

Article

Vectorial Image Representation on the Texture Space (VIR-TS) Applied to RGB Image Classification: Face Recognition

Héctor Guillen-Bonilla ¹, José Trinidad Guillen-Bonilla ^{2,*}, Maricela Jiménez-Rodríguez ³, Alex Guillen-Bonilla ⁴, Jorge Aguilar-Santiago ⁵, Lucía Ivonne Juárez-Amador ¹ and Antonio Casillas Zamora ²

- ¹ Departamento de Ingeniería de Proyectos, Centro Universitario de Ciencias Exactas e Ingenierías, Universidad de Guadalajara, Blvd. M. García Barragán 1421, Guadalajara 44410, Jalisco, Mexico; hector.guillen1775@academicos.udg.mx (H.G.-B.); lucia.juarez@academicos.udg.mx (L.I.J.-A.)
- ² Departamento de Electro-Fotónica, Centro Universitario de Ciencias Exactas e Ingenierías, Universidad de Guadalajara, Blvd. M. García Barragán 1421, Guadalajara 44410, Jalisco, Mexico; antonio.czamora@academicos.udg.mx
- ³ Departamento de Ciencias Básicas, Centro Universitario de la Ciénega, Universidad de Guadalajara, Av. Universidad No. 1115, Lindavista, Ocotlán 47810, Jalisco, Mexico; maricela.jrodriguez@academicos.udg.mx
- ⁴ Departamento de Ciencias Computacionales e Ingenierías, Centro Universitario de los Valles, Universidad de Guadalajara, Carretera Guadalajara-Ameca Km. 45.5, Ameca 46600, Jalisco, Mexico; alex.guillen@academicos.udg.mx
- ⁵ Departamento de Ciencias Tecnológicas, Centro Universitario de la Ciénega, Universidad de Guadalajara, Av. Universidad No. 1115, Lindavista, Ocotlán 47810, Jalisco, Mexico; jorge.asantiago@academicos.udg.mx
- * Correspondence: trinidad.guillen@academicos.udg.mx

Abstract

In this paper, an RGB image with S is separated by its channels, obtaining an image in each color channel S_R , S_G and S_B . The Vectorial Image Representation on the Texture Space (VIR-TS) transform is calculated for each channel; ergo, each image is represented with a given vector, $S_R \rightarrow \vec{C}_R$, $S_G \rightarrow \vec{C}_G$ and $S_B \rightarrow \vec{C}_B$. Employing the $\vec{C}_R$, $\vec{C}_G$, and $\vec{C}_B$ vectors in a multi-class classifier, a database of RGB images was identified with the aim of verifying the classification efficiency of the VIR-TS transform. Based on the experimental results, the VIR-TS technique presents high efficiency when the noise is not added to the class and when the signal-to-noise ratio is high. For both instances, the efficiency presented is 100%. Nonetheless, if the class noise is high, the efficiency diminishes from 100%, to 95%, to 90% until it decreases to 10%. Based on the results obtained, the VIR-TS transform can be efficiently applied for the development of security systems and access control and can also be implemented in computer vision systems for medical diagnosis, drones, etc.

Keywords: RGB image; Vectorial Image Representation on the Texture Space (VIR-TS) transform; multi-class classifier; texture space; classification efficiency

Academic Editor: Nikolaos Mitianoudis

Received: 23 April 2026

Revised: 20 June 2026

Accepted: 29 June 2026

Published: 2 July 2026

Copyright: © 2026 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 \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Computer vision is frequently employed to identify points of interest within images, applied in facial recognition, medical areas or in the classification of any type of object. To perform these tasks, it is necessary to determine the texture characteristics of images with regions of interest. This task is commonly performed with texture descriptors that work closely with statistical classifiers or neural networks, and these are in charge of identifying the class to which the region of interest being studied belongs. Texture descriptors and

statistical classifiers or neural networks have been deployed in previous papers to cease identity theft by applying facial recognition [1–3]. The biometric identification of people is performed by facial recognition through a multilevel hierarchy representation that applies a set of local classifiers to make decisions [4]. One of the most commonly used descriptors is the Local Binary Pattern (LBP) [5–7], which works with binary patterns; it is applied by dividing the data in three arrays, where two remain fixed, to carry out an in-depth analysis with the intention of optimizing them considering the decomposition of singular values [8]. A study was performed to compare the characteristics of 10 texture descriptors employed for facial and palmprint recognition [9]. A new method to detect facial expressions was presented where Local Binary Patterns (LBP), Local Ternary Patterns (LTP) and Completed Local Binary Patterns (CLBP) were deployed with a Convolutional Neural Network [10]. An optimized Histogram of Oriented Gradients (HOG) algorithm is applied to extract facial characteristics along with a Tree-structured Parzen Estimator (TPE)-based Bayesian optimization technique [11]. An MB-ZZLBP descriptor was deployed for face analysis to perform facial recognition with SVM and NN [12]. Multi-Resolution Multi-Threshold Local Binary Patterns (MRMT-LBPs) were applied to emphasize shades of gray, and the authors of that study proposed systematically generating multiple LBP representations to perform facial recognition [13]. Even employing excellent descriptors and texture classifiers, there exist great challenges that can affect the performance of facial recognition methods, such as lighting variance in the photographs, pose and facial expressions, occlusion or data loss; additionally, there could be disturbances or failures in facial recognition techniques when the images present noise. To mitigate these problems, modifications to the LBP method were proposed to improve the virtual electrical binary pattern descriptor against noise [14], and Support Vector Machine (SVM) was applied for classification. Furthermore, a multi-level adaptative mining model was applied for filtering and a pyramidal block was used to capture the inconsistent characteristics provoked by regional noise [15]. To perform facial expression recognition, a hybrid framework was proposed applying MobileNetV2 deep learning and LBP and HOG descriptors [16]. Facial recognition is employed to determine the drowsiness state of drivers using the HOG descriptor and SVM classifier [17]. Variations in the LBP texture descriptor were performed to generate a Multiscale Discriminative Binary Pattern (MDBP), which was deployed for facial recognition; the central pixel is replaced by a threshold value generated with the mean and the median [18]. A paper presented an investigation of how noise can propagate from the source domain to the target domains where it cannot be detected, causing a decrement in performance [19]; noise alters image quality, which in turn influences facial recognition [20,21]. Without a doubt, any degradation present in facial recognition techniques when images present noise is a subject of great interest.

In reference [22], the Vectorial Image Representation on the Texture Space (VIR-TS) transform was described and applied. The VIR-TS transform extracts the local texture characteristics of a grayscale image, and these are represented in the texture space as vectors on which the direction and magnitude depend on the image under study. The vector was employed in a multi-class classifier where each class belonged to a tree trunk bearing a very high classification efficiency, reaching up to 100% when images had no additional noise. In this paper, facial recognition in RGB images is performed considering their local texture characteristics. The RGB images are analyzed for each color channel and the Vectorial Image Representation on the Texture Space (VIR-TS) transform is applied for extraction of the local texture characteristics. Each class is represented on the texture space by three random vectors, $\vec{C}_R$, $\vec{C}_G$, and $\vec{C}_B$, where each vector corresponds to a channel and its magnitude and phase are functions of the texture. These vectors are deployed in a multi-class classifier to measure the similarities with the classes in the texture space

through the projection of a prototype vector and a test vector. Using a facial image database, the images are identified with and without added noise to their respective classes. The VIR-TS transform yields an efficiency of 100% if and only if the classes bear no added noise and if the signal-to-noise ratio is very high in the class. Otherwise, if the signal-to-noise ratio is small, the VIR-TS transform becomes less efficient at identifying the class.

Taking in consideration these results, the VIR-TS transform can be employed to build highly efficient machine vision systems. These systems would be based on the identification of textural characteristics and they could be applied in several fields, for example, medical diagnosis, security, access control, analysis of medical images, and the location of missing persons, among other applications.

2. Theory

2.1. Previous Work

The Vectorial Image Representation on the Texture Space (VIR-TS) transform represents an S grayscale digital image through a $\vec{C}$ vector in the texture space [22]. In the VIR-TS transformation, the grayscale image in a random $S = \{s_{mn}\}$ ($m = 1, 2, \dots, M; n = 1, 2, \dots, N$) array, with a size of $M \times N$ pixels and where every position has a discrete pattern $P = \{p_{i,j}\}$ ($i = 1, 2, 3; j = 1, 2, 3$), is detected by an observation window with a size of $W = I \times J = 3 \times 3$. Each P pattern is employed to calculate a texture unit vector $\vec{T}$, and thus all texture unit vectors are measured to estimate the $\vec{C}$ vector.

