

A distinction point $T \in \{0, 1, 2, \dots, 255\}$ is a particular pixel value used to measure the ECDF value at that point. Furthermore, a change value of $y \in [0, 1]$ is required, which is the realization of the ECDF value $\mathcal{F}_j(x)$ at point T , denoted $\mathcal{F}_j(T)$. The DFDV function denoted by $G_T(y)$ calculates the probability that $\mathcal{F}_j(T)$ is less than or equal to y .

2.5. Goodness of Fit: Kolmogorov–Smirnov Test

The Kolmogorov–Smirnov test evaluates how well the empirical distribution of a sample of data ($F_N(y)$) matches a specified theoretical distribution ($F(y)$). The hypothesis is formulated as follows:

H₀. $F_N(y) = F(y)$ (Data follows a particular distribution).

H₁. There is at least one value of y for which $F_N(y) \neq F(y)$ (data do not follow a particular distribution).

The test statistics used are as follows:

$$\mathcal{D} = \max |F_N(y) - F(y)| \quad (10)$$

In this equation, $\mathcal{D}$ is the KS statistic, quantifying the maximum difference between the ECDF of the sample, denoted as $F_N(y)$, and the cumulative distribution function (CDF) of the reference distribution, denoted as $F(y)$. The critical value for the test statistic $\mathcal{D}$, denoted by D_α , where α is the actual level or limit used to determine whether H_0 is rejected. If $\mathcal{D} \leq D_\alpha$, H_0 is accepted, the data follows a particular distribution. Conversely, if $\mathcal{D} > D_\alpha$, H_0 is rejected, meaning the data do not follow a particular distribution. Testing can also be done using the p_{value} . If $p_{value} < \alpha$, then H_0 is rejected; if $p_{value} \geq \alpha$, then H_0 is not rejected [27]. Parametric distributions are not always appropriate for describing data distributions. Non-parametric approaches, such as kernel density estimation, can be an alternative to describe data that does not follow a particular distribution pattern.

3. Applications and Theoretical Results

This section explores practical applications and theoretical insights into ECDF usage in specific contexts. It begins with essential data preprocessing steps to prepare research data. The section then discusses outcomes from using ECDF to describe empirical data distributions, alongside theoretical discoveries and the application of DFDV to identify

significant differences among tested distributions. Validation of distribution models using the Kolmogorov–Smirnov goodness of fit test is also covered. This section provides an exploration of CDF application in data analysis, emphasizing practical uses and theoretical insights.

3.1. Research Data and Pre-Processing

This research uses the Modified National Institute of Standards and Technology (MNIST) dataset, which is widely used in experiments and the development of machine learning models. The MNIST dataset contains a set of handwritten number images from 0 to 9, gray in color with a black background. Each image has a resolution of 28×28 pixels (784 pixels in total) and is labeled according to the number contained in the image. An illustration of the MNIST data is given in Figure 3.

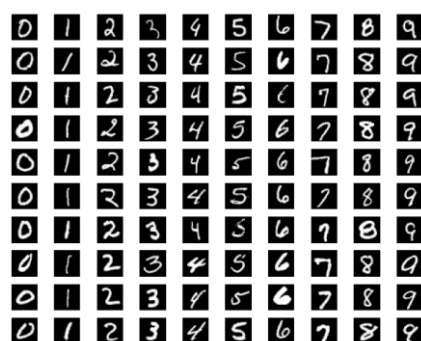


Figure 3. Illustration of the MNIST dataset for handwritten digit recognition.

The initial stage in performing image feature extraction is pre-processing. Pre-processing aims to prepare data and improve image quality for better analysis results. One technique that is often used in pre-processing is cropping. Cropping involves taking a portion of the original image to obtain an essential area of the image object. The most common cropping is taking the center of the image [28]. Image cropping reduces the image's background and makes the image more focused on the object to be analyzed. The goal is to reduce visual complexity and focus attention on the main object. Reducing visual complexity can improve data quality, allow algorithms to work more efficiently, reduce computational load, and speed up analysis time.

In addition to pre-processing in the form of image cropping, this research also involves image partitioning. This research uses the empirical cumulative distribution function (ECDF) pixel value feature, a method to describe the distribution of pixel value usage in an image. If the number of pixels with the same value is high, the distribution of pixel values is likely to be uniform. Therefore, a partitioning technique is used to capture object pixels in more detail and distinguish the distribution between image classes. Partitioning aims to capture a more detailed representation of an object or image pattern. Partitioning allows for a better distinction between the characteristics of image pixel intensity values.

The choice of partition shape depends on the properties and characteristics of the image objects. Each partition shape provides a different analytical perspective and can significantly impact the final image processing outcome. This study uses distribution-based features, so the chosen partition should effectively capture image value characteristics. In the MNIST dataset, which consists of handwritten digits 0–9, for example, the digit '1' has a distinctive single vertical line, and column or diagonal partitions might contain no objects since the single line may fall into one specific column or diagonal, leaving others empty. This affects the stability of the resulting pixel value distribution function. However, row partitions are likely to contain objects in each partition. An illustration of cutting and partitioning the image into two rows is provided in Figure 4.

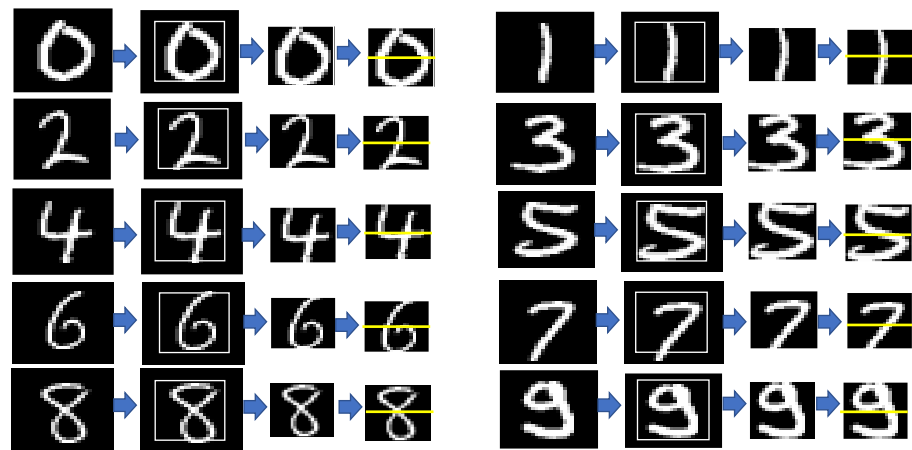


Figure 4. Illustration of the center image cropping. The result is a 20×20 image with a more focused object.

