

Article

A New Texture Spectrum Based on Parallel Encoded Texture Unit and Its Application on Image Classification: A Potential Prospect for Vision Sensing

José Trinidad Guillen Bonilla ^{1,*}, Nancy Elizabeth Franco Rodríguez ², Héctor Guillen Bonilla ³, Alex Guillen Bonilla ⁴, Verónica María Rodríguez Betancourt ³, Maricela Jiménez Rodríguez ⁵, María Eugenia Sánchez Morales ⁶ and Oscar Blanco Alonso ⁷

- ¹ Departamento de Electro-Fotónica, Centro Universitario de Ciencias Exactas e Ingenierías, Universidad de Guadalajara, Blvd.-M. García Barragán 1421, Guadalajara 44430, Jalisco, Mexico
 - ² Departamento de Farmacología, Centro Universitario de Ciencias Exactas e Ingenierías, Universidad de Guadalajara, Blvd.-M. García Barragán 1421, Guadalajara 44430, Jalisco, Mexico; nancy.frodriguez@academicos.udg.mx
 - ³ Departamento de Ingeniería de Proyectos, Centro Universitario de Ciencias Exactas e Ingenierías, Universidad de Guadalajara, Blvd.-M. García Barragán 1421, Guadalajara 44430, Jalisco, Mexico; hector.guillen1775@academicos.udg.mx (H.G.B.); veronica.rbetancourt@academicos.udg.mx (V.M.R.B.)
 - ⁴ Departamento de Ciencias Computacionales e Ingenierías, CUVALLES, Universidad de Guadalajara, Carretera Guadalajara-Ameca Km. 45.5, Ameca 46600, Jalisco, Mexico; alex.guillen@academicos.udg.mx
 - ⁵ Departamento de Ciencias Básicas, Centro Universitario de la Ciénega (CUCiénega), Universidad de Guadalajara, Av. Universidad No. 1115, Linda Vista, Ocotlán 47810, Jalisco, Mexico; maricela.jrodriguez@academicos.udg.mx
 - ⁶ Departamento de Ciencias Tecnológicas, Centro Universitario de la Ciénega (CUCiénega), Universidad de Guadalajara, Av. Universidad No. 1115, Linda Vista, Ocotlán 47810, Jalisco, Mexico; eugenia.sanchez@academicos.udg.mx
 - ⁷ Departamento de Física, Centro Universitario de Ciencias Exactas e Ingenierías, Universidad de Guadalajara, Blvd.-M. García Barragán 1421, Guadalajara 44430, Jalisco, Mexico; oscar.blanco@academicos.udg.mx
- * Correspondence: trinidad.guillen@academicos.udg.mx; Tel.: +52-(33)-1378-5900 (ext. 27655)



Citation: Guillen Bonilla, J.T.; Franco Rodríguez, N.E.; Guillen Bonilla, H.; Guillen Bonilla, A.; Rodríguez Betancourt, V.M.; Jiménez Rodríguez, M.; Sánchez Morales, M.E.; Blanco Alonso, O. A New Texture Spectrum Based on Parallel Encoded Texture Unit and Its Application on Image Classification: A Potential Prospect for Vision Sensing. *Sensors* **2023**, *23*, 8368. <https://doi.org/10.3390/s23208368>

Academic Editors: Hong-Dar Lin, Hsin-Chieh Wu and Cheng-Hsiung Hsieh

Received: 12 September 2023
Revised: 4 October 2023
Accepted: 8 October 2023
Published: 10 October 2023



Copyright: © 2023 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/>).

Abstract: In industrial applications based on texture classification, efficient and fast classifiers are extremely useful for quality control of industrial processes. The classifier of texture images has to satisfy two requirements: It must be efficient and fast. In this work, a texture unit is coded in parallel, and using observation windows larger than 3×3 , a new texture spectrum called Texture Spectrum based on the Parallel Encoded Texture Unit (TS_PETU) is proposed, calculated, and used as a characteristic vector in a multi-class classifier, and then two image databases are classified. The first database contains images from the company Interceramic[®] and the images were acquired under controlled conditions, and the second database contains tree stems and the images were acquired in natural environments. Based on our experimental results, the TS_PETU satisfied both requirements (efficiency and speed), was developed for binary images, and had high efficiency, and its compute time could be reduced by applying parallel coding concepts. The classification efficiency increased by using larger observational windows, and this one was selected based on the window size. Since the TS_PETU had high efficiency for Interceramic[®] tile classification, we consider that the proposed technique has significant industrial applications.

Keywords: texture spectrum based on parallel encoded texture unit (TS_PETU); multi-class classifier; image classification; high efficiency

1. Introduction

Currently, image recognition based on the textural characteristics of images has a large number of applications in biomedicine [1,2], object detection [3,4], medical diagnostics [5–7], and remote sensing [8,9], among others. Because of this, around the world there is a large

number of research groups that carry out research on texture analysis, extraction, and/or classification. The goal of these groups is to develop new applications whose objective is to improve quality of life.

In texture analysis, there are three main problems: texture classification, where the goal is to determine to which class a test texture belongs; texture segmentation, where the goal is to partition an image into sections based on the textures that make it up; and texture synthesis, where the goal is to generate a mathematical model in order to build the desired texture. In particular, in texture classification, the extraction of textural features is very important, according to the author of reference [10]. There are four methods to solve the problem: geometry, mathematical models, signal processing, and statistics.

The statistical method for extracting textural characteristics basically consists of selecting an observation window, the size of which is frequently $W = 3 \times 3$ pixels [11]. By scrolling the window pixel by pixel over the entire image, patterns are detected and encoded to calculate the texture unit [12], which we will denote as k . The unit value k depends on the encoding method, as indicated in reference [11], in which the authors describe, apply, and compare 35 different texture extraction techniques. However, in all methods, the unit k is considered a discrete variable and is used as an index in a discrete histogram $h(k)$, whose length is from 0 to $K - 1$. The histogram $h(k)$ has R^K dimensions, where the value of K is the maximum texture unit value. Now, since the objective is to apply to the histogram $h(k)$ in image classification, it is interpreted as a texture spectrum and then is used as a characteristic vector in supervised (multi-class and one-class) and unsupervised (clustering) classifiers [10,13–17]. Such classification systems operate in real time and efficiently [18–20]; when the spectrum $h(k)$ has low dimensional space, the histogram contains a sufficient amount of texture information of the image under study, the classifier is optimized, and the electronic device is task-specific.

In this work, by conducting a local analysis on a binary image $s(m, n)$ through an observation window $W = I \times J$, an image $s(m, n)$ is represented by a probability density function $p_{I \times J}(k)$, where m, n are the coordinates of the pixels of the digital image and k is the unit of texture. The equalized histogram $p_{I \times J}(k)$ is called Texture Spectrum Based on the Parallel Encoded Texture Unit (TS_PETU) because the unit k is coded using parallel coding concepts. Due to parallel coding, the TS_PETU histogram can be computed using larger windows at $W = 3 \times 3$, and as a consequence, they contain a greater amount of texture information and their dimensional space can be selected since the probability function $p_{I \times J}(k)$ has $R^{I(2^J-1)+1}$ dimensions. Based on the behavior of $R^{I(2^J-1)+1}$, two regions are defined: low-dimensional space and high-dimensional space. In the first region, the TS_PETU histogram has from R^{22} up to $R^{10,231}$ dimensions and the window size is within the range of $W = 3 \times 3$ to $W = 10 \times 10$. On the other hand, in the second region, the TS_PETU histogram satisfies the condition $R^K > R^{10,231}$ dimension, and then the window size must satisfy $I \times J > 10 \times 10$.

By interpreting the TS_PETU histogram as a texture spectrum, it can be used as a feature vector in a multi-class classifier and then two image databases can be classified. The first database was acquired under controlled conditions by the Interceramic[®] company, and the images in the second database were acquired in natural environments. With the goal of verifying the classification efficiency of our proposal, three experiments were developed. In the first experiment, using window sizes within the range of $I \times J = 3 \times 3$ to $I \times J = 20 \times 20$, texture information was measured for Interceramic[®] tile images. Our experimental results confirm that the amount of texture information contained in the TS_PETU histogram increased due to the observation window size and its behavior was exponential, and the theoretical results are in agreement with the experimental results. In the second experiment, the TS_PETU histogram was calculated using observation windows from $I \times J = 3 \times 3$ to $I \times J = 20 \times 20$. It was used as a feature vector in a multi-class classifier and then the Interceramic[®] images were classified. Our experimental results confirm the high efficiency of the TS_PETU transform since the classification accuracy was 100%. This high efficiency is attributed to the precision with which the TS_PETU transform

can work and because the images were acquired under controlled conditions. In the third experiment, using windows within the interval of $I \times J = 3 \times 3$ to $I \times J = 10 \times 10$, the TS_PETU histogram was calculated. Next, this was used as a characteristic vector in the classifier for multiple classes, and subsequently, the images acquired in natural environments were classified. In our results, the classification efficiency was within the range of $Ef_{I \times J} = 84.84\%$ ($I \times J = 3 \times 3$) to 100% ($I \times J = 5 \times 5$ or greater). The classification errors are attributed to the fact that the images were acquired under uncontrolled conditions. The increase in efficiency can be attributed to the increase in texture information due to the size of the observation window.