2.1.1. Texture Unit

Given a grayscale image $S = \{s_{mn}\}$ ($m = 1, 2, \dots, M; n = 1, 2, \dots, N$) where a random pattern is detected with an $I \times J = 3 \times 3$ mobile observation window $P = \{p_{i,j}\}$ ($i = 1, 2, 3; j = 1, 2, 3$), the texture unit is calculated in the VIR-TS transform, and next a homogeneous equation system is considered [22]:

$$C_P T = 0 \Rightarrow \begin{bmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & p_{33} \end{bmatrix} \begin{pmatrix} t_1 \\ t_2 \\ t_3 \end{pmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \Rightarrow \begin{cases} p_{11}t_1 + p_{12}t_2 + p_{13}t_3 = 0 \\ p_{21}t_1 + p_{22}t_2 + p_{23}t_3 = 0 \\ p_{31}t_1 + p_{32}t_2 + p_{33}t_3 = 0 \end{cases} \quad (1)$$

where $C_P = \begin{bmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & p_{33} \end{bmatrix}$ is called the coefficient matrix of the homogeneous linear

system, represented with a 3×3 matrix of real elements, and $T = \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}$ is called a texture

unit vector. The homogeneous equation system yields a trivial solution $t_1 = 0, t_2 = 0, t_3 = 0$, though this solution is not functional for texture recognition. Consequently, a non-trivial solution must be obtained. Based on fundamental linear algebra concepts, a non-trivial solution can be found for an equation system if and only if the matrix determinant is equal to zero, $\det[C_P] = 0$; ergo

$$\det[C_P] = \det \begin{vmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & Kp_{33} \end{vmatrix} = 0 \quad (2)$$

where the K term must satisfy condition (2). Thus, our problem is to find the value of K as a function of the elements of the pattern P . To find the value of K , the determinant of the matrix C_P is calculated and the result is set equal to zero,

$$\det \begin{vmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & Kp_{33} \end{vmatrix} = Kp_{33}p_{22}p_{11} + p_{12}p_{23}p_{31} + p_{13}p_{21}p_{32} - p_{13}p_{22}p_{31} - Kp_{33}p_{12}p_{21} - p_{11}p_{23}p_{32} = 0 \quad (3)$$

From this result the parameter K can be determined, obtaining

$$K = \frac{p_{13}p_{22}p_{31} + p_{11}p_{23}p_{32} - p_{12}p_{23}p_{31} - p_{13}p_{21}p_{32}}{p_{33}p_{22}p_{11} - p_{33}p_{12}p_{21}} \quad (4)$$

Calculating the parameter K , it is introduced to the equation system:

$$\begin{aligned} p_{11}t_1 + p_{12}t_2 + p_{13}t_3 &= 0 \\ p_{21}t_1 + p_{22}t_2 + p_{23}t_3 &= 0 \\ p_{31}t_1 + p_{32}t_2 + Kp_{33}t_3 &= 0 \end{aligned} \quad (5)$$

The solution of the homogeneous equation system can be calculated via parameterization, obtaining [22]

$$\infty - \text{soluciones} \begin{cases} t_1 = \frac{p_{13}p_{22} - p_{23}p_{12}}{p_{11}p_{22} - p_{12}p_{21}}\lambda \\ t_2 = \frac{p_{11}p_{23} - p_{21}p_{13}}{p_{11}p_{22} - p_{12}p_{21}}\lambda \\ t_3 = \lambda \end{cases} \quad \lambda \in \mathbb{R}. \quad (6)$$

Observing expression (6), for each real value of λ , a unique solution from the infinite solutions is found. Based on Equations (1) and (6), the texture unit vector can be defined by

$$\mathbf{T} = \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix} = \begin{bmatrix} \frac{p_{13}p_{22} - p_{23}p_{12}}{p_{11}p_{22} - p_{12}p_{21}}\lambda \\ \frac{p_{11}p_{23} - p_{21}p_{13}}{p_{11}p_{22} - p_{12}p_{21}}\lambda \\ \lambda \end{bmatrix} \quad (7)$$

And its cartesian representation yields the form

$$\vec{T} = t_1 \hat{\mathbf{u}}_1 + t_2 \hat{\mathbf{u}}_2 + t_3 \hat{\mathbf{u}}_3 = \frac{p_{13}p_{22} - p_{23}p_{12}}{p_{11}p_{22} - p_{12}p_{21}}\lambda \hat{\mathbf{u}}_1 + \frac{p_{11}p_{23} - p_{21}p_{13}}{p_{11}p_{22} - p_{12}p_{21}}\lambda \hat{\mathbf{u}}_2 + \lambda \hat{\mathbf{u}}_3. \quad (8)$$

where $\hat{\mathbf{u}}_1, \hat{\mathbf{u}}_2, \hat{\mathbf{u}}_3$ are the unitary vectors of a three-dimensional rectangular coordinate system. Parting from (8), the scalars $\frac{p_{13}p_{22} - p_{23}p_{12}}{p_{11}p_{22} - p_{12}p_{21}}\lambda$, $\frac{p_{11}p_{23} - p_{21}p_{13}}{p_{11}p_{22} - p_{12}p_{21}}\lambda$ and λ are the components of vector $\vec{T}$ with u_1, u_2, u_3 as their directions. Lastly, in Equation (8), the magnitude of vector $\vec{T}$ is calculated by

$$|\vec{T}| = \sqrt{\left(\frac{p_{13}p_{22} - p_{23}p_{12}}{p_{11}p_{22} - p_{12}p_{21}}\lambda\right)^2 + \left(\frac{p_{11}p_{23} - p_{21}p_{13}}{p_{11}p_{22} - p_{12}p_{21}}\lambda\right)^2 + \lambda^2}. \quad (9)$$

and their cosines are

$$\begin{aligned} \cos\alpha &= \frac{\frac{p_{13}p_{22} - p_{23}p_{12}}{p_{11}p_{22} - p_{12}p_{21}}\lambda}{|\vec{T}|} \\ \cos\beta &= \frac{\frac{p_{11}p_{23} - p_{21}p_{13}}{p_{11}p_{22} - p_{12}p_{21}}\lambda}{|\vec{T}|} \\ \cos\gamma &= \frac{\lambda}{|\vec{T}|} \end{aligned} \quad (10)$$

The following condition is satisfied:

$$\cos^2\alpha + \cos^2\beta + \cos^2\gamma = 1, \quad (11)$$

where $|\vec{T}|$ is the magnitude of vector $\vec{T}$. In simpler terms, the texture unit $\vec{T}$ is a radius vector that goes from the origin to the coordinates $t_1 = \frac{p_{13}p_{22} - p_{23}p_{12}}{p_{11}p_{22} - p_{12}p_{21}}\lambda$, $t_2 = \frac{p_{11}p_{23} - p_{21}p_{13}}{p_{11}p_{22} - p_{12}p_{21}}\lambda$, $t_3 = \lambda$.

2.1.2. Representation of a Gray-Level Image in the Texture Space

Given an image $\mathbf{S}$ with a size of $M \times N$ pixels and if this image is studied applying a $I \times J$ window size, then there are $(M - I + 1) \times (N - J + 1)$ patterns $\mathbf{P}$. Conversely, knowing that a texture unit vector $\vec{T}$ is calculated for each pattern based in a homogenous system equation, it follows that if the $\mathbf{S}$ image is analyzed with the observation window, for the n -th pattern $\mathbf{P}_n$, where $(n = 1, 2, \dots, N_p = (M - I + 1) \times (N - J + 1))$, the n -

th texture unit $\vec{T}_n$ (radius vector) is calculated. Consequently, the S image can be expressed through a series of radius vectors, such that by summing their components on the texture space, the S gray-level image is represented by a $\vec{C}$ vector defined as

$$\vec{C} = a_1\hat{u}_1 + a_2\hat{u}_2 + a_3\hat{u}_3, \quad (12)$$

where $\hat{u}_1, \hat{u}_2, \hat{u}_3$ are the unitary vectors and a_1, a_2, a_3 are the components, and they are calculated by

$$\begin{aligned} a_1 &= \sum_{n=1}^{N_p} t_{1n} \\ a_2 &= \sum_{n=1}^{N_p} t_{2n}, \\ a_3 &= \sum_{n=1}^{N_p} t_{3n} \end{aligned} \quad (13)$$

N_p is the total patterns found in the digital image. $A = (M - I + 1) \times (N - J + 1)$ and t_{1n}, t_{2n}, t_{3n} are the n -th component of the elements for t_1, t_2, t_3 , respectively. Figure 1 illustrates the $\vec{C}$ vector.