Image partitioning is performed on the data matrix $X_{N \times n}$ in Equation (4), where N is the total number of images and n is the number of pixels. A two-row partition can be made like in the following matrix:

$$X_{N \times n} = \begin{bmatrix} x_{11} & x_{12} & x_{13} & \cdots & \cdots & x_{1\frac{n}{2}} & x_{1(\frac{n}{2}+1)} & x_{1(\frac{n}{2}+2)} & x_{1(\frac{n}{2}+3)} & \cdots & \cdots & x_{1n} \\ x_{21} & x_{22} & x_{23} & \cdots & \cdots & x_{2\frac{n}{2}} & x_{2(\frac{n}{2}+1)} & x_{2(\frac{n}{2}+2)} & x_{2(\frac{n}{2}+3)} & \cdots & \cdots & x_{2n} \\ x_{31} & x_{32} & x_{33} & \cdots & \cdots & x_{3\frac{n}{2}} & x_{3(\frac{n}{2}+1)} & x_{3(\frac{n}{2}+2)} & x_{3(\frac{n}{2}+3)} & \cdots & \cdots & x_{3n} \\ x_{41} & x_{42} & x_{43} & \cdots & \cdots & x_{4\frac{n}{2}} & x_{4(\frac{n}{2}+1)} & x_{4(\frac{n}{2}+2)} & x_{4(\frac{n}{2}+3)} & \cdots & \cdots & x_{4n} \\ x_{51} & x_{52} & x_{53} & \cdots & \cdots & x_{5\frac{n}{2}} & x_{5(\frac{n}{2}+1)} & x_{5(\frac{n}{2}+2)} & x_{5(\frac{n}{2}+3)} & \cdots & \cdots & x_{5n} \\ x_{61} & x_{62} & x_{63} & \cdots & \cdots & x_{6\frac{n}{2}} & x_{6(\frac{n}{2}+1)} & x_{6(\frac{n}{2}+2)} & x_{6(\frac{n}{2}+3)} & \cdots & \cdots & x_{6n} \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ x_{N1} & x_{N2} & x_{N3} & \cdots & \cdots & x_{N\frac{n}{2}} & x_{N(\frac{n}{2}+1)} & x_{N(\frac{n}{2}+2)} & x_{N(\frac{n}{2}+3)} & \cdots & \cdots & x_{Nn} \end{bmatrix} \quad (11)$$

The sub-matrix indicates the element of the partition. In two-row partitioning, the partitioning is done at the pixel $(\frac{n}{2})$. In the image class k , the number of images is N_k . In this case, n is the number of pixels of a 20×20 image, therefore n is 400.

The data matrix elements need to be transformed into the interval $[0, 1]$ to avoid the risk of the mathematical calculation results being too large or too small. This transformation will make the calculation result more stable. In addition, machine learning algorithms can work better when the input values are in a smaller and uniform range. There are several transformation methods; this research uses the min–max transformation.

Suppose X is the original image with features represented by a matrix, and x_{bc} is the matrix element in the row b and the column c . The transformation equation is given as follows:

$$\tilde{x}_{bc} = \frac{x_{bc} - \min\{x_{bc}\}}{(\max\{x_{bc}\} - \min\{x_{bc}\})}; \quad b = 1, 2, \dots, m \text{ and } c = 1, 2, \dots, m \quad (12)$$

where $\tilde{x}_{bc}$ is the transformed feature value in the b th row and c th column, and x_{bc} is the feature value in row b and column c of the original image. This transformation will convert image feature values that range between $\min\{x_{bc}\}$ and $\max\{x_{bc}\}$ into values that range from $[0, 1]$ [29].

3.2. Results of ECDF Application

There are ten image classes $k = 0, 1, 2, \dots, 9$, each with $j = 1, 2, \dots, N_k$ images with $l = 1, 2$ partitions. Each partition has a distinguishing point $t = 1, 2$. Partitioning is done

in two rows with two distinguishing points at points $T_1 = 0.03$ and $T_2 = 0.6$. The ECDF pixel values are formed in Equation (2) using pixel values transformed using Equation (12). One image in one class and one partition produces ECDF, as illustrated in Figure 2, so a set of images will form a set of ECDF given in Figure 5.