Based on the analysis performed on the binary image and our experimental results, the following relevant points were inferred for the TS_PETU transform: (1) The transformation has potential industrial applications, (2) its dimensional space and texture information can be selected based on the observation window size, (3) its classification efficiency improves when the observation window is larger, (4) the transformation has potential real-time application due to the parallel encoding of the texture unit, and (5) vision systems can be implemented for quality control.

2. Texture Spectrum Based on Parallel Encoded Texture Unit

2.1. Proposal Methodology

Figure 1 shows the proposed procedure schematically. The methodology consists of two phases: calculation of the texture spectrum and its application. In the first phase, called texture spectrum calculation, the digital image $s(m, n)$ is interpreted as a binary matrix $S = \{s_{mn}\}$, where the white pixels are ones and the black pixels are zeros. Once the observation window size $I \times J$ is selected, we describe the procedure to calculate the texture unit k , the encoding of which is done in parallel. Subsequently, the unit k is used as an index in the histogram $h_{I \times J}(k)$ and is also used to calculate the dimensional space R^K . Finally, an algorithm is proposed to calculate the histogram $h_{I \times J}(k)$, and with it, a probability density function is defined, $p_{I \times J}(k)$. This function is named Texture Spectrum based on the Parallel Encoded Texture Unit (TS_PETU). In the second phase, called application, the TS_PETU histogram is used to measure texture information and applied in image recognition. Both the texture measurement and classification efficiency are experimentally validated using two image databases. The first is of industrial origin and the acquisition conditions are controlled, whereas in the second database, the images were acquired in natural environments.

2.2. Parallel Encoded Texture Unit

With the goal of obtaining a reduced dimensional space in the texture spectrum, in this section, the texture unit is coded in parallel. The procedure is described below.

Let $s(m, n)$ ($m = 1, 2, \dots, M; n = 1, 2, \dots, N$) be a binary image with $M \times N$ pixels, which is interpreted as a binary matrix $\{s_{mn}\}$ with M rows and N columns, and let W an observation window be of the size $W = I \times J$. Then, for each position on the matrix $\{s_{mn}\}$,

the window W detects a binary pattern denoted by $P = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1J} \\ a_{21} & a_{22} & \cdots & a_{2J} \\ \vdots & \vdots & \ddots & \vdots \\ a_{I1} & a_{I2} & \cdots & a_{IJ} \end{bmatrix}$, whose

number of rows is I and number of columns is J , and the number of patterns P in the matrix $\{s_{mn}\}$ is denoted by $P_p = (M - I + 1)(N - J + 1)$. This can be seen in Figure 2a, where a white pixel is 1, a black pixel is 0, and the window size is $I \times J = 3 \times 3$.

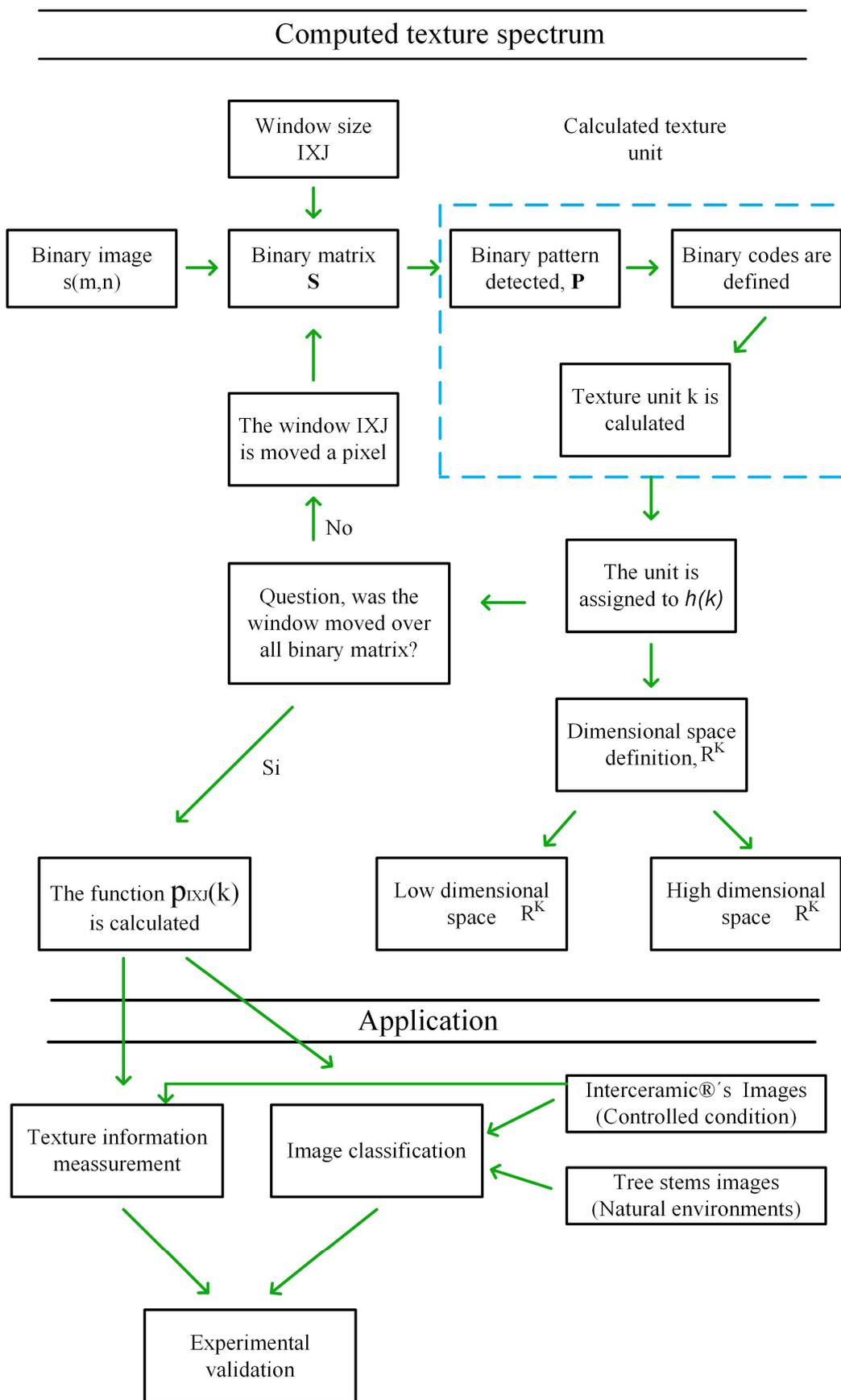


Figure 1. Proposed methodology for calculating the TS_PETU histogram and its application in measuring texture information and image classification: the texture unit is in blue.

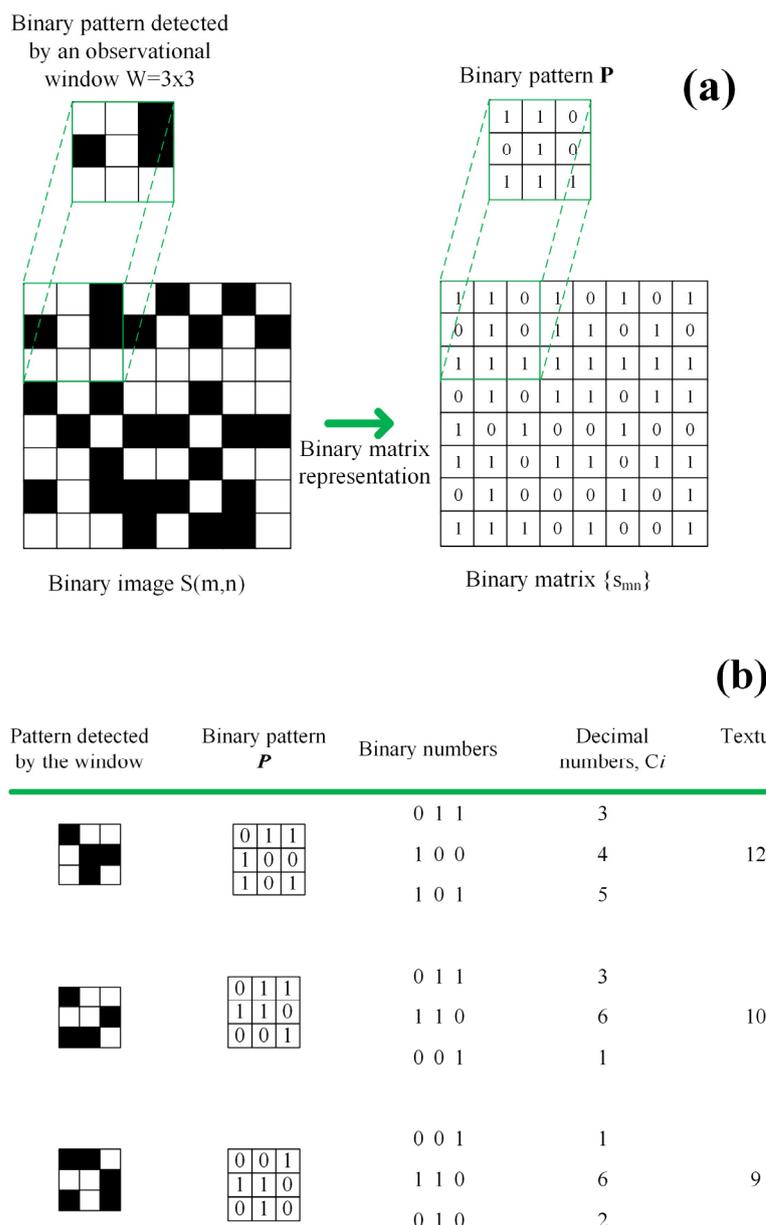


Figure 2. (a) Binary image $s(m, n)$ represented by the binary matrix $\{s_{mn}\}$: $M = 8, N = 8$, and the binary pattern was detected by the observational window $W = 3 \times 3$; (b) three texture unit examples calculated using the parallel codification.