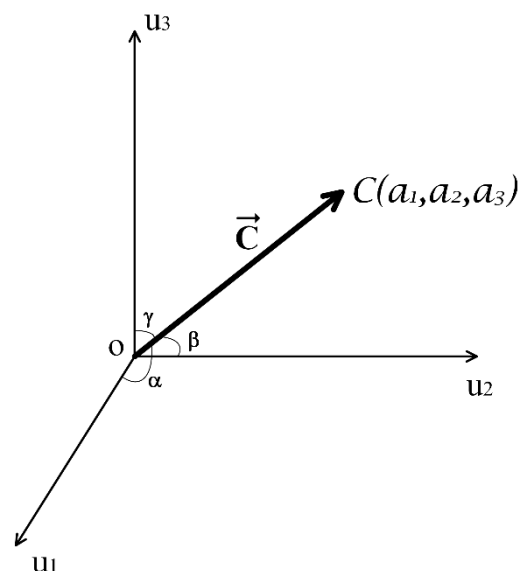


Figure 1. Graphic representation of the texture vector $\vec{C}$ with its cosines.

Based in Figure 1, the $\vec{C}$ vector has a magnitude of

$$|\vec{C}| = \sqrt{a_1^2 + a_2^2 + a_3^2}, \quad (14)$$

The cosines are

$$\begin{aligned} \cos\alpha &= \frac{a_1}{|\vec{C}|} \\ \cos\beta &= \frac{a_2}{|\vec{C}|} \\ \cos\gamma &= \frac{a_3}{|\vec{C}|} \end{aligned} \quad (15)$$

And it is fulfilled by

$$\cos^2\left(\frac{a_1}{|\vec{C}|}\right) + \cos^2\left(\frac{a_2}{|\vec{C}|}\right) + \cos^2\left(\frac{a_3}{|\vec{C}|}\right) = 1. \quad (16)$$

Considering the analysis, the grayscale image $\mathbf{S}$ can be represented like a radius vector $\vec{\mathbf{C}}$ in the texture space, the direction and magnitude of which depend on the randomness of the digital image.

2.2. VIR-TS Applied to RGB Images

As it is known, digital images can be in color, grayscale or binary. All three types are commonly employed in texture recognition. In particular, RGB images are conformed by three channels, where each channel corresponds to a basic color: red, green and blue. When these images are classified based on their local texture, their identification is implemented by performing the recognition for each color channel. The recognition efficiency depends on the texture extraction technique. Taking this into consideration, in this section, the recognition of RGB images is implemented with the VIR-TS technique. To achieve this, the procedure described in Section 2.1 is applied to each color channel, conforming the RGB image.

2.2.1. Texture Unit Definition for Each Color Channel

In this section, our proposal is that the color image $\mathbf{S}$ can be separated by channel; thus, $\mathbf{S}_R$ is the red channel image, $\mathbf{S}_G$ corresponds to the green channel image and $\mathbf{S}_B$ contains the blue channel image. The three images $\mathbf{S}_R$, $\mathbf{S}_G$ and $\mathbf{S}_B$ have a size of $M \times N$ ($m = 1, 2, 3, \dots, M; n = 1, 2, 3, \dots, N$) pixels, and each pixel yields a value within the interval of 0 to 255. Under these conditions, the Vectorial Image Representation on the Texture Space (VIR-TS) transform can be applied to each channel of the RGB image. Developing a process for each color channel of an RGB is performed following the approach in Section

2.1, Therefore, if the W_R , W_G , W_B windows detect $\mathbf{R} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$, $\mathbf{G} = \begin{bmatrix} g_{11} & g_{12} & g_{13} \\ g_{21} & g_{22} & g_{23} \\ g_{31} & g_{32} & g_{33} \end{bmatrix}$, $\mathbf{B} = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix}$ patterns from the RGB image, then the texture unit for each channel can be defined by a homogenous linear equation system [22]:

$$\mathbf{R}\mathbf{T}_R = \mathbf{0} \Rightarrow \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{pmatrix} t_{r1} \\ t_{r2} \\ t_{r3} \end{pmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad (17)$$

$$\mathbf{G}\mathbf{T}_G = \mathbf{0} \Rightarrow \begin{bmatrix} g_{11} & g_{12} & g_{13} \\ g_{21} & g_{22} & g_{23} \\ g_{31} & g_{32} & g_{33} \end{bmatrix} \begin{pmatrix} t_{g1} \\ t_{g2} \\ t_{g3} \end{pmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad (18)$$

$$\mathbf{B}\mathbf{T}_B = \mathbf{0} \Rightarrow \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix} \begin{pmatrix} t_{b1} \\ t_{b2} \\ t_{b3} \end{pmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad (19)$$

Equation (17) is the texture unit definition for the red channel: $\mathbf{T}_R = \begin{pmatrix} t_{r1} \\ t_{r2} \\ t_{r3} \end{pmatrix}$ is the unity, and the components are t_{r1}, t_{r2}, t_{r3} . Equation (18) is the texture unit definition for the green channel: $\mathbf{T}_G = \begin{pmatrix} t_{g1} \\ t_{g2} \\ t_{g3} \end{pmatrix}$ is the unity, and the components are t_{g1}, t_{g2}, t_{g3} . Finally,

Equation (19) is the texture unit definition for the blue channel: $\mathbf{T}_B = \begin{pmatrix} t_{b1} \\ t_{b2} \\ t_{b3} \end{pmatrix}$ is the unity

and the components are t_{b1}, t_{b2}, t_{b3} . Analyzing Equations (17)–(19), each color channel is analyzed independently; thus, the condition $\mathbf{T}_R \neq \mathbf{T}_G \neq \mathbf{T}_B$ is satisfied because each color channels bear different intensity values in their pixels due to the random nature of the light source. From Equations (17)–(19), the texture unit definition is fundamental in a homogeneous equation system with a trivial solution defined by $t_{r1} = 0, t_{r2} = 0, t_{r3} = 0$ (red channel), $t_{g1} = 0, t_{g2} = 0, t_{g3} = 0$ (green channels) and $t_{b1} = 0, t_{b2} = 0, t_{b3} = 0$

(blue channel). However, this trivial solution is not applicable in our texture recognition problem. Consequently, it is proposed to find an infinite solution for our homogeneous equation system and, afterwards, find a unique solution based on the resolution of the equation system through parameterization. As a first step, the equation systems (17)–(19) are structured as follows:

$$\mathbf{RT}_R = \mathbf{0} \Rightarrow \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & K_R r_{33} \end{bmatrix} \begin{pmatrix} t_{r1} \\ t_{r2} \\ t_{r3} \end{pmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad (20)$$

$$\mathbf{GT}_G = \mathbf{0} \Rightarrow \begin{bmatrix} g_{11} & g_{12} & g_{13} \\ g_{21} & g_{22} & g_{23} \\ g_{31} & g_{32} & K_G g_{33} \end{bmatrix} \begin{pmatrix} t_{g1} \\ t_{g2} \\ t_{g3} \end{pmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad (21)$$

$$\mathbf{BT}_B = \mathbf{0} \Rightarrow \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & K_B b_{33} \end{bmatrix} \begin{pmatrix} t_{b1} \\ t_{b2} \\ t_{b3} \end{pmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad (22)$$

where K_R, K_G, K_B are constant values to be determined such that the determinants of the systems of equations are equal to zero:

$$\begin{aligned} \det[\mathbf{R}] &= \det \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & K_R r_{33} \end{bmatrix} = 0 \\ \det[\mathbf{G}] &= \det \begin{bmatrix} g_{11} & g_{12} & g_{13} \\ g_{21} & g_{22} & g_{23} \\ g_{31} & g_{32} & K_G g_{33} \end{bmatrix} = 0, \\ \det[\mathbf{B}] &= \det \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & K_B b_{33} \end{bmatrix} = 0 \end{aligned} \quad (23)$$

Resolving the determinants, we obtain

$$\begin{aligned} \det \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & K_R r_{33} \end{bmatrix} &= 0 \\ &= K_R (r_{33} r_{22} r_{11} - r_{33} r_{12} r_{21}) + r_{12} r_{23} r_{31} + r_{13} r_{21} r_{32} - r_{13} r_{22} r_{31} - r_{11} r_{23} r_{32} = 0 \\ \det \begin{bmatrix} g_{11} & g_{12} & g_{13} \\ g_{21} & g_{22} & g_{23} \\ g_{31} & g_{32} & K_G g_{33} \end{bmatrix} &= 0 \\ &= K_G (g_{33} g_{22} g_{11} - g_{33} g_{12} g_{21}) + g_{12} g_{23} g_{31} + g_{13} g_{21} g_{32} - g_{13} g_{22} g_{31} - g_{11} g_{23} g_{32} = 0 \\ \det \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & K_B b_{33} \end{bmatrix} &= 0 \\ &= K_B (b_{33} b_{22} b_{11} - b_{33} b_{12} b_{21}) + b_{12} b_{23} b_{31} + b_{13} b_{21} b_{32} - b_{13} b_{22} b_{31} - b_{11} b_{23} b_{32} = 0 \end{aligned} \quad (24)$$