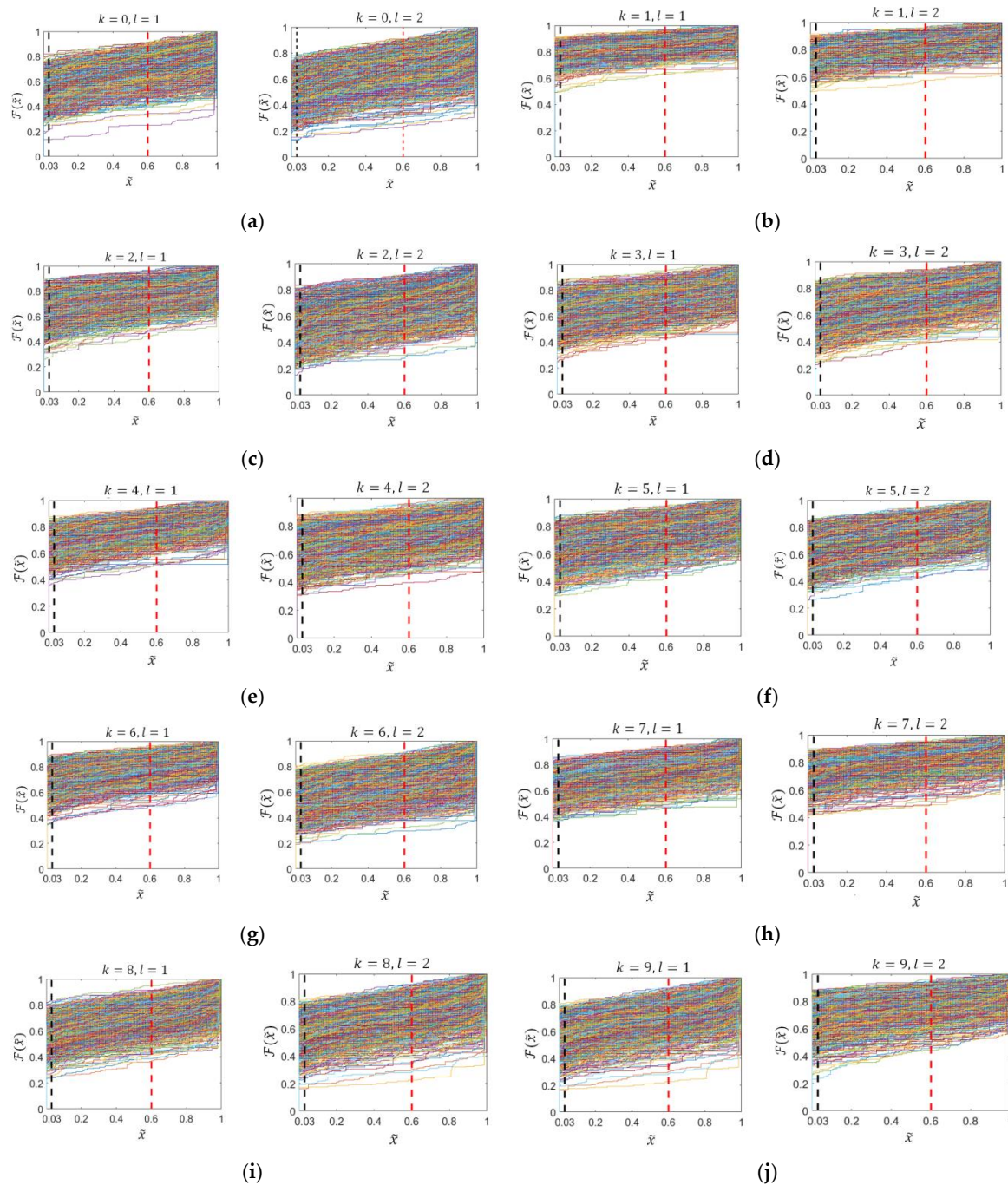


Figure 5. ECDF results of image class $k = 0, 1, \dots, 9$, each panel represents a different image class. The results are divided into two subsets, $l = 1, 2$, with distinguishing points $T_1 = 0.03$ (black vertical dotted line) and $T_2 = 0.6$. (red vertical dotted line).

The results of the ECDF indicate that there are variations in the distribution patterns among different image classes. The resulting ECDF in class 1 of the picture dataset does not exhibit a widely dispersed distribution throughout the images. This is a result of using a smaller number of pixels that are less diverse. On the other hand, certain other categories of

images exhibit a more scattered distribution pattern. This is attributed to the use of a wider range of pixels, resulting in a more diverse distribution. The distribution patterns offer crucial data regarding the attributes and fluctuations in pixel intensity within each image category. Subsequent image analysis and classification can utilize this information. This thorough research allows for a comprehensive comprehension of how various partitions and distinguishing points impact the ECDFs across different image classes. Preprocessing the pixel values before generating the ECDF enhances the reliability of comparing the distributions of pixel intensities. This approach offers a systematic approach to capturing the range of pixel intensities, enabling more accurate categorization and examination of images using their ECDF.

3.3. Theoretical Results and DFDV Application

We can determine the distinguishing points, which distinguish characteristics between image classes for each image partition, from the ECDF symbolic data. In this case, take the distinguishing points $T_1 = 0.03$ and $T_2 = 0.6$. At the separator point, Equation (8) DFDV function can be defined. As an illustration, consider the formation of the DFDV in the image class $k = 0$ with the partition $l = 2$ in Figure 6.