Let P be a binary pattern detected through window W . If each row is considered a binary number, then each binary number can be independently codified as a decimal number through a BCD conversion such that the i th decimal number is calculated by means of

$$c_i = \sum_{j=0}^{J-1} b_j 2^j = b_0 2^0 + b_1 2^1 + b_2 2^2 + \dots + b_{J-1} 2^{J-1} \tag{1}$$

where c_i ($i = 1, 2, \dots, I$) is the i th decimal number, b_j ($j = 0, 1, \dots, J - 1$) is the j th bit number, 2 is the base, and j indicates the j th element. Thus, the texture unit is defined by summing all decimal numbers, which are calculated from the pattern

$$k = \sum_{i=1}^I c_i = c_1 + c_2 + \dots + c_I \tag{2}$$

where k is our texture unit definition calculated through a parallel codification. Figure 2b shows three texture units calculated using the parallel codification $I \times J = 3 \times 3$. Based on Figure 2, the texture unit k is calculated by carrying out the following simple procedure: (a) An observational window is defined and its size is $W = I \times J$; (b) the window W detects a binary pattern for each position on the binary image $s(m, n)$; (c) the binary pattern is interpreted as a binary state P , whose number of rows is I and number of columns is J ; (d) a decimal number is calculated from each row and there are I numbers; and (e) the texture unit k is estimated by calculating the sum of all decimal numbers.

To calculate the minimum value for the texture unit k , all elements are zeros for the pattern $P = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix}$. Then, by applying the procedure described and considering Equation (1), decimal numbers are

$$\begin{aligned} c_1 = 0 &= \sum_{j=0}^{J-1} b_j 2^j = 0 \times 2^0 + 0 \times 2^1 + 0 \times 2^2 + \cdots + 0 \times 2^{J-1} \\ c_2 = 0 &= \sum_{j=0}^{J-1} b_j 2^j = 0 \times 2^0 + 0 \times 2^1 + 0 \times 2^2 + \cdots + 0 \times 2^{J-1} \\ c_3 = 0 &= \sum_{j=0}^{J-1} b_j 2^j = 0 \times 2^0 + 0 \times 2^1 + 0 \times 2^2 + \cdots + 0 \times 2^{J-1} \cdot \\ &\vdots \\ c_I = 0 &= \sum_{j=0}^{J-1} b_j 2^j = 0 \times 2^0 + 0 \times 2^1 + 0 \times 2^2 + \cdots + 0 \times 2^{J-1} \end{aligned} \quad (3)$$

Based on Equations (2) and (3), the minimum value for the texture unit is

$$k = 0 = c_1 + c_2 + c_3 + \cdots + c_I = 0 + 0 + 0 + \cdots + 0 \quad (4)$$

On the other hand, to estimate the maximum value, all element are ones for the pattern

$P = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & 1 & 1 & \cdots & 1 \\ 1 & 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & 1 & \cdots & 1 \end{bmatrix}$. By carrying out the previous procedure and using Equation (1), the decimal numbers are

$$\begin{aligned} c_1 &= 1 \times 2^0 + 1 \times 2^1 + 1 \times 2^2 + \cdots + 1 \times 2^{J-1} \\ c_2 &= 1 \times 2^0 + 1 \times 2^1 + 1 \times 2^2 + \cdots + 1 \times 2^{J-1} \\ c_3 &= 1 \times 2^0 + 1 \times 2^1 + 1 \times 2^2 + \cdots + 1 \times 2^{J-1} \cdot \\ &\vdots \\ c_I &= 1 \times 2^0 + 1 \times 2^1 + 1 \times 2^2 + \cdots + 1 \times 2^{J-1} \end{aligned} \quad (5)$$

From Equation (5), the following is obtained:

$$\begin{aligned} c_1 &= 2^J \\ c_2 &= 2^J \\ c_3 &= 2^J, \\ &\vdots \\ c_I &= 2^J \end{aligned} \quad (6)$$

The texture unit is

$$k = c_1 + c_2 + c_3 + \dots + c_I = 2^J + 2^J + 2^J + \dots + 2^J, \tag{7}$$

and then the maximum value of the texture unit is defined by

$$K = k = I(2^J - 1) + 1 \tag{8}$$

Based on Equations (4) and (8), the texture unit k can take a discrete value into the interval of 0 to $K - 1 = I(2^J - 1)$, where K is the maximum value. This interval considers all possible states for the pattern P .

2.3. Dimensional Space R^K

When an image $s(m, n)$ is transformed in a texture spectrum $h(k)$ where k is the texture unit, m, n are the coordinates for the digital image, and the unit k is used as an index, the spectrum has a length of 0 to $K - 1$ and the discrete histogram $h(k)$ has a dimensional space denoted by R^K . In the R^K space, all possible binary states for the pattern P are considered. There are R^K dimensions, and the maximum dimension is the maximum value of the texture unit definition. Clearly, the dimensional space is a function of the observation window size W , whose size is $I \times J$, and therefore Equation (8) can be written as

$$K(I, J) = I(2^J - 1) + 1 \tag{9}$$

Moving forward in this work, $K(I, J)$ will be used or only K . The behavior is linear for the term I but the behavior is exponential for the term J . Therefore, the behavior of I reduces exponentially in space from $R^{2^{i \times j}}$ [12,14] to $R^{I(2^J-1)+1}$. In Figure 3, the behavior of $R^{I(2^J-1)+1}$ vs. $W = I \times J$ is observable.

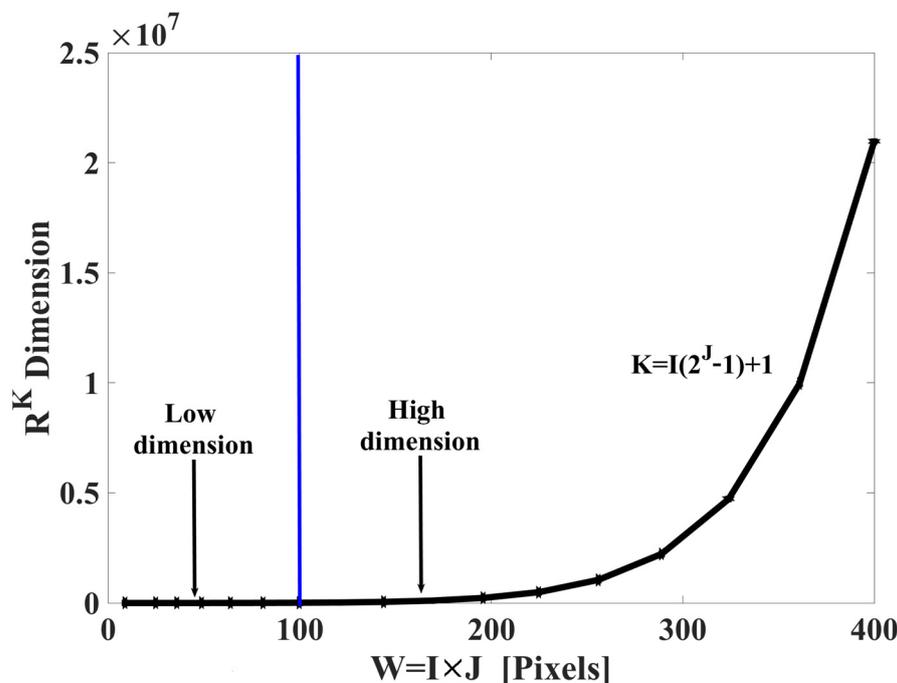


Figure 3. Behavior of R^K vs. $W = I \times J$, where $I = J$.

By analyzing Figure 3, we can define two regions: low-dimensional space and high-dimensional space. The threshold between both regions is indicated by the blue line. In the low-dimensional space region, the observation window must be within the interval

$$3 \times 3 \leq I \times J \leq 10 \times 10 \quad (10)$$

and as a consequence,

$$R^{22} \leq R^K \leq R^{10,231} \quad (11)$$

On the other hand, in the region of high-dimensional space, the observation window satisfies the condition

$$I \times J > 10 \times 10 \quad (12)$$

and then

$$R^K > R^{10,232} \quad (13)$$

is satisfied.

Computationally speaking, the texture spectrum $h(k)$ can operate in the region of low- or high-dimensional space. This offers versatility in its application.

2.4. Algorithm

In texture classification, a digital image $s(m, n)$ is transformed into a discrete histogram $h(k)$, where the texture unit k is used as the index and the length of the discrete histogram must be within the interval of 0 to $K - 1$. The histogram $h(k)$ is interpreted as a texture spectrum and shows the frequency of occurrence of the units calculated from the image $s(m, n)$.