From Equation (24), the constant values, K_R, K_G, K_B , are calculated with

$$\begin{aligned} K_R &= \frac{r_{13} r_{22} r_{31} + r_{11} r_{23} r_{32} - r_{12} r_{23} r_{31} - r_{13} r_{21} r_{32}}{r_{33} r_{22} r_{11} - r_{33} r_{12} r_{21}} \\ K_G &= \frac{g_{13} g_{22} g_{31} + g_{11} g_{23} g_{32} - g_{12} g_{23} g_{31} - g_{13} g_{21} g_{32}}{g_{33} g_{22} g_{11} - g_{33} g_{12} g_{21}} \\ K_B &= \frac{b_{13} b_{22} b_{31} + b_{11} b_{23} b_{32} - b_{12} b_{23} b_{31} - b_{13} b_{21} b_{32}}{b_{33} b_{22} b_{11} - b_{33} b_{12} b_{21}} \end{aligned} \quad (25)$$

Once the constant values are known, the system of Equations (20)–(22) can be expressed by

$$\begin{aligned} r_{11} t_{r1} + r_{12} t_{r2} + r_{13} t_{r3} &= 0 \\ r_{21} t_{r1} + r_{22} t_{r2} + r_{23} t_{r3} &= 0 \\ r_{31} t_{r1} + r_{32} t_{r2} + K_R r_{33} t_{r3} &= 0 \end{aligned} \quad (26)$$

$$\begin{aligned} g_{11} t_{g1} + g_{12} t_{g2} + g_{13} t_{g3} &= 0 \\ g_{21} t_{g1} + g_{22} t_{g2} + g_{23} t_{g3} &= 0 \\ g_{31} t_{g1} + g_{32} t_{g2} + K_G g_{33} t_{g3} &= 0 \end{aligned} \quad (27)$$

$$\begin{aligned}
 b_{11}t_{b1} + b_{12}t_{b2} + b_{13}t_{b3} &= 0 \\
 b_{21}t_{b1} + b_{22}t_{b2} + b_{23}t_{b3} &= 0 \\
 b_{31}t_{b1} + b_{32}t_{b2} + K_B b_{33}t_{b3} &= 0
 \end{aligned}
 \quad (28)$$

If the constants indicated in (25) are taken into consideration in the homogenous system of Equations (26)–(28), consequently, there are infinite solutions for each equation system. To find a unique solution, the system of equations can be resolved through parameterization, as was done in Reference [22], in a manner such that we reach

$$\infty - \text{solutions} \begin{cases} t_{r1} = \frac{r_{13}r_{22} - r_{23}r_{12}}{r_{11}r_{22} - r_{12}r_{21}} \lambda \\ t_{r2} = \frac{r_{11}r_{23} - r_{21}r_{13}}{r_{11}r_{22} - r_{12}r_{21}} \lambda \\ t_{r3} = \lambda \end{cases} \quad \lambda \in \mathbb{R} \quad (29)$$

$$\infty - \text{solutions} \begin{cases} t_{g1} = \frac{g_{13}g_{22} - g_{23}g_{12}}{g_{11}g_{22} - g_{12}g_{21}} \lambda \\ t_{g2} = \frac{g_{11}g_{23} - g_{21}g_{13}}{g_{11}g_{22} - g_{12}g_{21}} \lambda \\ t_{g3} = \lambda \end{cases} \quad \lambda \in \mathbb{R} \quad (30)$$

$$\infty - \text{solutions} \begin{cases} t_{b1} = \frac{b_{13}b_{22} - b_{23}b_{12}}{b_{11}b_{22} - b_{12}b_{21}} \lambda \\ t_{b2} = \frac{b_{11}b_{23} - b_{21}b_{13}}{b_{11}b_{22} - b_{12}b_{21}} \lambda \\ t_{b3} = \lambda \end{cases} \quad \lambda \in \mathbb{R} \quad (31)$$

From (29)–(31), for each λ value, there is a unique solution for the homogeneous equation system. In this way, in the texture space, each texture unit calculated for each color channel is represented like a radio vector conformed by three components and then the texture unit can be defined in vector form as follows:

$$\vec{T}_R = t_{r1} \hat{u}_1 + t_{r2} \hat{u}_2 + \lambda \hat{u}_3 \quad (32)$$

$$\vec{T}_G = t_{g1} \hat{u}_1 + t_{g2} \hat{u}_2 + \lambda \hat{u}_3. \quad (33)$$

$$\vec{T}_B = t_{b1} \hat{u}_1 + t_{b2} \hat{u}_2 + \lambda \hat{u}_3 \quad (34)$$

Considering this, when an RGB image is analyzed with the VIR-TS transform, the texture unit is constituted by three vectors, $\vec{T}_R$, $\vec{T}_G$, $\vec{T}_B$, where their magnitude and direction depend on all elements in the patterns.

2.2.2. Magnitude and Cosines

Once the texture unit has been defined in vector form for each channel (see Equations (32)–(34)), its magnitude and phase can be measured through

$$|\vec{T}_R| = \sqrt{(t_{r1})^2 + (t_{r2})^2 + \lambda^2} \quad (35)$$

$$|\vec{T}_G| = \sqrt{(t_{g1})^2 + (t_{g2})^2 + \lambda^2}. \quad (36)$$

$$|\vec{T}_B| = \sqrt{(t_{b1})^2 + (t_{b2})^2 + \lambda^2} \quad (37)$$

where $|\vec{T}_R|$, $|\vec{T}_G|$, $|\vec{T}_B|$ are the magnitude for each texture unit and the cosines are estimated through

$$\begin{array}{lll}
 \text{red channel} & \text{green channel} & \text{blue channel} \\
 \cos \alpha_r = \frac{t_{r1}}{|\vec{T}_R|} & \cos \alpha_g = \frac{t_{g1}}{|\vec{T}_G|} & \cos \alpha_b = \frac{t_{b1}}{|\vec{T}_B|} \\
 \cos \beta_r = \frac{t_{r2}}{|\vec{T}_R|} & \cos \beta_g = \frac{t_{g2}}{|\vec{T}_G|} & \cos \beta_b = \frac{t_{b2}}{|\vec{T}_B|} \\
 \cos \gamma_r = \frac{\lambda}{|\vec{T}_R|} & \cos \gamma_g = \frac{\lambda}{|\vec{T}_G|} & \cos \gamma_b = \frac{\lambda}{|\vec{T}_B|}
 \end{array} \quad (38)$$

Satisfying these conditions:

$$\begin{aligned}
\cos^2\alpha_r + \cos^2\beta_r + \cos^2\gamma_r &= 1 \\
\cos^2\alpha_g + \cos^2\beta_g + \cos^2\gamma_g &= 1. \\
\cos^2\alpha_b + \cos^2\beta_b + \cos^2\gamma_b &= 1
\end{aligned}
\tag{39}$$

By the analysis, we can determine that noise affects the direction and magnitude of the texture unit, and such changes yield an effect on the classification efficiency of the VIR-TS transform.

2.2.3. Algorithm

This section describes the algorithm to calculate the Vectorial Image Representation on the Texture Space (VIR-TS) transform of a color image and then represent it on the texture space. The algorithm consists of the following steps: (a) the color image $\mathbf{S}$ is separated in channels, obtaining the images $\mathbf{S}_R$, $\mathbf{S}_G$ and $\mathbf{S}_B$; (b) three mobile observation windows are defined, $W_R = I \times J = 3 \times 3$, $W_G = I \times J = 3 \times 3$, and $W_B = I \times J = 3 \times 3$; (c) the windows are displaced pixel by pixel across the image under study, with the W_R window moving over the $\mathbf{S}_R$ image, the W_G window traversing image $\mathbf{S}_G$ and the W_B windows displacing over the $\mathbf{S}_B$ image; (d) for each position over the images, the W_R , W_G , and W_B windows find a pattern $\mathbf{R}_k = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$, $\mathbf{G}_k = \begin{bmatrix} g_{11} & g_{12} & g_{13} \\ g_{21} & g_{22} & g_{23} \\ g_{31} & g_{32} & g_{33} \end{bmatrix}$, $\mathbf{B}_k = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix}$, where $k = 0, 1, 2, 3, \dots, N_p - 1$; (e) the texture units $\vec{T}_{R_k}$, $\vec{T}_{G_k}$, $\vec{T}_{B_k}$ are calculated according to Sections 2.2.1 and 2.2.2; (f) the procedure is repeated from (a) to (e) for each position on the image under study, all possible texture units $\vec{T}_{R_k}$, $\vec{T}_{G_k}$, $\vec{T}_{B_k}$ are calculated, and thus every color channel is represented by a series of radio vectors [22]; (g) the $\vec{C}_R$, $\vec{C}_G$, $\vec{C}_B$ vectors are calculated, and thus the $\mathbf{S}$ image is represented by three vectors where each vector corresponds to an image channel:

$$\vec{C}_R = a_{r1}\hat{\mathbf{u}}_1 + a_{r2}\hat{\mathbf{u}}_2 + a_{r3}\hat{\mathbf{u}}_3 \tag{40}$$

$$\vec{C}_G = a_{g1}\hat{\mathbf{u}}_1 + a_{g2}\hat{\mathbf{u}}_2 + a_{g3}\hat{\mathbf{u}}_3. \tag{41}$$

$$\vec{C}_B = a_{b1}\hat{\mathbf{u}}_1 + a_{b2}\hat{\mathbf{u}}_2 + a_{b3}\hat{\mathbf{u}}_3 \tag{42}$$

where the components for the vectors, $\vec{C}_R$, $\vec{C}_G$, $\vec{C}_B$, are calculated employing (43).