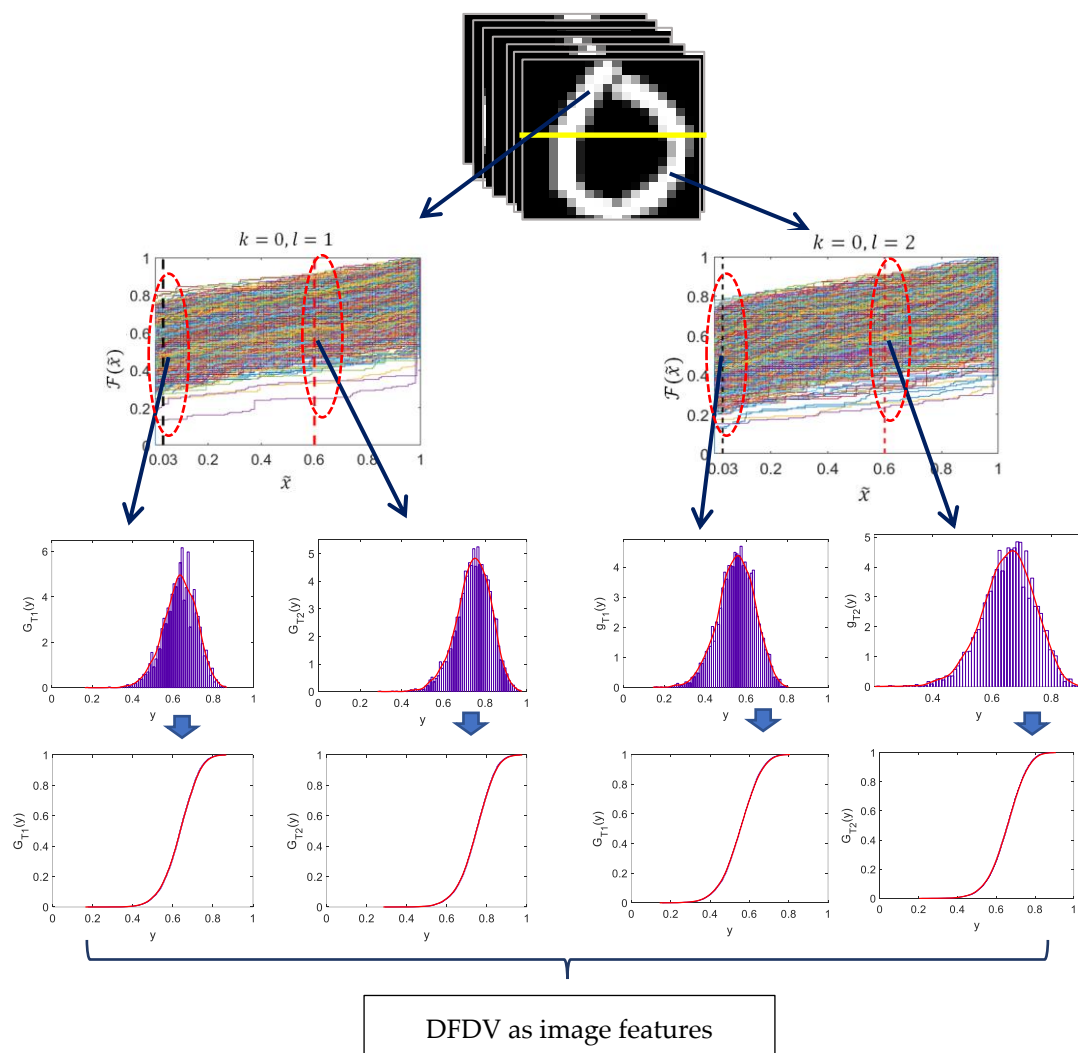


Figure 6. Formation at differentiation points $T_1 = 0.03$ (black vertical dotted line) and $T_2 = 0.6$ (red vertical dotted line).

The set of ECDF values at each distinguishing point forms a DFDV. This DFDV characterizes the image features in the partition and describes the likelihood that the distribution of image features has a value less than or equal to a certain threshold. For each image class k and partition l , $\mathcal{F}_{klj}(T)$ represents the symbolic data of the ECDF value for the image class k , the partition l , and the image j at point T . If $\mathfrak{F} = \{\mathcal{F}_{kl1}(T), \mathcal{F}_{kl2}(T), \dots, \mathcal{F}_{klN_k}(T)\}$ is the collection of $\mathcal{F}_{klj}$ values at point T , we can then form a DFDV, denoted as $G_T(y)$, where y represents the realized ECDF probability value at point T . The DFDV provides a comprehensive statistical depiction of the variability and distribution of pixel intensity across various image classes and partitions. The DFDV utilizes symbolic ECDF data to translate raw pixel values, enabling a systematic approach to image comparison and analysis. In the upcoming sections, we will clearly define and thoroughly examine the characteristics of the DFDV.

Definition 1. *Distribution Function of Distribution Values (DFDV).*

The distribution function refers to the mathematical function that describes the probability distribution of a given value in a statistical distribution. The distribution function G_T is defined as the function that represents the distribution value at point T . $G_T : [0, 1] \rightarrow [0, 1], y \mapsto G_T(y)$ with

$$G_T(y) = P(\mathcal{F}_{klj} \in \mathfrak{F} | \mathcal{F}_{klj}(T) \leq y) \quad (13)$$

The function $G_T(y)$ maps the value of y in the interval $[0, 1]$ to the associated probability, with the domain of y ranging from $[0, 1]$ to $[0, 1]$. Multiple points, T , can be defined, leading to different distribution functions $G_T(y)$. Let us represent this as $T = (T_1, T_2, \dots, T_s)$ and $= (y_1, y_2, \dots, y_s)$. Each point T_t , where t is a positive integer ranging from 1 to s , corresponds to a unique distribution function $G_{T_t}(y_t)$.

Each function $G_{T_t}(y_t)$ is a distribution function of a random variable $\mathcal{F}_{klj}(T_t)$ with values in the interval $[0, 1]$. As a distribution function, each function $G_{T_t}(y_t)$ has the properties of a distribution function, which yields the following theorem:

Theorem 1. *The properties of distribution functions of distribution values.*

The following conditions apply to every continuous random variable $\mathcal{F}(T_k)$ that has a cumulative distribution function $G_{T_t}(y_t)$:

1. The function $G_{T_t}(y_t)$ is monotonically increasing, which implies that if $y_1 < y_2$ then $G_{T_t}(y_1) < G_{T_t}(y_2)$.
2. As y_t approaches negative infinity, the limit of the distribution function, denoted as $G_{T_t}(y_t)$ is equal to zero: $\lim_{y_t \rightarrow -\infty} G_{T_t}(y_t) = 0$
3. As y_t approaches positive infinity, the limit of the distribution function, denoted as $G_{T_t}(y_t)$, is equal to: $\lim_{y_t \rightarrow \infty} G_{T_t}(y_t) = 1$
4. The distribution functions of distribution values is right-continuous: $\lim_{\delta \rightarrow 0} G_{T_t}(y_t + \delta) = G_{T_t}(y_t)$.

Proof of Theorem 1.

1. Given that $y_1 < y_2$, our objective is to demonstrate that $G_{T_t}(y_1) < G_{T_t}(y_2)$. According to the definition, $G_{T_t}(y_1) = P(\mathcal{F}(T_t) \leq y_1)$, whereas $G_{T_t}(y_2) = P(\mathcal{F}(T_t) \leq y_2)$. Given that $y_1 < y_2$, we can conclude that the set of values for which $\{\mathcal{F}(T_t) \leq y_1\}$ is a subset of the set of values for which $\{\mathcal{F}(T_t) \leq y_2\}$. Simply expressed, $\{\mathcal{F}(T_t) \leq y_1\} \subseteq \{\mathcal{F}(T_t) \leq y_2\}$. Given that the likelihood of a set being a subset of another set can never exceed the probability of the larger set, we can conclude that $P(\mathcal{F}(T_t) \leq y_1) \leq P(\mathcal{F}(T_t) \leq y_2)$. This expression represents $G_{T_t}(y_1) < G_{T_t}(y_2)$.
2. Consider the series $\{y_t : t = 1, 2, \dots\}$, which consists of real integers that are infinitely negative and decreasing. We observe that when $y_t \rightarrow -\infty$ when $t \rightarrow \infty$. Define the set $A_{y_t} = \{\mathcal{F}(T_t) \leq y_t\} = G_{T_t}(y_t)$ for each t . Put simply, A_{y_t} is the set ω in the sample

space Ω where the value of $\mathcal{F}(T_t)$ is smaller than or equal to y_t . Consider Ω as the sample space that encompasses all the values of the random variable $\mathcal{F}(T_t)$, and let ω represent one of the possible values of $\mathcal{F}(T_t)$. Given that y_t is a decreasing sequence, if $y_t > y_{t+1}$, then there exists a set $A_{y_t} \supseteq A_{y_{t+1}}$. Put simply, every set A_{y_t} includes all the components found in set $A_{y_{t+1}}$. Given that A_{y_t} is a superset of $A_{y_{t+1}}$, we can conclude that the intersection $\bigcap_{t=1}^{\infty} A_{y_t} = \emptyset$. Therefore $\lim_{t \rightarrow \infty} G_{T_t}(y_t) = \lim_{t \rightarrow \infty} P(A_{y_t}) = P(\bigcap_{t=1}^{\infty} A_{y_t}) = P(\emptyset) = 0$. Given that it applies to every sequence $\{y_t\}$, in whatever order, we may deduce that when $\{y_t\} \rightarrow -\infty$, concluded $\lim_{y_t \rightarrow -\infty} G_T(y_t) = 0$.