To avoid mathematical complexity, an algorithm to transform the image into a texture spectrum is described below: $s(m, n) \rightarrow h(k)$: (1) the binary image $s(m, n)$ is interpreted as a binary array $\{s_{mn}\}$, where the white pixels are 1 s and the black pixels are 0 s; (2) the observation window size $W = I \times J$ is selected; (3) by moving the observation window pixel by pixel over the entire matrix $\{s_{mn}\}$, a binary pattern P is detected for each position on the image under study; (4) the texture unit k is calculated for each pattern P and then the unit k is used as an index in the discrete histogram $h(k)$ ($k = 0, 1, 2, \dots, K - 1$), whose length is between 0 and $K - 1$; and (5) the histogram $h(k)$ is divided by the total texture units P_p , obtaining the function of probability densities:

$$p_{I \times J}(k) = \frac{h(k)}{P_p} = \frac{h(k)}{(M - I + 1)(N - J + 1)} \quad (14)$$

The function $p_{I \times J}(k)$ is called Texture Spectrum based on the Parallel Encoded Texture Unit (TS_PETU) because the unit k is encoded in parallel, the index $I \times J$ indicates the observation window size, and the equalized histogram TS_PETU shows the frequency of occurrence of the texture units calculated from the digital image $\{s_{mn}\}$. The Algorithm 1: TS_PETU to calculate the TS_PETU histogram is shown below.

By applying the described algorithm and the Algorithm 1: TS_PETU, and using different observation window sizes, the histogram TS_PETU was calculated for a binary digital image; see Figure 4.

Figure 4a shows the binary image $s(m, n)$ of a tree stem, and its size is $M \times N = 4169 \times 3120$ pixels. Figure 4b,c show the texture spectra calculated with observation windows of 5×5 and 6×6 pixels, respectively. Both TS_PETU histograms have low-dimensional space since R^{156} and R^{379} are the respective numbers of dimensions and Conditions (10) and (11) are met. On the other hand, Figure 4d,e show the spectra calculated with windows with 14×14 and 15×15 pixels, respectively. Both TS_PETU histograms have high-dimensional space because the number of dimensions is $R^{229,363}$ and $R^{491,506}$, respectively, and as a consequence, Conditions (12) and (13) are satisfied.

Algorithm 1: TS_PETU

Beginning

Input:

 $s(m, n) \leftarrow$ user I and $J \leftarrow$ user

Calculus:

 M and $N \leftarrow s(m, n)$ for $m : M$ for $m : N$

Texture unit calculation:

 $P \leftarrow s(m, n)$ $c_1, c_2, \dots, c_I \leftarrow P$ $k = c_1 + c_2 + \dots + c_I$ Texture unit mapping k to the discrete histogram $h_{I \times J}(k)$ $h_{I \times J}(k) \leftarrow k$

end

end

Calculation of probability density function $p_{I \times J}(k)$ or histogram TS_PETU $p_{I \times J}(k) \leftarrow h_{I \times J}(k)$

End

Binary image to transform

Observation window size selection

Binary image size

Displacement over image lines

Displacement over image columns

Extraction of binary pattern P from image $s(m, n)$

Calculation of decimal values by conversion BCD

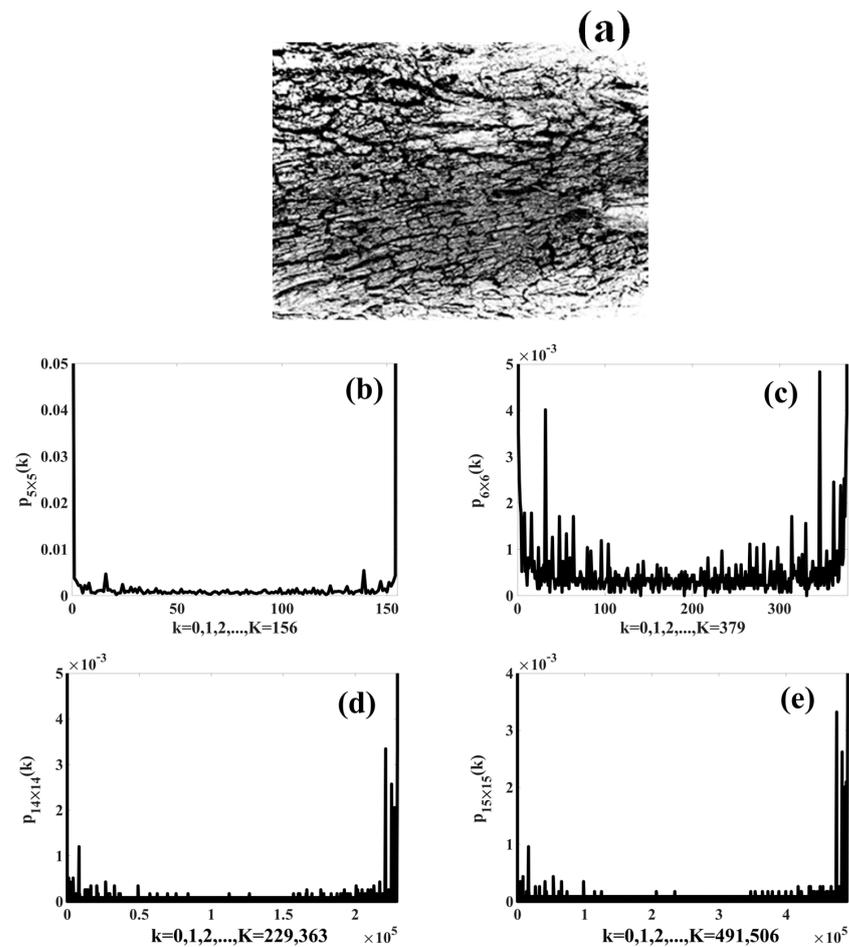
Texture unit calculation k Unit k is assigned to the histogram $h_{I \times J}(k)$ Histogram calculation TS_PETU, $p_{I \times J}(k)$ 

Figure 4. (a) Binary image of tree stem used to calculate its texture spectra $p_{I \times J}(k)$; (b) $p_{5 \times 5}(k)$ calculated with $W = 5 \times 5$; (c) $p_{6 \times 6}(k)$ calculated with $W = 6 \times 6$; (d) $p_{14 \times 14}(k)$ calculated with $W = 14 \times 14$; (e) $p_{15 \times 15}(k)$ calculated with $W = 15 \times 15$.

Based on the results shown in Figure 4, the binary image $s(m, n)$ was transformed into the TS_PETU texture spectrum, which can operate with low- and high-dimensional space. Its region of operation depends on the selected observation window size.

2.5. Texture Information (Entropy)

Knowing that the transformed TS_PETU generates the odds function $p_{I \times J}(k)$, and knowing that in the information theory the amount of information is measured based on a function of probability densities, then by applying the information theory, the amount of texture information extracted from the image can be measured.

Since the texture unit k is a discrete random variable, let us assume that its initial indeterminacy is equal to k_α ($\alpha = 0, 1, 2, \dots, K - 1$), where there are K possible states due to dimensional space R^K , and if we consider that all states are equiprobable, the information provided by the texture unit is [21]

$$I(k_\alpha) = \log_2\left(\frac{1}{k_\alpha}\right) = -\log_2(k_\alpha) \tag{15}$$

such that the information provided by all the texture units is estimated with

$$I(k_0, k_1, k_2, \dots, k_{K-1}) = I(k_0) + I(k_1) + I(k_2) + \dots + I(k_{K-1}) \tag{16}$$

Combining Equations (15) and (16), the texture information is expressed by

$$I(k_0, k_1, k_2, \dots, k_{K-1}) = -\log_2(k_0) - \log_2(k_1) - \log_2(k_2) - \dots - \log_2(k_{K-1}) \tag{17}$$

Since the texture unit has probability $p_k(k)$, the amount of texture information is obtained by means of the weighted sum

$$H = -p_0(k)\log_2(p_0(k)) - p_1(k)\log_2(p_1(k)) - p_2(k)\log_2(p_2(k)) - \dots - p_{K-1}(k)\log_2(p_{K-1}(k)) \tag{18}$$

Finally, the amount of average texture information is the weighted average value of the amount of texture information of the various states of the unit k , and this is determined with Probability Function (14).

$$H = -\sum_{\alpha=0}^{K-1} p_{I \times J}(k)\log_2(p_{I \times J}(k)) = \sum_{\alpha=0}^{K-1} p_{I \times J}(k)\log_2\left(\frac{1}{p_{I \times J}(k)}\right), \tag{19}$$

Note that, in the limit of the summation, we have the parameter K . Considering the dimensional space of the TS_PETU transform (Equation (8)) in Expression (19), the amount of texture information is calculated through

$$H = -\sum_{\alpha=0}^{I(2^J-1)} p_{I \times J}(k)\log_2(p_{I \times J}(k)) \tag{20}$$

That is, the amount of texture information H is a function of the observation window size $w = I \times J$. To illustrate what has been mentioned, let us consider that all texture units are equiprobable, and as a consequence, Expression (20) can be rewritten as follows:

$$H = -\sum_{\alpha=0}^{I(2^J-1)} \frac{1}{I(2^J-1)+1} \log_2\left(\frac{1}{I(2^J-1)+1}\right), \tag{21}$$

whose behavior is shown in Figure 5.