$$\begin{array}{ccc}
\text{red channel components} & \text{green channel components} & \text{blue channel components} \\
a_{r1} = \sum_{k=0}^{N_p-1} t_{r1k} & a_{g1} = \sum_{k=0}^{N_p-1} t_{g1k} & a_{b1} = \sum_{k=0}^{N_p-1} t_{b1k} \\
a_{r2} = \sum_{k=0}^{N_p-1} t_{r2k} & a_{g2} = \sum_{k=0}^{N_p-1} t_{g2k} & a_{b2} = \sum_{k=0}^{N_p-1} t_{b2k} \\
a_{r3} = \sum_{k=0}^{N_p-1} t_{r3k} & a_{g3} = \sum_{k=0}^{N_p-1} t_{g3k} & a_{b3} = \sum_{k=0}^{N_p-1} t_{b3k}
\end{array} \tag{43}$$

The total radio vectors will be $N_p = (M - I + 1) \times (N - J + 1)$, as was aforementioned. In Figure 2b, the vectors (40)–(42) are represented in the texture space. It can be noted that the vectors yield a unique position because the image is noiseless. In this case, if $\vec{C}_R$, $\vec{C}_G$, $\vec{C}_B$ are deployed as feature vectors in a classifier, the efficiency of the VIR-TS transforms will result in 100%. This maximum efficiency is attributed to the lack of randomness in the bidimensional signal (RGB image).

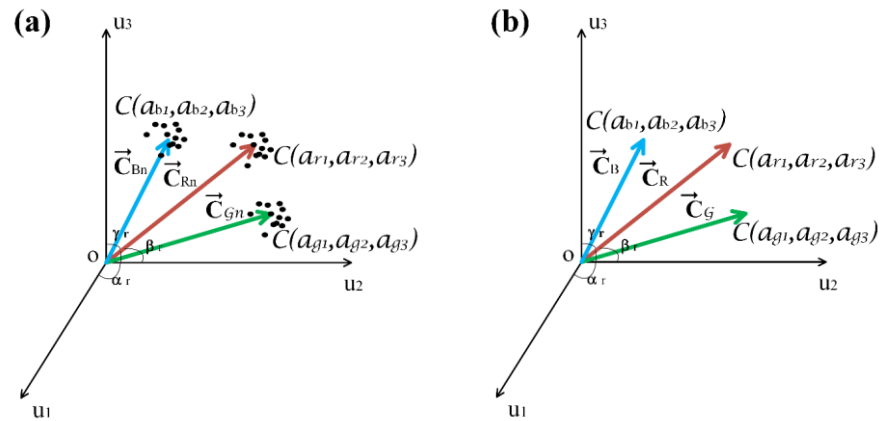


Figure 2. Texture space representation of the RGB image: (a) $\vec{C}_{Rn}$, $\vec{C}_{Gn}$, $\vec{C}_{Bn}$ vectors with added noise; (b) $\vec{C}_R$, $\vec{C}_G$, $\vec{C}_B$ vectors without present noise.

On the other hand, when an RGB image is acquired through an artificial vision system, this contains noise produced by the light source, photodetection, and the algorithm for signal and instrumentation processing, causing randomness in texture of the units $\vec{T}_R$, $\vec{T}_G$, $\vec{T}_B$; as a consequence, randomness is found in $\vec{C}_R$, $\vec{C}_G$, $\vec{C}_B$. Ergo, when the VIR-TS is calculated for the RGB image, the vectors $\vec{C}_R$, $\vec{C}_G$, $\vec{C}_B$ will wield additional noise, and then they can be expressed through

$$\vec{C}_{Rn} = \vec{C}_R + n \quad (44)$$

$$\vec{C}_{Gn} = \vec{C}_G + n. \quad (45)$$

$$\vec{C}_{Bn} = \vec{C}_B + n \quad (46)$$

where $\vec{C}_{Rn}$, $\vec{C}_{Gn}$, $\vec{C}_{Bn}$ are the vectors with noise and n is the random noise of the light source, photodetection, signal and instrumental processing, as was mentioned above. In Figure 2a the three vectors $\vec{C}_{Rn}$, $\vec{C}_{Gn}$, $\vec{C}_{Bn}$ are represented in the texture space, where the black dots are an indicator of a possible random position caused by the noise present in the image. This randomness leads to errors in texture recognition, and consequently, the efficiency of the VIR-TS transform is reduced due to the noise in the RGB image.

2.3. Classifier

In reference [22], a multi-class classifier on the texture space was proposed and employed for gray-level image classification. The characteristic vector developed was named $\vec{C}$, which represents a grayscale image in the texture space. This section proposes a scheme deploying this classifier to classify RGB images on the texture space where $\vec{C}_R$, $\vec{C}_G$, $\vec{C}_B$ are employed as feature vectors. Likewise, it consists of two phases: learning and recognition. Figure 3 presents a scheme of its structure.

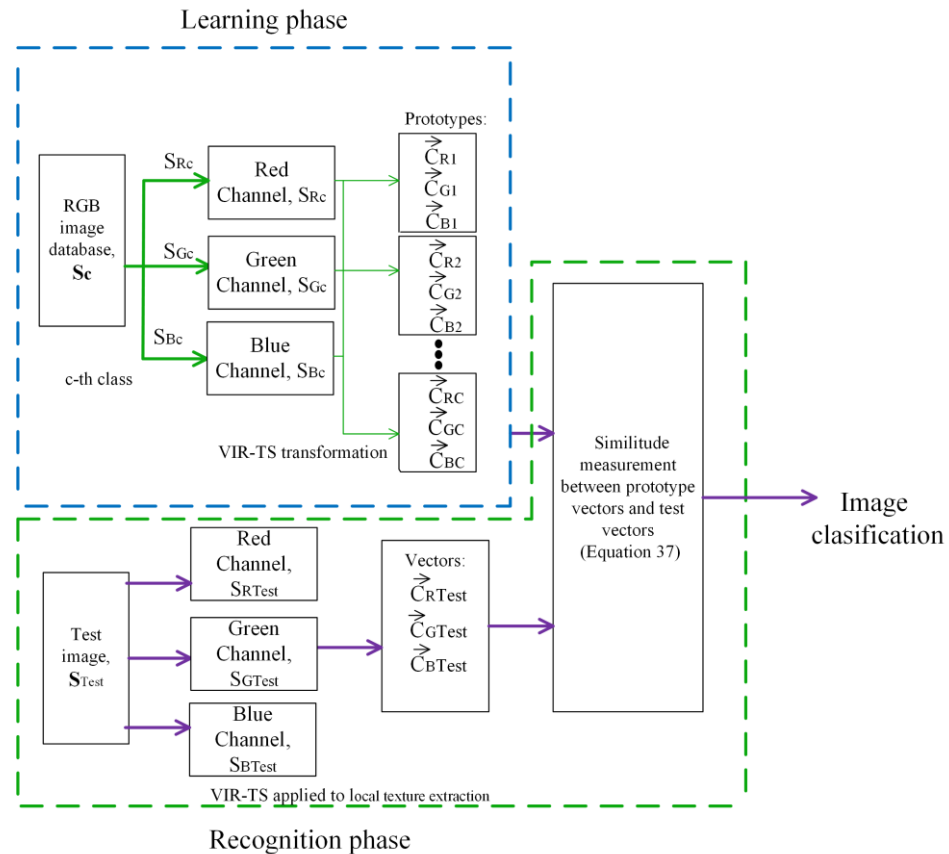


Figure 3. RGB image classifier based on VIR-TS transform.