3. Idem for point 2.

4. A function $G_{T_t}(y_t)$ is considered to be right-continuous if $\lim_{\delta \rightarrow 0} G_{T_t}(y_t + \delta) = G_{T_t}(y_t)$. Consider a sequence of real numbers $\{y_t : t = 1, 2, \dots\}$ that is decreasing. As $t \rightarrow \infty$, then $y_t \rightarrow y$. For every value of y_n , there exists a set $A_{y_t} = \{\mathcal{F}(T_t) \leq y_t\}$, such that $A_{y_t} \subseteq A_{y_{t+1}}$. Further A_y is defined as the intersection of all sets A_{y_t} , denoted as $\bigcap_{t=1}^{\infty} A_{y_t} = A_y$. Therefore, $\lim_{t \rightarrow \infty} G_T(y_t) = \lim_{t \rightarrow \infty} P(A_{y_t}) = P(\bigcap_{t=1}^{\infty} A_{y_t}) = P(A_y) = G_T(y)$. Given that it applies to every sequence $\{y_t\}$ regardless of the order, we can deduce that as $\{y_t\} \rightarrow -\infty$, the limit as $\lim_{\delta \rightarrow 0} G_{T_t}(y_t + \delta) = G_{T_t}(y_t)$ for all values of y . Therefore, $G_T(y)$ exhibits the right continuity. $\square$

3.4. Results of the Kolmogorov–Smirnov Goodness of Fit Test

The DFDV function will adhere to a specific distribution established by the Kolmogorov–Smirnov model fit test, which utilizes Equation (10). Based on the results we saw, this DFDV does not follow a parametric distribution. This means that a non-parametric kernel density estimation (KDE) distribution has to be used. For instance, there is a set of ECDFs at a specific point T , and we are interested in determining the distribution function. We calculate the KDE estimate at position y using the following procedure:

$$\hat{f}(y) = \frac{1}{N_k} \sum_{j=1}^{N_k} K\left(\frac{y - y_j}{h}\right) \quad (14)$$

where N_k represents the number of data points, K represents the kernel function, y_j represents the data point j , and h represents the bandwidth width.

The probability density function $\hat{f}(y)$ is estimated to characterize the data's probability distribution. The parameter h will influence the level of smoothness and sharpness in the KDE estimation. The estimation uniformly spreads the estimated probabilities across the curve as the value of h increases. Conversely, the smaller h value will generate a probability estimator with more distinct and pronounced peaks. This implies that the probability estimate will be more sensitive to fluctuations in the data [30]. Figure 7 provides the DFDV findings for each class, each partition, and each differentiating point. Table 1 presents the results of the Kolmogorov–Smirnov distribution matching test.

The Kolmogorov–Smirnov distribution fit test for the kernel density estimation (KDE) yielded $p_{value} < \alpha$ for all picture classes. This suggests that the DFDV adheres to the non-parametric distribution of KDE. Table 1 demonstrates that the estimation of the bandwidth width parameter (h) yields a diminutive result. A lower bandwidth value indicates that the KDE produces a more precise and sensitive density curve to variations in the data. The KDE density curve yields the following results (Figure 8):

The distributions formed show different distribution patterns, which highlight the unique characteristics of each image class. Appropriate bandwidth values ensure that KDE effectively captures the variations and nuances in the data for each image class.