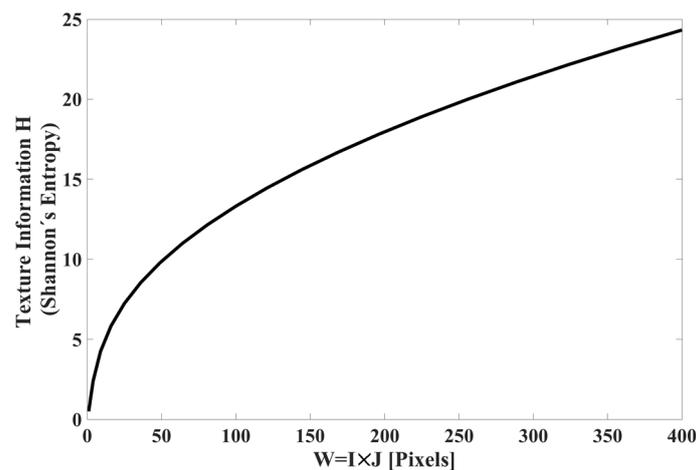


Figure 5. Amount of texture information behavior (Shannon's entropy) H vs. window size $I \times J$, with an interval of $I \times J = 3 \times 3$ to $I \times J = 20 \times 20$.

Observing Figure 5, the y-axis corresponds to the amount of texture information, which can also be identified as the Shannon entropy; the x-axis corresponds to the size of the observation window; and the behavior of H vs. $I \times J$ has exponential growth. We can conclude that the amount of texture information extracted from image S is a function of the window size; the larger the window, the more information is extracted. That is, the TS_PETU transform is more efficient in image classification when the window $W = I \times J$ it is bigger.

2.6. Application

Various classifiers have been reported in the literature to which texture spectra are applied as a multidimensional characteristic vector [13–20]. In particular, in references [12,14], a multi-class classifier based on image statistics is described, applied, and optimized. Due to its experimentally demonstrated efficiency, in this section, the TS-PETU texture spectrum is applied as the characteristic vector. In Figure 6, the classifier for multiple classes is schematically shown.

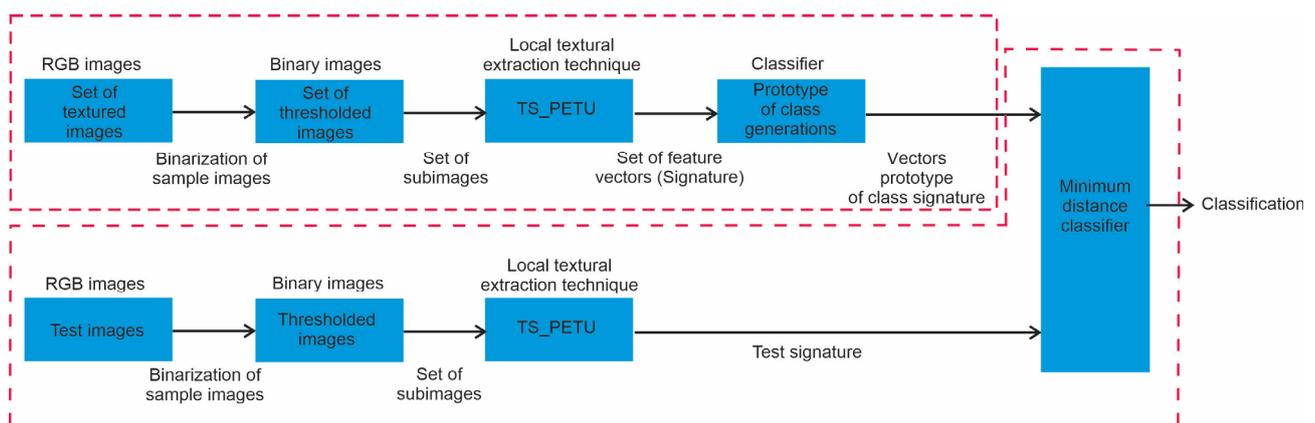


Figure 6. Multi-class classifier based on the image statistic.

As shown in Figure 6, the classifier consists of two stages: learning and recognition. In the learning stage, the digital images are classified by a human expert, each image is considered a class, and in the database there are C classes. To characterize each class based on its local textural characteristics, for each class $S_c(m, n)$ ($c = 1, 2, \dots, C$), a series of subimages is drawn at random $S_{c,s}(m, n)$ ($s = 1, 2, \dots, S$) and then for each subimage $S_{c,s}(m, n)$, the texture spectrum is calculated $p_{I \times J}^{c,s}(k)$. Finally, the characteristic vector of

class c is denoted by $p_{I \times J}^c(k)$ and is determined with the average of the texture spectra $p_{I \times J}^{c,s}(k)$, $p_{I \times J}^c(k) = \frac{1}{S} \sum_{s=1}^S p_{I \times J}^{c,s}(k)$ [12,14]. In the recognition stage, from a test image $S_t(m, n)$, a series of subimages is randomly extracted $S_{t,p}(m, n)$ ($p = 1, 2, \dots, P$). For each subimage $S_{t,p}(m, n)$, the texture spectrum is calculated $p_{I \times J}^{t,p}(k)$, and then the characteristic vector of the test image $p_{I \times J}^t(k)$ is obtained with the average of the texture spectrum $p_{I \times J}^{t,p}(k)$, $p_{I \times J}^t(k) = \frac{1}{P} \sum_{p=1}^P p_{I \times J}^{t,p}(k)$. Finally, the test image $S_t(m, n)$ is classified using the minimum distance between the prototype vector $p_{I \times J}^c(k)$ and the test image vector $p_{I \times J}^t(k)$ [12].

3. Experimental Work

In this experimental work, two digital image databases were used and three series of experiments were developed. One database was provided by Interceramic[®], and the second database was acquired in natural environments. In the first experiment, each image from the Interceramic[®] database was binarized using a global threshold technique. Subsequently, each binary image had its local texture characteristics extracted by means of the TS_PETU transform, and then the texture information was measured. With the results, the H vs. $I \times J$ behavior graph was created. In the second experiment, the digital images from the Interceramic[®] database were classified, the TS_PETU histogram was used as a characteristic vector in the classifier described in Section 2.6, and with the results, the following behavior graphs were constructed: classification efficiency ($Ef_{I \times J}$) vs. window size ($W = I \times J$), sorting efficiency ($Ef_{I \times J}$) vs. dimensional space (R^K), and sorting efficiency ($Ef_{I \times J}$) vs. texture information (H). In the third series of experiments, the TS_PETU histogram was used again as a feature vector in the classifier described in Section 2.6, and then a database of natural images of tree stems was classified. With the results of this series of experiments, the following behavior graphs were again generated: classification efficiency ($Ef_{I \times J}$) vs. window size ($W = I \times J$), sorting efficiency ($Ef_{I \times J}$) vs. dimensional space (R^K), and sorting efficiency ($Ef_{I \times J}$) vs. texture information (H). Finally, the numerical experiments were implemented by applying the MatLab 2016b software and a GHIA computer with the following characteristics: Intel(R) Core (TM) i7-4790 CPU 3.60 GHz processor and 8GB RAM.

3.1. Experiment 1: Measurement of Texture Information Based on TS_PETU

As a first step, the Interceramic[®] image database was taken and the digital images were binarized by applying a global threshold technique. The RGB images are observable in Figure 7, and each image has a size of $M \times N = 300 \times 300$. Using a window size within the interval of $I \times J = 3 \times 3$ to $I \times J = 20 \times 20$, local texture features were extracted from the binary image, and then the amount of texture information H in the histogram $p_{I \times J}(k)$ was calculated. Next, using the measurements of H and the window size, the behavior graph of H vs. $W = I \times J$ was created (see Figure 8).

As can be observed in Figure 8, the texture information H grows when the observation window size $W = I \times J$ is bigger. The behavior graph of H vs. $I \times J$ has an exponential form, and its behavior is similar for all images. The minimum texture information value corresponds to “Calabria Ambroto Gray” and its value is $H_{3 \times 3} = 2.86$. The maximum texture information value corresponds to “Dover Rochester Grey Mate” and the value is $H_{20 \times 20} = 16.26$. Then, by comparing Figure 5 (theoretical results) with Figure 8 (experimental results), it can be seen that the theory and the experiments are in accordance. We can then infer that the image classification efficiency with the TS_PETU transform is better when the observation window is larger. This is corroborated with two series of image classification experiments, whose results are shown in the following sections.



Figure 7. Ceramic tiles produced by Interceramic[®].

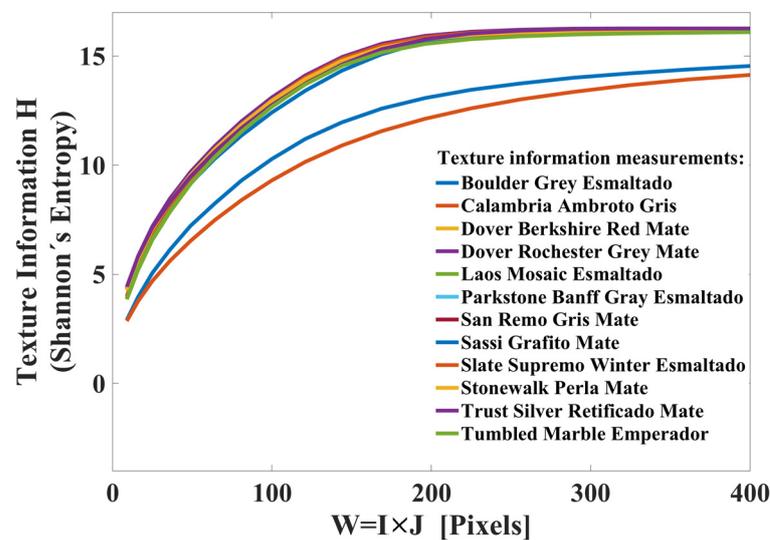


Figure 8. Experimental texture information measured from Interceramic[®]'s database.