2.3.1. Learning and Recognition Phases

From Figure 3, in the learning phase prototype vectors are generated from RGB images previously identified by a human expert, S_c ($c = 1, 2, 3, \dots, C$), where c -th is the c -nth class and C is the total number of classes. For each class, the $\vec{C}_{Rc}$, $\vec{C}_{Gc}$, $\vec{C}_{Bc}$ vectors are calculated. These vectors are generated with the following procedure: (a) the S_c image is separated by its color channels, obtaining separated channel images (S_{Rc} , S_{Gc} and S_{Bc}) for the c -th class; (b) by applying the algorithm from Section 2.2 to the S_{Rc} , S_{Gc} and S_{Bc} images, its VIR-TS transform is calculated and then the prototype vectors for the c -th class are obtained, $\vec{C}_{Rc}$, $\vec{C}_{Gc}$, $\vec{C}_{Bc}$; (c) the procedure indicated in steps a and b is repeated for each class.

In the recognition phase, an unknown test image S_{Test} is separated by its color channels, obtaining S_{RTest} , S_{GTest} and S_{BTest} images. (a) Each S_{RTest} , S_{GTest} and S_{BTest} image has its VIR-TS transform calculated, and then the $\vec{C}_{RTest}$, $\vec{C}_{GTest}$, $\vec{C}_{BTest}$ vectors are obtained. (b) The similarity between the test image S_{Test} and the prototype image S_c is measured in the texture space by means of a projection between vectors:

$$\max(\text{sim}(S_{RTest}, S_{Rc})) = \max(\cos \varphi_R) = \frac{\vec{C}_{RTest} \cdot \vec{C}_{Rc}}{|\vec{C}_{RTest}| |\vec{C}_{Rc}|} \quad (47)$$

$$\max(\text{sim}(S_{GTest}, S_{Gc})) = \max(\cos \varphi_G) = \frac{\vec{C}_{GTest} \cdot \vec{C}_{Gc}}{|\vec{C}_{GTest}| |\vec{C}_{Gc}|} \quad (48)$$

$$\max(\text{sim}(S_{BTest}, S_{Bc})) = \max(\cos \varphi_B) = \frac{\vec{C}_{BTest} \cdot \vec{C}_{Bc}}{|\vec{C}_{BTest}| |\vec{C}_{Bc}|} \quad (49)$$

where $\max(\text{sim}(S_{RTest}, S_{Rc}))$, $\max(\text{sim}(S_{GTest}, S_{Gc}))$, $\max(\text{sim}(S_{BTest}, S_{Bc}))$ is the maximum similarity between images, $\cos$ is the trigonometric function cosine, φ_R , φ_G , φ_B

yield the angle formed between vectors, $|\vec{C}_{RTest}|$, $|\vec{C}_{GTest}|$, $|\vec{C}_{BTest}|$ are the vector magnitudes for the test image, $|\vec{C}_{Rc}|$, $|\vec{C}_{Gc}|$, $|\vec{C}_{Bc}|$ represent the magnitude for the prototype images and the dot operator is the scalar product between two vectors. In the classifier, the S_{Test} image is assigned to a c class if and only if the projection between the test vectors and the prototype is close to the unit (or it is the unit).

2.3.2. Decision Criteria

Given that the RGB images are classified by color channels, the classifier yields three possible Boolean outputs: $A_R = 1$ or $A_R = 0$ is the acceptance or rejection of the red channel, $A_G = 1$ or $A_G = 0$ is the acceptance or rejection of the green channel and $A_B = 1$ or $A_B = 0$ is the acceptance or rejection of the blue channel. If the classifier outputs are considered Boolean variables, their possible combinations are in concordance with Table 1, where all input variables are A_R , A_G , A_B , and the output variable, O_D , is the classifier's decision.

Table 1. Decision criteria employed in the texture space multi-class classifier.

Classifier Output by Channel		Classification Decision		Case
A_R	A_R	A_R	A_R	Not applicable
0	0	0	0	Not applicable
0	0	1	0	Not applicable
0	1	0	0	Not applicable
0	1	1	1	Not rigorous (2)
1	0	0	0	Not applicable
1	0	1	1	Not rigorous (2)
1	1	0	1	Not rigorous (2)
1	1	1	1	Rigorous (1)

Note: 1 indicates acceptance and 0 indicates rejection.

Based on Table 1, two cases are considered.

Case I: The classifier has a rigorous criterion if and only if the condition $A_R = 1$, $A_G = 1$, $A_B = 1$ is satisfied.

Case II: The classifier has a not rigorous criterion if and only if the images of two channels are accepted and then the decision of the classifier satisfies the Boolean equation

$$O_D = \overline{A_R}A_GA_B + A_R\overline{A_G}A_B + A_RA_G\overline{A_B} \quad (50)$$

Applying Boolean algebra, Equation (32) can be reduced to

$$O_D = A_GA_B + A_RA_B + A_RA_G \quad (51)$$

Equation (33) becomes the decision criterion of the multi-class classifier when two color channels are classified correctly and one is rejected.

2.3.3. Texture Descriptors and Classifiers

LBP (Local Binary Patterns)

LBP are designed to describe grayscale images employing an observation window of 3×3 to obtain numeric values between 0 and 255 considering the patterns $P_{i,j} = [P_{1,1}, P_{2,1}, P_{3,1}, P_{3,2}, P_{3,3}, P_{2,3}, P_{1,3}, P_{1,2}, P_{2,2}]$, where $P_{2,2}$ is the central pixel (see Figure 4); this descriptor compares $P_{2,2}$ against each of its neighboring pixels $P = [P_{2,1}, P_{3,1}, P_{3,2}, P_{3,3}, P_{2,3}, P_{1,3}, P_{1,2}, P_{1,1}]$. If the condition $P_{i,j} \geq P_{2,2}$ is satisfied, a 1 is as-

signed, otherwise, $P_{i,j} < P_{2,2}$, and a 0 is assigned instead. An 8-bit binary value is generated, which is later converted to its respective decimal value between 0 and 255, which is stored in the resulting feature vector of R^{256} [23].

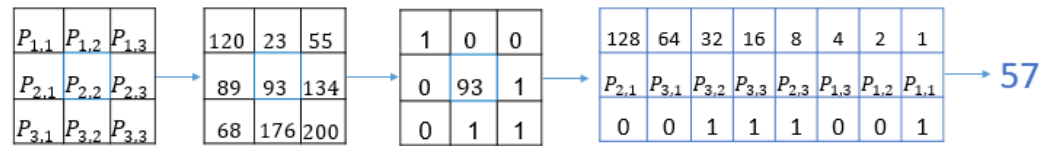


Figure 4. LBP descriptor.

CS_LBPs (Center-Symmetric Local Binary Patterns)

CS_LBPs are descriptors that, contrary to LBPs, discard the central pixel P_C ; instead, pairs of symmetric pixel values are considered by ignoring the center: (P_0, P_4) , (P_1, P_5) , (P_3, P_7) , (P_2, P_6) (see Figure 5).

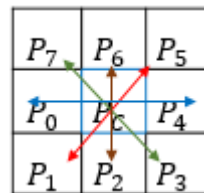


Figure 5. Symmetric pairs.

A threshold is applied to the gray tone differences employing the parameter Δ ; then, a set of 2^4 patterns is generated. Its kernel function is expressed by

$$f(x) = \sum_{j=0}^3 2^j s(P_j - P_{j+4} - \Delta - 1) \quad (52)$$

where P_j represents the intensity of the neighbor, P_{j+4} represents the intensity of the symmetric neighbor and s corresponds to the threshold function defined by

$$s(x) = \begin{cases} 1, & x \geq \Delta \\ 0, & x < \Delta \end{cases} \quad (53)$$

As a result, 4 bits are extracted, defining a dimensional space of R^{16} [24].

SVM (Support Vector Machine) Classifier

SVM is a statistical classifier commonly deployed for image classification. Give a class c composed by a set of images of a single individual, a texture descriptor (such as LBP or CS_LBP) is applied to generate a vector with textural characteristics, which is represented by x , where $x_i = \{x_1, x_2, x_3, \dots, x_i\}$; the class labels are defined by $y_l \in \{-1, 1\}$, becoming $y_l = \{y_1, y_2, y_3, \dots, y_l\}$. Equation (36) defines the hyperplane:

$$w \times x + b = 0 \quad (54)$$

Considering (w, b) of the image x_i , it can be described with Equation (37):

$$f(x_i) = \text{sign}(w \times x + b) = \begin{cases} 1 & y_l = 1 \\ -1 & y_l = -1 \end{cases} \quad (55)$$

where y_l determines the class c of the image x [25].