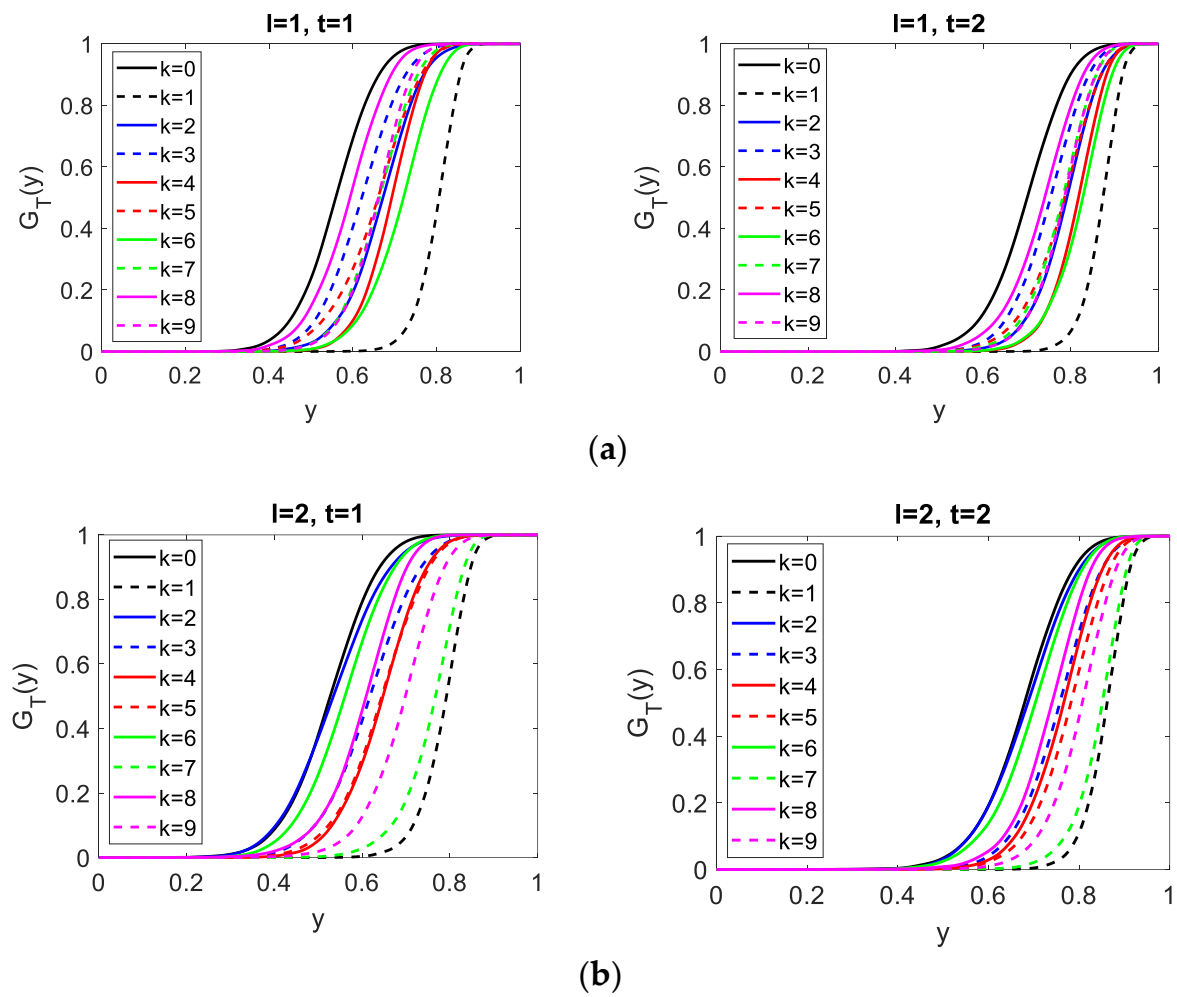


Figure 7. DFDV results on partition (a) $l = 1$, and (b) $l = 2$, with distinguishing points $T_1 = 0.03$ and $T_2 = 0.6$.

Table 1. KDE distribution fit test results using the Kolmogorov–Smirnov test.

k	$l=1$				$l=2$			
	$t = 0.03$		$t = 0.6$		$t = 0.03$		$t=0.6$	
	h	p_{value}	h	p_{value}	h	p_{value}	h	p_{value}
0	0.0152	0.0511	0.0152	0.0731	0.0166	0.1048	0.0166	0.0997
1	0.0081	0.0504	0.0067	0.0502	0.0094	0.0502	0.0081	0.0505
2	0.0145	0.0852	0.0124	0.0537	0.0193	0.1910	0.0179	0.0903
3	0.0151	0.0741	0.0137	0.0506	0.0165	0.1347	0.0151	0.0751
4	0.0125	0.0545	0.0110	0.0529	0.0152	0.0830	0.0139	0.0561
5	0.0169	0.0685	0.0140	0.0503	0.0167	0.0973	0.0141	0.1069
6	0.0138	0.0534	0.0110	0.0593	0.0166	0.1490	0.0166	0.1475
7	0.0137	0.0597	0.0110	0.0567	0.0109	0.0037	0.0096	0.0010
8	0.0152	0.0632	0.0138	0.0546	0.0152	0.0765	0.0138	0.0751
9	0.0124	0.0575	0.0110	0.0059	0.0138	0.0560	0.0124	0.0154

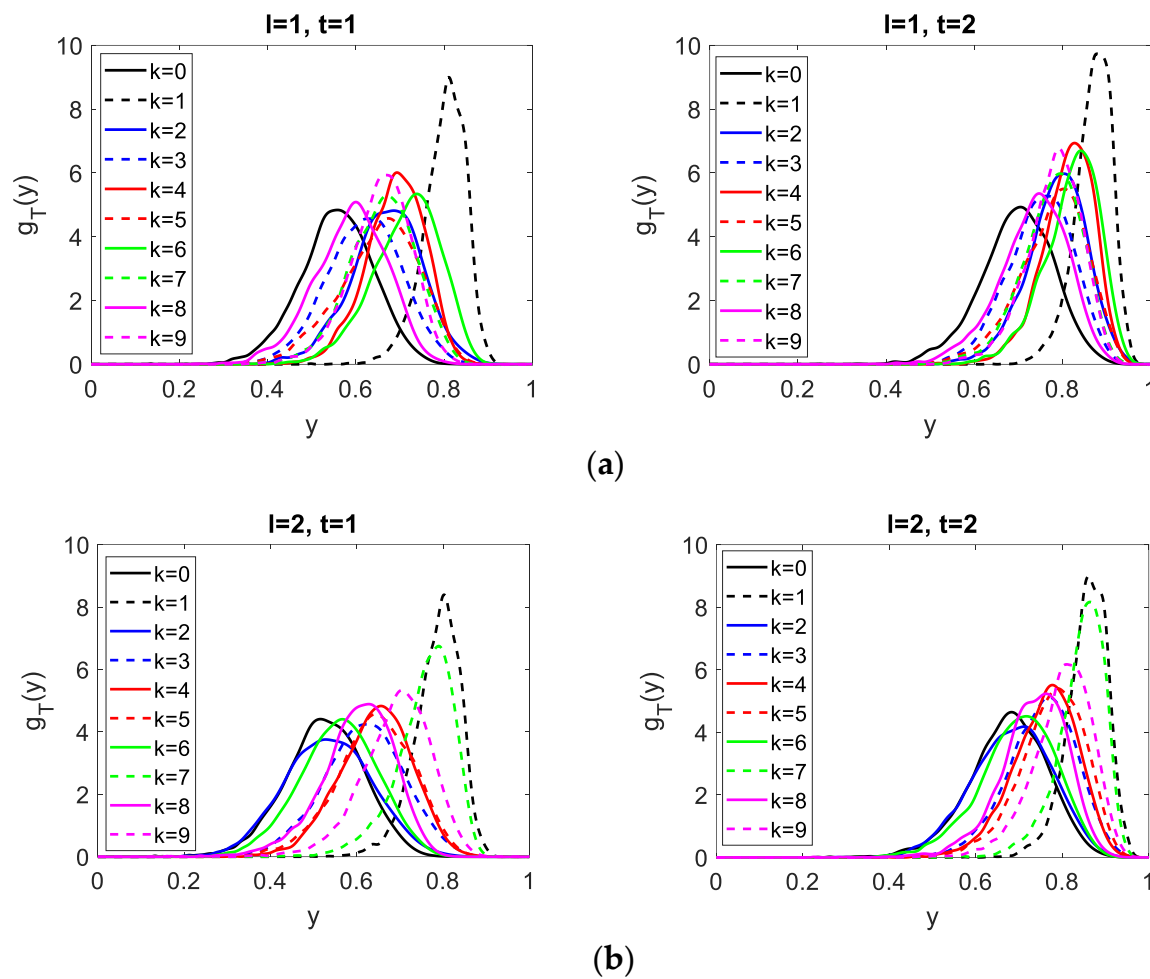


Figure 8. KDE function curve results at partition (a) $l = 1$, and (b) $l = 2$, with distinguishing points $T_1 = 0.03$ and $T_2 = 0.6$.