During the computing process, the execution time and the number of operations were measured in the calculation of the histogram $p_{I \times J}(k)$. Table 1 was generated with the measurements. By analyzing Table 1, it can be seen that the execution time increases due to the increase in the size of the observation window. The increase in time is attributed to the fact that the number of computational operations also grows with the window size. In our experiments, the minimum execution time measured was 0.3770 s for window $I \times J = 3 \times 3$ and the number of computational operations was 1,509,668. On the other hand, the maximum execution time was 0.6390 s, the window size was $I \times J = 20 \times 20$ pixels, and the number of operations was 63,089,839.

Table 1. Average execution time and number of operations required to calculate the histogram $p_{I \times J}(k)$ for tile images from Interceramic[®], $M \times N = 300 \times 300$ pixels.

Window Size $I \times J$ Pixels	Number of Operations	Runtime, Seconds
3 × 3	1,509,668	0.3770
4 × 4	2,734,479	0.3770
5 × 5	4,293,184	0.3770
6 × 6	6,178,775	0.3770
7 × 7	8,384,292	0.3770
8 × 8	10,902,823	0.3780
9 × 9	13,727,504	0.3800
10 × 10	16,851,519	0.3810
11 × 11	20,268,100	0.3860
12 × 12	23,970,527	0.3880
13 × 13	27,952,128	0.3890
14 × 14	32,206,279	0.3900
15 × 15	36,726,404	0.3980
16 × 16	41,505,975	0.4060
17 × 17	46,538,512	0.4310
18 × 18	51,817,583	0.4370
19 × 19	57,336,804	0.4670
20 × 20	63,089,839	0.6390

3.2. Experiment 2: Classification of Database Images from Interceramic[®]

In this section, the digital image database provided by Interceramic[®] was classified. The tiles were manufactured in their industrial plant and were analyzed and named by a human expert (see Figure 7). In addition, the ceramic tiles were photographed at Interceramic[®] under controlled lighting conditions, rotation, and scale. The size of the images is 300×300 pixels. On the other hand, by calculating the TS_PETU histogram with window sizes within the interval of $I \times J = 3 \times 3$ to $I \times J = 20 \times 20$ and using the TS_PETU histogram as a characteristic vector in the multi-class classifier (see Section 2.6), the Interceramic[®] database was identified. In the classifier, the images used in the learning stage were the same images as those used in the recognition stage. In addition, the subimage size was similar for both stages. In Table 2, the parameters for the classifier are shown.

Table 2. Parameters used in the classifier for multiple classes.

Learning Stage		Recognition Stage	
Number of classes, C	$C = 12$	Test images number, T	$T = 12$
Image size, $S_c(m, n)$	$M \times N = 300 \times 300$	Test image size, $S_t(m, n)$	$M \times N = 300 \times 300$
Number of subimages, $S_{c,s}(m, n)$	$S = 100$	Number of subimages, $S_{t,p}(m, n)$	$P = 100$
Subimage size, $S_{c,s}(m, n)$	150×150	Subimage size, $S_{t,p}(m, n)$	150×150

The results obtained in the classifier are expressed through a confusion matrix M_c , the main diagonal of which is the correct identification [12], the elements outside the main diagonal are classification errors, the rows correspond to the test images, and the columns correspond to the prototype images (master images). Table 3 shows an example of confusion matrix M_c , which was obtained in our experimental work. In this example, the TS_PETU histogram was calculated using a $I \times J = 5 \times 5$ pixel observation window.

Table 3. Confusion matrix M_c obtained using a 5×5 window size.

		Master Images											
		Boulder Grey	Calabria Ambroto	Dover Bershire	Dover Rochester	Laos Mosaic	Parkstone Banff	Sam Remo	Sassi Grafito	Slate Supremo	Stonewalk Perla	Trust Silver	Tumber Marble
Test Images	Boulder Grey	1	0	0	0	0	0	0	0	0	0	0	0
	Calabria Ambroto	0	1	0	0	0	0	0	0	0	0	0	0
	Dover Bershire	0	0	1	0	0	0	0	0	0	0	0	0
	Dover Rochester	0	0	0	1	0	0	0	0	0	0	0	0
	Laos Mosaic	0	0	0	0	1	0	0	0	0	0	0	0
	Parkstone Banff	0	0	0	0	0	1	0	0	0	0	0	0
	Sam Remo	0	0	0	0	0	0	1	0	0	0	0	0
	Sassi Grafito	0	0	0	0	0	0	0	1	0	0	0	0
	Slate Supremo	0	0	0	0	0	0	0	0	1	0	0	0
	Stonewalk Perla	0	0	0	0	0	0	0	0	0	1	0	0
	Trust Silver	0	0	0	0	0	0	0	0	0	0	1	0
	Tumber Marble	0	0	0	0	0	0	0	0	0	0	0	1
Classification efficiency												100%	

From the confusion matrix of the example (Table 3), the classification efficiency for the texture spectrum $p_{5 \times 5}(k)$ is calculated by

$$Ef_{I \times J} = \frac{\sum \text{diag}(M_c)}{C} \times 100, \quad (22)$$

where $Ef_{I \times J}$ is the classification efficiency in terms of percentage, the term $I \times J$ is the observation window size, $\sum \text{diag}(M_c)$ is the sum of all of the elements of the main diagonal (marked in blue) in the matrix M_c , and C is the total number of classes. Considering the values of the confusion matrix (Table 3) and the parameters of Table 2, the efficiency $Ef_{5 \times 5}$ is

$$Ef_{5 \times 5} = \frac{1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1}{12} \times 100 = 100\% \quad (23)$$

Based on Equation (23), the classification efficiency of the texture spectrum $p_{5 \times 5}(k)$ is equal to $Ef_{5 \times 5} = 100\%$. The experimental results are shown in Figure 9.

In Figure 9, the classification efficiency for the ceramic tile images is 100%. This high efficiency can be attributed to the following points:

1. The images of the ceramic tiles were acquired under controlled conditions, such as: lighting, scale, rotation, and translation. This increased the possibility of success in the identification of images, and as a consequence, also reduced possible classification errors.
2. The TS_PETU texture extraction technique correctly characterized the digital image through its local texture characteristics—that is, the TS_PETU transform extracted sufficient texture information to achieve high image classification efficiency.
3. The statistical classifier for multiple classes was optimized to achieve high image identification efficiency. Optimization was achieved based on the size of the subimages and the number of subimages S and P .

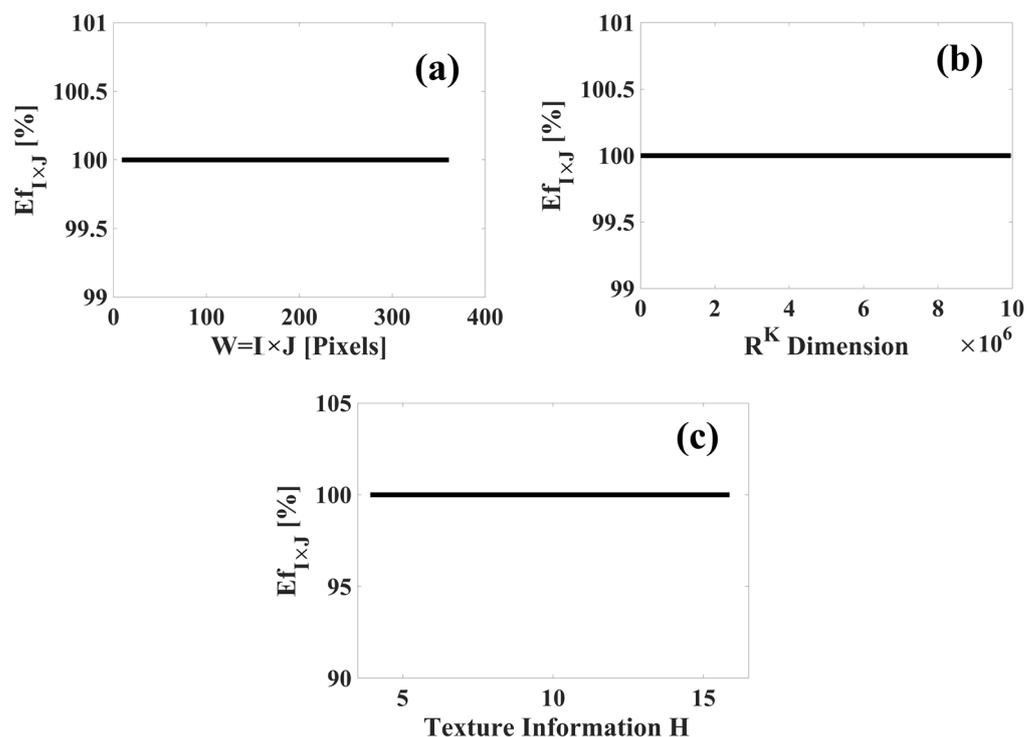


Figure 9. Experimental results obtained in the classification of images of ceramic tiles: (a) classification efficiency ($Ef_{I \times J}$) vs. window size ($W = I \times J$); (b) classification efficiency ($Ef_{I \times J}$) vs. dimensional space (R^K); (c) classification efficiency ($Ef_{I \times J}$) vs. texture information (H).