Random Forest Classifier

Random Forest is a popular classifier used to perform the regression and classification of datasets. It builds multiple individual classifiers to generate a prediction. Given a set of data $x_i = \{x_1, x_2, x_3, \dots, x_i\}$, where $y_l = \{y_1, y_2, y_3, \dots, y_l\}$ represents the classes c [26], the classifier determines to which class c the set of data x belongs.

3. Experimental Work

Numerical experiments are presented in this section. The image classification efficiency is tested for the VIR-TS transform, and the database employed contains face images, which were previously classified by a human expert. For the first experiment, an image is selected from the database, and random noise is added to it; hence, when the VIR-TS transform is calculated, $\vec{C}_R$, $\vec{C}_G$, $\vec{C}_B$ and $\vec{C}_{Rn}$, $\vec{C}_{Gn}$, $\vec{C}_{Bn}$ vectors have different positions due to noise. For the second experiment, the image database is recognized according to the classifier described in Section 2.3. In this experiment random noise is not added; therefore, the efficiency of the VIR-TS transform without additional noise can be verified. For the third experiment, the classification efficiency of the VIR-TS transform is studied considering the effects of noise. The results are presented in a behavior graphic of efficiency vs. SNR. For experiments 2 and 3, the decision criterion is rigorous, called case 1.

3.1. Database

The Georgia Tech (GT) database was employed [27]. It contains images captured by the signal and image processing center of the Georgia Institute of Technology. Each individual had 15 images taken in uncontrolled environments; thus, some have frontal or tilted faces, with different facial expressions, in addition to different lighting and scale and with an approximate resolution of 150×150 pixels. To perform the tests of this paper, 20 pictures (shown in Figure 6) were selected, where each individual image is considered as a class.



Figure 6. Collage of selected images [27].

3.2. FEI Database

The FEI Database consists of a collection of photographs of 50 different individuals, where each individual has a set of 15 pictures. This database was compiled by the Artificial Intelligence Laboratory of FEI in São Bernardo do Campo, São Paulo, Brazil [28].

3.3. Experiment 1: Effect of Noise on $\vec{C}_R$, $\vec{C}_G$, $\vec{C}_B$ Vectors

Based on reference [29], Gaussian distributed noise was added to the color image. The procedure is described below: (a) image S in RGB format is taken as input; (b) the RGB image is divided in the channels that compose it, generating the S_R , S_G and S_B images, where S_R is the red channel image, S_G is the green channel image and S_B is the blue channel image; (c) random noise with Gaussian distribution, $\sigma=6$, is generated; (d) the noise is added to 10% of each S_R , S_G and S_B image, where n is the noise in the color image, n_r is the noise added to the red channel, n_g is the noise added to the green channel and, lastly, n_b is the noise added to the blue channel, resulting in $S_R + n_r$, $S_G + n_g$ and $S_B + n_b$. Figure 7a presents the RGB image with no additional noise, and Figure 7b presents the result after the addition of noise: $\sigma = 6$ and (signal-to-noise ratio) $SNR = 33.85$.

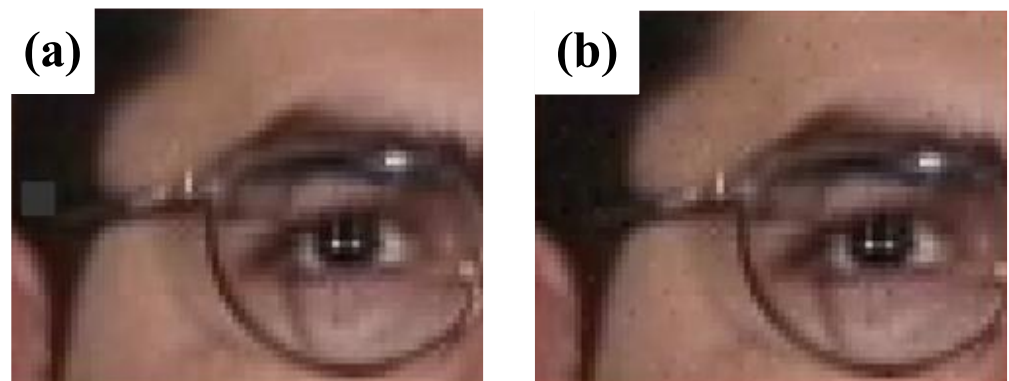


Figure 7. Segment of class 3: (a) RGB image without additional noise; (b) RGB image with noise: $\sigma = 6$ and (signal-to-noise ratio) $SNR = 33.85$.

Following the procedure described in Section 2.2, the VIR-TS transform was calculated for class 3, obtaining the result in Figure 8. Figure 8a presents the $\vec{C}_R$, $\vec{C}_G$, $\vec{C}_B$ vectors when class 3 has no additional noise. Meanwhile, Figure 8b presents the $\vec{C}_{Rn}$, $\vec{C}_{Gn}$, $\vec{C}_{Bn}$ vectors, but in this case, class 3 has noise. Comparing Figure 8a,b, when noise is added to the RGB image in every channel, the vector position changes in the texture space, increasing its randomness. This can be observed in Table 2, where the values of the $\vec{C}_R$, $\vec{C}_G$, $\vec{C}_B$, $\vec{C}_{Rn}$, $\vec{C}_{Gn}$, $\vec{C}_{Bn}$ vectors are displayed: Column 1 shows class 3; Column 2 presents the component values of the $\vec{C}_R$ and $\vec{C}_{Rn}$ vectors; Column 3 displays the component values of the $\vec{C}_G$ and $\vec{C}_{Gn}$ vectors; and last, Column 4 shows the component values for the $\vec{C}_B$ and $\vec{C}_{Bn}$ vectors.

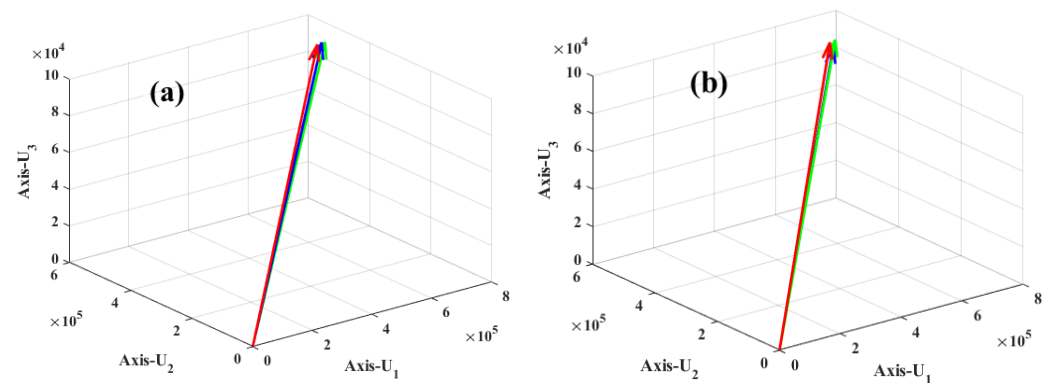


Figure 8. Texture space representation of class 3: (a) $\vec{C}_R$, $\vec{C}_G$, $\vec{C}_B$ vectors; (b) $\vec{C}_{Rn}$, $\vec{C}_{Gn}$, $\vec{C}_{Bn}$ vectors.

Table 2. Vector positions in the texture space when class 3 has no additional noise and with additional noise.

$\vec{C}_R$, $\vec{C}_G$, $\vec{C}_B$ vectors calculated for class 3 without additional noise									
$\vec{C}_R$			$\vec{C}_G$			$\vec{C}_B$			
Components			Components			Components			
t_{r1}	t_{r2}	t_{r3}	t_{g1}	t_{g2}	t_{g3}	t_{b1}	t_{b2}	t_{b3}	
802,100	549,400	105,600	852,300	570,400	103,300	833,500	563,800	104,400	
Class 3									
$\vec{C}_{Rn}$, $\vec{C}_{Gn}$, $\vec{C}_{Bn}$ vectors calculated for class 3 with additional noise									
$\vec{C}_{Rn}$			$\vec{C}_{Gn}$			$\vec{C}_{Bn}$			
Components			Components			Components			
$t_{r1} + n_r$	$t_{r2} + n_r$	$t_{r3} + n_r$	$t_{g1} + n_g$	$t_{g2} + n_g$	$t_{g3} + n_g$	$t_{b1} + n_b$	$t_{b2} + n_b$	$t_{b3} + n_b$	
724,200	528,100	109,700	764,900	548,900	107,900	754,300	545,400	104,400	

Analyzing Figure 8 and Table 2, on the texture space, the $\vec{C}_R$, $\vec{C}_G$, $\vec{C}_B$ vectors have a different position from the $\vec{C}_{Rn}$, $\vec{C}_{Gn}$, $\vec{C}_{Bn}$ vectors due to the noise added to class 3, causing a greater randomness in the $\vec{C}_{Rn}$, $\vec{C}_{Gn}$, $\vec{C}_{Bn}$ vectors. These position changes account for errors in image classification when the VIR-TS transform is employed for recognition.