4. Discussion

Symbolic data analysis is an emerging discipline within statistics, showing promising potential to become a widely accepted inferential technique in the future. Unlike standard methods, symbolic data analysis prioritizes group-level data summaries, referred to as symbols, for both exploratory analysis and statistical inference. It uses group-level summary distributions as a specific statistical unit of focus [31]. Symbolic data offer the benefit of minimizing computational requirements by consolidating individual-level data into a smaller number of group-level symbols. This methodology is ideal for examining extensive and intricate datasets by condensing them into more understandable formats. By considering distribution as a key unit of analysis, symbolic data analysis aligns with the belief that future data analysis will increasingly rely on comprehending and handling aggregated data.

Image features can be extracted as symbolic data using the empirical cumulative distribution function (ECDF), an alternative to traditional image analysis. The ECDF can reduce large pixel data dimensions to pixel intensity distribution values. The ECDF value describes the proportion of pixel intensity that is less than a given value. From a set of ECDF values at a specific differentiation point, we can form the distribution function distribution (DFDV), which describes the probability that a distribution will have a value less than or equal to a specific value at a specific point. The ECDF and DFDV represent images using distribution functions, allowing for image classification based on differences between various image classes.

This research aligns with other studies that depict image features not just as a single pixel intensity value. Among these are image feature extraction using color and texture [32], and spectral feature extraction [8,11]. Recent developments include image representations in the form of graphs and hypergraphs [9], as well as deep learning [10]. These image features provide richer and more detailed information than pixel value information alone. Using these representations makes image analysis more accurate and comprehensive, enabling more complex pattern identification and improved performance in various applications, such as image classification and segmentation.

The ECDF results provide a strong basis for image grouping based on pixel value distribution. Understanding these patterns allows for the development of more effective and accurate clustering algorithms. Furthermore, machine learning models can utilize these results as features to enhance the accuracy of image classification. We can also use DFDV symbolic data representations for clustering analysis. Despite the useful information provided by the ECDF results, this study faces challenges and limitations. Selecting an optimal differentiation point and correctly transforming pixel values require careful adjustment. Furthermore, more in-depth analysis is necessary to fully understand the variations in pixel value distributions within different classes.

Future research will focus on using ECDF and DFDV as random variables in joint distribution function-based classification. This research is in line with studies that use a joint distribution function approach for image classification [33–38]. By leveraging the joint distribution of pixel values, we aim to enhance the accuracy and effectiveness of image classification methods. This approach represents a shift from traditional pixel-based classification to a more sophisticated method that considers the distributional properties of image features.

5. Conclusions

Image feature extraction using symbolic data with ECDF and DFDV offers an efficient way to analyze large and complex images by reducing the dimensions of pixel data. This method allows for more accurate grouping and classification of images. Although useful, this method requires adjusting the differentiation point and transforming the correct pixel value. Further research is needed to optimize this method and test its reliability on more diverse datasets.

Author Contributions: Conceptualization, S.W. and S.W.I.; methodology, S.W.; software, S.W. and R.A.; validation, S.W.I. and R.S.P.; formal analysis, S.W.; investigation, S.W.I.; resources, R.A.; data curation, R.A. and R.S.P.; writing—original draft preparation, S.W. and S.W.I.; writing—review and editing, R.A. and R.S.P.; visualization, R.S.P.; supervision, S.W.I.; project administration, S.W.; funding acquisition, R.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Hibah Riset Data Pustaka dan Daring (RDPD) Universitas Padjadjaran, grant number 2125/UN.6.3.1/PT.00/2024 and The APC was funded by Padjadjaran University.

Data Availability Statement: The data used in this research can be downloaded at <http://yann.lecun.com/exdb/mnist/> (accessed on 4 May 2021).

Acknowledgments: Thanks to the Directorate of Research, Community Service, and Innovation of Universitas Padjadjaran for providing research grants.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Nixon, M.S.; Aguado, A.S. *Feature Extraction and Image Processing for Computer Vision*; Academic Press Elsevier: Amsterdam, The Netherlands, 2019.
2. Bishop, C.M. *Pattern Recognition and Machine Learning*; Springer: New York, NY, USA, 2006.
3. Ranjan, R.; Patel, V.M.; Chellappa, R. HyperFace: A Deep Multi-Task Learning Framework for Face Detection, Landmark Localization, Pose Estimation, and Gender Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *41*, 121–135. [CrossRef] [PubMed]