The TS_PETU transform must be calculated with a window size within the interval of $I \times J = 3 \times 3$ to $I \times J = 19 \times 19$, since if the window has a size of $I \times J = 20 \times 20$ or bigger, there is an overflow of physical memories. However, with these results, it is confirmed that the TS_PETU transformed has potential application in digital images and surface quality control and can be operated in low- or high-dimensional space.

3.3. Experiment 3: Classification of Natural Images

In this section, using the TS_PETU histogram as a characteristic vector in the classifier described in Section 2.6 and with window sizes of $I \times J = 3 \times 3$ to $I \times J = 10 \times 10$, a database of 64 natural images was classified. The images were of trees, they were RGB, they were acquired using an LG-Q50 camera, and the rotation, scale, and translation were controlled but the lighting was natural. The size of each image is 3120×4260 pixels and the images can be observed in Figure 10.

On the other hand, the characteristics of the classifier were based on the data in Table 4, where the number of classes is $C = 68$, the number of subimages is $S = 100$, and the subimage size is 1560×2130 .

Again, the results obtained in the classification of natural images are expressed through the confusion matrix M_c . As previously mentioned, the main diagonal is the correct identification, the elements outside of the main diagonal are classification errors, the rows correspond to the test images, and the columns correspond to the master images. Finally, the classification efficiency $Ef_{I \times J}$ was calculated with Equation (22). The experimental results are observable in Figure 11, where the behavior graphs of $Ef_{I \times J}$ vs. $W = I \times J$, $Ef_{I \times J}$ vs. R^K , and $Ef_{I \times J}$ vs. H are shown.



Figure 10. Natural image database used in the numerical experiments.

Table 4. Parameters used in the classifier for multiple classes.

Learning Stage		Recognition Stage	
Number of classes, C	$C = 68$	Number of test images, T	$T = 68$
Image size, $S_c(m, n)$	$M \times N = 3120 \times 4260$	Test image size, $S_t(m, n)$	$M \times N = 3120 \times 4260$
Number of subimages, S	$S = 100$	Number of subimages, $S_{t,p}(m, n)$	$S = 100$
Subimage size, $S_{c,s}(m, n)$	1560×2130	Subimage size, $S_{t,p}(m, n)$	1560×2130

As can be observed in Figure 11, the efficiency $Ef_{I \times J}$ increases with window size growth W , dimensional space R^K , and texture information H . According to these results, the spectrum TS_PETU has an efficiency of $Ef_{I \times J} = 100\%$ when $W = I \times J = 5 \times 5$ or bigger, there is a dimension of R^{156} or more, and the texture information is $H = 6.25$ or more. This high efficiency can be attributed to points 2 and 3, which were mentioned in Section 3.2. On the other hand, the lowest efficiency was measured in $Ef_{I \times J} = 84.84\%$, where $W = I \times J = 3 \times 3$, there is a dimension of R^{22} , and the texture information is $H = 3.92$. The classification error is attributed to the lighting conditions not being controlled during the image acquisition process. However, from these results, the following can be concluded: The TS_PETU transform has high efficiency in image recognition if it is carried out in a region of low-dimensional space, (see Equations (10)–(13) and Figure 3). In addition, our proposal has potential application in the recognition of images in natural environments since its efficiency is very high when the window W is of a larger size.

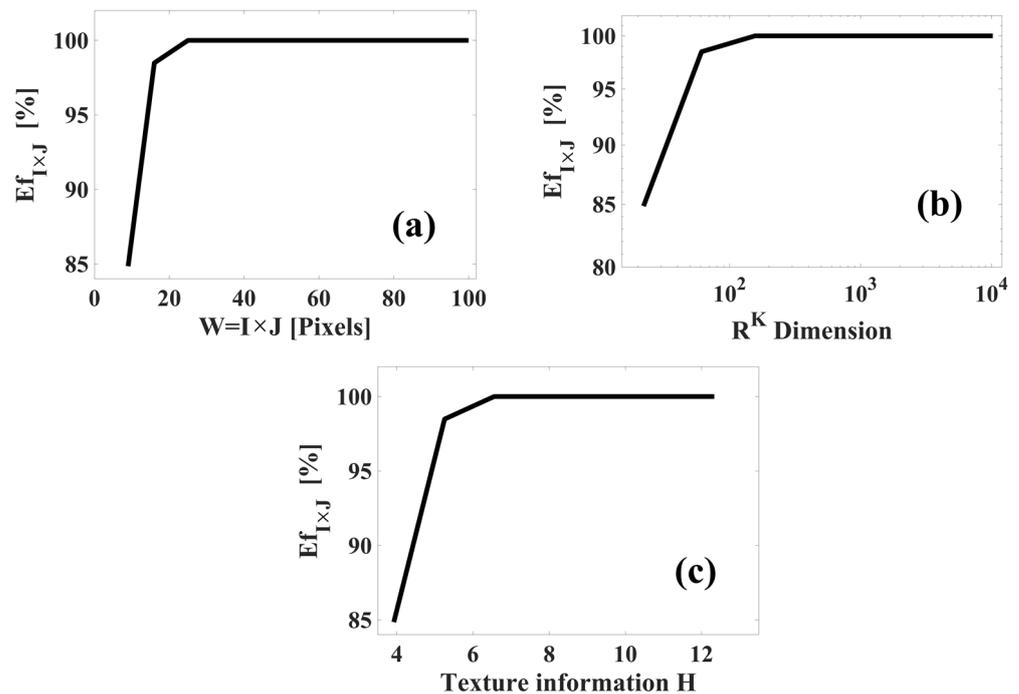


Figure 11. Experimental results obtained in the classification of natural images of trees: (a) $Ef_{I \times J}$ vs. $W = I \times J$; (b) $Ef_{I \times J}$ vs. R^K dimension; (c) $Ef_{I \times J}$ vs. texture information (H).

4. Discussion

In this work, a new texture extraction technique is proposed, which is called Texture Spectrum based on the Parallel Encoded Texture Unit (TS_PETU) because the texture unit is calculated based on a parallel coding. The TS_PETU technique transforms a binary image into a probability density function (equalized histogram) in terms of texture units. The equalized histogram can be calculated using windows greater than 3×3 and the histogram is in dimensional space. Its efficiency in the extraction of texture information and image classification was verified by conducting three series of experiments: In the first series, it is corroborated that the amount of texture information depends on the window size W ; in the second series, its efficiency is confirmed when the images are acquired under controlled conditions; and in the third series, its efficiency is verified when the images are acquired under non-controlled lighting conditions. Based on the results obtained, the following points can be inferred:

1. Texture Spectrum based on the Parallel Encoded Texture Unit (TS_PETU) represents a binary image $s(m, n)$ as a probability density function in terms of texture units $p_{I \times J}(k)$, whose characteristics are low-dimensional space and high efficiency of image classification.
2. The TS_PETU histogram shows the frequency of occurrence of the texture units calculated from the binary image under study.
3. Because the texture unit is calculated by applying parallel coding concepts, the TS_PETU histogram has low-dimensional space and it is possible to use large windows.
4. The amount of texture information contained in the TS_PETU histogram is based on the observation window size $W = I \times J$.
5. It is experimentally corroborated that the TS_PETU transform has high efficiency in image classification.
6. Classification efficiency is improved using windows W with larger sizes; see Sections 3.1–3.3.
7. The TS_PETU histogram can work in low- and high-dimensional space regions, and in both regions there can be high image classification efficiency.

8. The efficiency of the TS_PETU transform increases when the conditions are controlled during the image acquisition process.
9. The TS_PETU histogram can be calculated using parallel compute.
10. The classification efficiency with the TS_PETU transform can be reduced due to noise produced by the illumination source and the electronic systems used during image acquisition and processing due to numerical computational errors and retroreflections generated on the surface of the material under study [22].
11. The TS_PETU transform has significant practical application, and some benefits for the user are high efficiency, short execution times, low-dimensional space, selectivity through the observation window, implementation with parallel computing, and easy implementation with electronic cards.

Comparing TS_PETU with the techniques reported in reference [11], our proposal can be estimated with observation windows greater than 3×3 and its low-dimensional space can be conserved, the amount of texture information increases with the window size, and as a consequence, the classification efficiency is improved. Now, comparing the TS_PETU with the original versions of the LBP and CCR transforms, which are based on BCD coding, our proposal offers some advantages, as can be seen in Table 5.

Table 5. Comparative table between TS_PETU and the original versions of the CCR and LBP.