4. Seo, Y.; Shin, K. Hierarchical convolutional neural networks for fashion image classification. *Expert Syst. Appl.* **2019**, *116*, 328–339. [CrossRef]
5. Liu, L.; Wang, X. Plant diseases and pests detection based on deep learning: A review. *Plant Methods* **2021**, *17*, 22. [CrossRef] [PubMed]
6. Yamashita, T.K.; Nishio, M.; Do, R.K.G. Convolutional neural networks: An overview and application in radiology. *Insights Imaging* **2018**, *9*, 611–629. [CrossRef] [PubMed]
7. Kalantari, L.; Gader, P.; Graves, S.; Bohlman, S.A. One-Class Gaussian Process for Possibilistic Classification Using Imaging Spectroscopy. *IEEE Geosci. Remote Sens. Lett.* **2016**, *13*, 967–971. [CrossRef]
8. Zhao, Z.; Liao, G. Imaging Hyperspectral Feature Fusion for Estimation of Rapeseed Pod's Water Content and Recognition of Pod's Maturity Level. *Mathematics* **2024**, *12*, 1693. [CrossRef]
9. Qu, Y.; Fu, K.; Wang, L.; Zhang, Y.; Wu, H.; Liu, Q. Hypergraph-Based Multitask Feature Selection with Temporally Constrained Group Sparsity Learning on fMRI. *Mathematics* **2024**, *12*, 1733. [CrossRef]
10. Naem, A.; Anees, T.; Khalil, M.; Zahra, K.; Naqvi, R.A.; Lee, S.W. SNC_Net: Skin Cancer Detection by Integrating Handcrafted and Deep Learning-Based Features Using Dermoscopy Images. *Mathematics* **2024**, *12*, 1030. [CrossRef]
11. Zhu, W.; Zhang, X.; Zhu, Z.; Fu, W.; Liu, N.; Zhang, Z. A Rapid Detection Method for Coal Ash Content in Tailings Suspension Based on Absorption Spectra and Deep Feature Extraction. *Mathematics* **2024**, *12*, 1685. [CrossRef]
12. Lin, X.; Chen, R.; Feng, C.; Chen, Z.; Yang, X.; Cui, H. Automatic Evaluation Method for Functional Movement Screening Based on a Dual-Stream Network and Feature Fusion. *Mathematics* **2024**, *12*, 1162. [CrossRef]
13. Fei-Fei, L.; Deng, J.; Li, K. ImageNet: Constructing a large-scale image database. *J. Vis.* **2010**, *9*, 8. [CrossRef]
14. Lecun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [CrossRef]
15. Zhang, Z.; Sabuncu, M.R. Generalized cross entropy loss for training deep neural networks with noisy labels. *Adv. Neural Inf. Process. Syst.* **2018**, *2018*, 8778–8788.
16. Billard, L.; Diday, E. From the statistics of data to the statistics of knowledge: Symbolic data analysis. *J. Am. Stat. Assoc.* **2003**, *98*, 470–487. [CrossRef]
17. Billard, L.; Diday, E. *Symbolic Data Analysis: Conceptual Statistics and Data Mining*; John Wiley & Sons, Ltd.: Hoboken, NJ, USA, 2007.
18. Vrac, M.; Chédin, A.; Diday, E. Clustering a global field of atmospheric profiles by mixture decomposition of copulas. *J. Atmos. Ocean. Technol.* **2005**, *22*, 1445–1459. [CrossRef]
19. Vrac, M.; Billard, L.; Diday, E.; Chédin, A. Copula analysis of mixture models. *Comput. Stat.* **2012**, *27*, 427–457. [CrossRef]
20. Winarni, S.; Indratno, S.W.; Sari, K.N. Pemodelan Gambar Menggunakan Copula Gaussian Dengan Metode Partisi. *Stat. J. Theor. Stat. Appl.* **2021**, *21*, 37–43. [CrossRef]
21. Winarni, S.; Indratno, S.W.; Sari, K.N. Character of images development on gaussian copula model using distribution of cumulative distribution function. *Commun. Math. Biol. Neurosci.* **2021**, *2021*, 86. [CrossRef]
22. Gonzalez, R.C.; Richard, E.W. *Digital Image Processing*; Pearson: New York, NY, USA, 1980.
23. Nixon, M.S.; Aguado, A.S. Basic image processing operations. In *Feature Extraction & Image Processing for Computer Vision*; Elsevier: Amsterdam, The Netherlands, 2012; pp. 83–136. [CrossRef]
24. Awad, A.I.; Hassaballah, M. Studies in Computational Intelligence 630. In *Image Feature Detectors and Descriptors, Foundations and Applications*; Springer: Berlin/Heidelberg, Germany, 2016.
25. Castro, R. Introduction and the Empirical CDF. 2013, pp. 1–10. Available online: <https://www.win.tue.nl/~rmcastro/AppStat2013/files/lecture1.pdf> (accessed on 23 August 2021).
26. Diday, E.; Vrac, M. Mixture decomposition of distributions by copulas in the symbolic data analysis framework. *Discrete Appl. Math.* **2005**, *147*, 27–41. [CrossRef]
27. Gibbons, J.D.; Chakraborti, S. *Nonparametric Statistical Inference*; Marcel Dekker AG: Basel, Switzerland, 2010.
28. Chaki, J.; Dey, N. *A Beginner's Guide to Image Preprocessing Techniques*; Taylor & Francis Group: Abingdon, UK, 2019.
29. Henderi, H. Comparison of Min-Max normalization and Z-Score Normalization in the K-nearest neighbor (kNN) Algorithm to Test the Accuracy of Types of Breast Cancer. *IJIS Int. J. Inform. Inf. Syst.* **2021**, *4*, 13–20. [CrossRef]
30. Duong, J.E.C.T. *Multivariate Kernel Smoothing and Its Applications*; Taylor & Francis Group: Abingdon, UK, 2018.
31. Beranger, B.; Lin, H.; Sisson, S. New models for symbolic data analysis. *Adv. Data Anal. Classif.* **2023**, *17*, 659–699. [CrossRef]
32. Sun, Z.; Zhang, K.; Zhu, Y.; Ji, Y.; Wu, P. Unlocking Visual Attraction: The Subtle Relationship between Image Features and Attractiveness. *Mathematics* **2024**, *12*, 1005. [CrossRef]
33. Salinas-Gutiérrez, R.; Hernández-Aguirre, A.; Rivera-Meraz, M.J.J.; Villa-Diharce, E.R. Using Gaussian Copulas in supervised probabilistic classification. *Stud. Comput. Intell.* **2010**, *318*, 355–372. [CrossRef]
34. Bansal, R.; Hao, X.; Liu, J.; Peterson, B.S. Using Copula distributions to support more accurate imaging-based diagnostic classifiers for neuropsychiatric disorders. *Magn. Reson. Imaging* **2014**, *32*, 1102–1113. [CrossRef] [PubMed]
35. Stitou, Y.; Lasmar, N.; Berthoumieu, R. Copulas based multivariate gamma modeling for texture classification. In Proceedings of the 2009 IEEE International Conference on Acoustics, Speech and Signal Processing, Taipei, Taiwan, 19–24 April 2009; pp. 1045–1048. [CrossRef]
36. Lasmar, N.E.; Berthoumieu, Y. Gaussian copula multivariate modeling for texture image retrieval using wavelet transforms. *IEEE Trans. Image Process.* **2014**, *23*, 2246–2261. [CrossRef]

37. Bauer, A.; Czado, C.; Klein, T. Pair-copula constructions for non-Gaussian DAG models. *Can. J. Stat.* **2012**, *40*, 86–109. [[CrossRef](#)]
38. Li, C.; Huang, Y.; Xue, Y. Dependence structure of Gabor wavelets based on copula for face recognition. *Expert Syst. Appl.* **2019**, *137*, 453–470. [[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.