Interceramic®Image Database					
TS_PETU		CCR		LBP	
Window Size	Efficiency (%)	Window Size	Efficiency (%)	Window Size	Efficiency (%)
3×3	100	3×3	94.23	3×3	100
4×4	100	4×4	99.23	4×4	Not applicable [23,24]
5×5	100	5×5	Overflow	5×5	Overflow
6×6	100	6×6		6×6	
7×7	100	7×7		7×7	
8×8	100	8×8		8×8	
9×9	100	9×9		9×9	
10×10	100	10×10		10×10	
11×11	100	11×11		11×11	
12×12	100	12×12		12×12	
13×13	100	13×13		13×13	
14×14	100	14×14		14×14	
15×15	100	15×15		15×15	
16×16	100	16×16		16×16	
17×17	100	17×17		17×17	
18×18	100	18×18		18×18	
19×19	100	19×19		19×19	
20×20	Overflow	20×20		20×20	
Natural image database					
TS_PETU		CCR		LBP	
Window size	Efficiency (%)	Window size	Efficiency (%)	Window size	Efficiency (%)
3×3	84.848	3×3	78.083	3×3	98.40
4×4	98.484	4×4	87.030	4×4	Not applicable [23,24]
5×5	100	5×5	Overflow	5×5	Overflow
6×6	100	6×6		6×6	
7×7	100	7×7		7×7	
8×8	100	8×8		8×8	
9×9	100	9×9		9×9	
10×10	100	10×10		10×10	

When the images are acquired under controlled conditions, the three transforms have high efficiency greater than 90%. However, in our experiments, when the images

are acquired in natural environments, the CCR transform has an efficiency of less than 80%, the LBP has an efficiency of 98.40, and the TS_PETU has an efficiency of up to 100 if the observation window is equal to or greater than $I \times J = 5 \times 5$. Two important points to highlight are that (1) the CCR and LBP cannot work with windows larger than $I \times J = 5 \times 5$ because the dimensional space is very large and an overflow is generated in the computer, whereas the TS_PETU can operate with windows within the interval of $I \times J = 3 \times 3$ to $I \times J = 19 \times 19$ and the computational overflow is generated for the window $I \times J = 20 \times 20$, and (2) the decrease in efficiency is attributed to the fact that the conditions were not controlled during the acquisition of the images.

Due to the definition of the texture unit and the efficiency obtained experimentally, our future research lines are (1) to apply parallel compute to reduce execution times, (2) to develop artificial vision applications, (3) to describe the mathematical foundation of the TS_PETU transform, and (4) to optimize image classification with the TS_PETU transform and perform a sensitivity analysis.

5. Conclusions

In this work, a new texture extraction technique is proposed and applied that is called Texture Spectrum based on the Parallel Encoded Texture Unit (TS_PETU) because the unit k is calculated based on parallel coding. The TS_PETU transformation is based on a local analysis of the binary image $s(m, n)$, and this represents the image as a probability density function $p_{I \times J}(k)$ in terms of texture units. The main characteristics of the function $p_{I \times J}(k)$ are: (1) It can be calculated with windows greater than $W = I \times J = 3 \times 3$, (2) it has very low dimensional space $R^{I(2^J-1)+1}$, (3) the amount of texture information H is based on the window size $I \times J$, and (4) it has high efficiency in image classification since it can reach up to 100%. Experimentally, the four characteristics were verified, confirming that the TS_PETU transform has high efficiency in image recognition.

The TS_PETU transform can be implemented in real time due to the parallel coding of the texture unit. In addition, it has potential industrial application when the surface or texture characteristics are important.

Author Contributions: J.T.G.B., H.G.B., A.G.B. and M.E.S.M. proposed the method and analysis; H.G.B., N.E.F.R., V.M.R.B. and O.B.A. generated the digital image database and conducted the formal analysis; J.T.G.B., A.G.B., N.E.F.R. and M.J.R. developed the numerical experiment; J.T.G.B., M.J.R., O.B.A. and M.E.S.M. carried out the results analysis. All authors wrote the article. All authors have read and agreed to the published version of the manuscript.

Funding: This work has no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data related to the results that support our conclusions are available upon request to the authors. It can be done via e-mail. We will be pleased to respond.

Acknowledgments: The authors thank Mexico's National Council of Science and Technology (CONACyT) and Guadalajara University for the support granted. This investigation was carried out following the lines of research "Nanostructured Semiconductors Oxides" by the academic group UDG-CA-895 and "Nano-structured Semiconductors" by C.U.C.E.I., Guadalajara University.

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

References

1. Rui Henriques, S.; Madeira, C. FleBiC: Learning classifiers from high-dimensional biomedical data using discriminative biclusters with non-constant patterns. *Pattern Recognit.* **2021**, *115*, 107900. [[CrossRef](#)]
2. Takahiro, N.; Osumu, O.; Shujiro, O. Deep learning classification of urinary sediment crystals with optimal parameter tuning. *Sci. Rep.* **2022**, *12*, 21178.
3. Hang, S.; Cheng, L. A new cast shadow detection method for traffic surveillance video analysis using color and statistical modeling. *Image Vis. Comput.* **2020**, *94*, 103863.

4. Leone, A.; Distanto, C. Shadow detection for moving object based on texture analysis. *Pattern Recognit.* **2007**, *40*, 1222–1233. [[CrossRef](#)]
5. Steve Tsham, M.A.; Lameiras Koerich, A. A novel bio-inspired texture descriptor based on biodiversity and taxonomic measures. *Pattern Recognit.* **2022**, *123*, 108382.
6. Duen-Pang, K.; Po-Chin, K.; Yung-Chieh, C.; Yu-Chieh, J.K.; Ching-Yen, L.; Hsiao-Wen, C.; Cheng-Yu, C. Machine learning-based segmentation of ischemic penumbra by using diffuser tensor metrics in a rat model. *J. Biomed. Sci.* **2020**, *27*, 80.
7. Muhammad, S.; Siraj, K.; Zahoor, J.; Khan, M.; Hyeonjoon, M.; Jin Tae, K.; Seungmin, R.; Shung Wook, B.; Irfan, M. Leukocytes classification and segmentation in Microscopic blood smear: A resolution-aware healthcare service in smart cities. *IEEE Access* **2016**, *5*, 3475–3489.
8. Benjamin, P.; Ariane, M.; Bodo, B. Image Texture as Quality Indicator for Optical DEM Generation: Geomorphic Applications in the Arid Central Andes. *Remote Sens.* **2023**, *15*, 85. [[CrossRef](#)]
9. Marziye, G.; Hooman, L.; Mehdi, P. A novel method for detecting and delineating coppice trees in UAV images to monitor tree decline. *Remote Sens.* **2022**, *14*, 5910.
10. Turcercyan, M.; Jain, A.K. Texture analysis. In *The Handbook of Pattern Recognition and Computer Vision*, 2nd ed.; Chen, C.H., Pau, L.F., Wang, P.S.P., Eds.; World Scientific Publishing Co.: Singapore, 1998; pp. 207–248.
11. Fernández, A.; Álvarez, M.X.; Bianconi, F. Texture description through histograms of equivalent patterns. *J. Math. Imaging Vis.* **2013**, *45*, 76–102. [[CrossRef](#)]
12. Sánchez Yáñez, R.; Kurmyshev, E.K.; Cuevas, F.J. A framework for texture classification using the coordinated clusters representation. *Pattern Recognit. Lett.* **2003**, *24*, 21–31. [[CrossRef](#)]
13. Samuel, B.; Harald, F.; Eugen, R.; Johannes, T.; Doris, E. Enhancing Classification in correlate microscopy using multiple classifier systems with dynamic selection. *Ultramicroscopy* **2022**, *24*, 113567.
14. Guillen Bonilla, J.T.; Kurmyshev, E.; Fernandez, A. Quantifying a similarity of classes of texture image. *Appl. Opt.* **2007**, *46*, 5562–5570. [[CrossRef](#)] [[PubMed](#)]
15. Minkyung, K.; Junsik, K.; Jongmin, Y.; Jun Kyun, C. Active anomaly detection based on Deep one-class classification. *Pattern Recognit. Lett.* **2023**, *167*, 18–24.
16. Thanh Tuan, N.; Thanh, P.; Nguyen, N.; Thirion, M. Location robust patterns based on invariance of LTP-based features. *Pattern Recognit. Lett.* **2023**, *165*, 9–16.
17. Rossouw van der Merwe, J.; Contreras Franco, D.; Hansen, J.; Tobias, B.; Feigi, T.; Ott, F.; Jdidi, D.; Rugamer, A.; Feiber, W. Low-cost COTS GNSS interference monitoring, detection and classification systems. *Sensors* **2023**, *23*, 3452. [[CrossRef](#)] [[PubMed](#)]
18. Chang, C.W.; Chang, C.Y.; Lin, Y.Y.; Su, W.W.; Chen, H.S.L. A glaucoma detection system based on generative adversarial network and incremental learning. *Appl. Sci.* **2023**, *13*, 2195. [[CrossRef](#)]
19. Anil Kumar, B.; Bansal, M. Face mask detection on photo and real-time video images using coffee-MobileNetV2 transfer learning. *Appl. Sci.* **2023**, *13*, 935. [[CrossRef](#)]
20. Abd Elaziz, M.; Dahou, A.; Mabrou, A.; Ali Ibrahim, R.; Ahmad, O. Aseeri, Medical Image classification for 6G IoT-Enabled smart Health systems. *Diagnostics* **2023**, *13*, 834. [[CrossRef](#)]
21. Richard, W. *Hamming, Coding and Information Theory*, 3rd ed.; Prentice Hall: Hoboken, NJ, USA, 1986.
22. Bianconi, F.; Bello, R.; Fernández, A.; González, E. On comparing colour spaces from a performance perspective: Application to automatic classification of polished natural stones. In *New Trends in image Analysis and Processing—ICIAP 2015 Workshops, Geneo, Italy, 7–8 September 2015*; The Series Lecture Notes in Computer Science; Springer International Publishing: Berlin/Heidelberg, Germany, 2015; Volume 9281, pp. 71–78.
23. Kurmyshev, E.V. Is the Coordinated Clusters Representation an analog of the Local Binary Pattern? *Comput. Syst.* **2010**, *14*, 54–62.
24. Ojala, T.; Pietikainen, M.; Harwood, D. A comparative study of texture measures with classification based on feature distribution. *Pattern Recognit.* **1996**, *29*, 51–59. [[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.