3.4. Experiment 2: Image Classification of RGB Images Without Added Noise

This section verifies the efficiency of the VIR-TS transform when random noise is not added to any of the 20 classes, as shown in Figure 6. The multi-class classifier was described in Section 2.3, and its characteristic vector is defined by three ($\vec{C}_R$, $\vec{C}_G$, $\vec{C}_B$) vectors. In the multi-class classifier, each RGB image is considered a class, and the $\vec{C}_R$, $\vec{C}_G$, $\vec{C}_B$ vectors are calculated for each S_c class with the following procedure: (a) the c -th class is separated by channels, obtaining the S_{Rc} , S_{Gc} and S_{Bc} images; (b) the VIR-TS transform is calculated for the S_{Rc} , S_{Gc} and S_{Bc} images, resulting in the $\vec{C}_{Rc}$, $\vec{C}_{Gc}$, $\vec{C}_{Bc}$ vectors; (c) steps a and b are repeated for each class until all prototypes are generated for the classifier; (d) a test image S_{Test} is divided in channels to generate the S_{RTest} , S_{GTest} and S_{BTest} images; (e) each S_{RTest} , S_{GTest} and S_{BTest} image has its VIR-TS transform calculated to obtain the $\vec{C}_{RTest}$, $\vec{C}_{GTest}$, $\vec{C}_{BTest}$ vectors; (f) the resulting $\vec{C}_{RTest}$, $\vec{C}_{GTest}$, $\vec{C}_{BTest}$ vectors are compared against the prototype vectors, $\vec{C}_{Rc}$, $\vec{C}_{Gc}$, $\vec{C}_{Bc}$, where the similarity between vectors is calculated by means of vector projection; (g) the test image S_{Test} is assigned to a c -th class based on Equation (8), i.e., the vectors that bear the most similarity on the texture space.

Evaluation Metrics

The effectiveness of the classification model is measured to evaluate the efficiency, taking into account the following [30,31]: TP, correctly classified positives; TN, correctly classified negatives; FP, negatives classified as positives; and FN, positives classified as negatives.

Precision identifies positive simples of the positive prediction results.

$$Precision = \frac{TP}{TP + FP} \quad (56)$$

Accuracy identifies the correctly classified classes through all the predictions.

$$Accuracy = \frac{TN + TP}{TP + TN + FP + FN} \quad (57)$$

Recall identifies correct predictions of the positive results.

$$Recall = \frac{TP}{TP + FN} \quad (58)$$

F1-Score identifies the mean harmonic between recall and precision.

$$F1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (59)$$

The image classification results were obtained through a confusion matrix where the successes appear in the main diagonal and the classification errors are the elements outside of the scope of the main diagonal. Figure 9 illustrates the classification efficiency of the VIR-TS transform if the classes have no added noise.

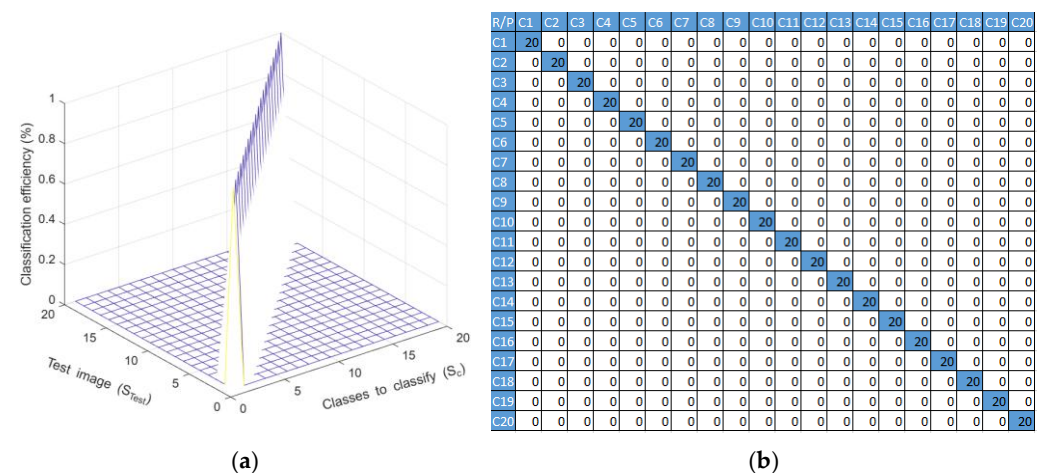


Figure 9. (a) Image classification results implementing VIR-TS transforms when classes have no added noise. (b) Confusion matrix.

In Figure 9, the classification efficiency can be calculated with

$$Ef_{\%} = \frac{1}{C} \sum_c diag(M) \times 100 \quad (60)$$

where $Ef_{\%}$ is the classification efficiency in a percentage, M is the confusion matrix, $\sum_c diag(M)$ is the sum of all elements within the confusion matrix and C is the total of classes from the image database.

$$Ef_{\%} = \frac{1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1}{20} \times 100 \quad (61)$$

The efficiency, EF, is 100% when noise is not added to the classes. Hence, we can confirm that the VIR-TS transform is efficient in image recognition applications; nonetheless, its effectiveness is due to the sensibility of random noises in the classes.

Table 3 displays a comparative study to verify the evaluation metrics of the descriptors and classifiers proposed in this paper against LBP and CS_LBP descriptors with the SVM and RF classifiers. It can be noted that the proposed technique surpasses the results in all the measured metrics.

Table 3. Classification comparison between the different descriptors and classifiers.

		FEI Face DB	GT Face DB
Proposed	Accuracy	100	100
	Precision	100	100
	Recall	100	100
	F1-Score	100	100
LBP-SVM	Accuracy	80.00	80.00
	Precision	70.00	70.00
	Recall	80.00	80.00
	F1-Score	73.33	73.33
CS-LBP-SVM	Accuracy	80.00	80.00
	Precision	70.00	70.00
	Recall	80.00	80.00
	F1-Score	73.33	73.33
LBP-RF	Accuracy	80.00	80.00
	Precision	71.67	70.00
	Recall	80.00	80.00
	F1-Score	74.17	73.33
CS-LBP-RF	Accuracy	80.00	80.00
	Precision	71.67	70.00
	Recall	80.00	80.00
	F1-Score	74.17	73.33

3.5. Experiment 3: Classification of RGB Images with Added Noise

In this section, the face images presented in Figure 6 are classified with the VIR-TS transform after each class received a Gaussian distributed random noise addition in 10% of their pixels. The images were catalogued by employing the multi-class classifier described in Section 2.3. The procedure is detailed as follows: (a) each class $\mathbf{S}_c$ was separated by its color channels, $\mathbf{S}_{Rc}$, $\mathbf{S}_{Gc}$ and $\mathbf{S}_{Bc}$, and each extracted image had a Gaussian distributed random noise added to them, obtaining $\mathbf{S}_{Rc} + n_r$, $\mathbf{S}_{Gc} + n_g$, $\mathbf{S}_{Bc} + n_b$ as a result; (b) from each image 100 sub-images with randomly added Gaussian noise were extracted, obtaining a total of 5000 images from the 50 original ones; (c) afterwards, a mean was calculated for each image and the VIR-TS transform was applied to it, obtaining the $\vec{\mathbf{C}}_{Rc}$, $\vec{\mathbf{C}}_{Gc}$, $\vec{\mathbf{C}}_{Bc}$ vectors for the c -th class; (d) an $\mathbf{S}_{Test}$ test image was acquired and separated by the component channels to generate $\mathbf{S}_{RTest}$, $\mathbf{S}_{GTest}$ and $\mathbf{S}_{BTest}$; (e) the VIR-TS transform was calculated for the $\mathbf{S}_{RTest}$, $\mathbf{S}_{GTest}$ and $\mathbf{S}_{BTest}$ images, and, consequently, the $\vec{\mathbf{C}}_{RTest}$, $\vec{\mathbf{C}}_{GTest}$, $\vec{\mathbf{C}}_{BTest}$ vectors were obtained; (f) the $\vec{\mathbf{C}}_{Rc}$, $\vec{\mathbf{C}}_{Gc}$, $\vec{\mathbf{C}}_{Bc}$ prototype vectors and the $\vec{\mathbf{C}}_{RTest}$, $\vec{\mathbf{C}}_{GTest}$, $\vec{\mathbf{C}}_{BTest}$ vectors were compared; then, the $\mathbf{S}_{Test}$ test image was assigned to the class it was closer to. The similarity between classes was measured by means of vector projection on the texture space, as indicated in Equation (31). Figure 10 illustrates the behavior of classification efficiency (%) vs. the signal-to-noise ratio.

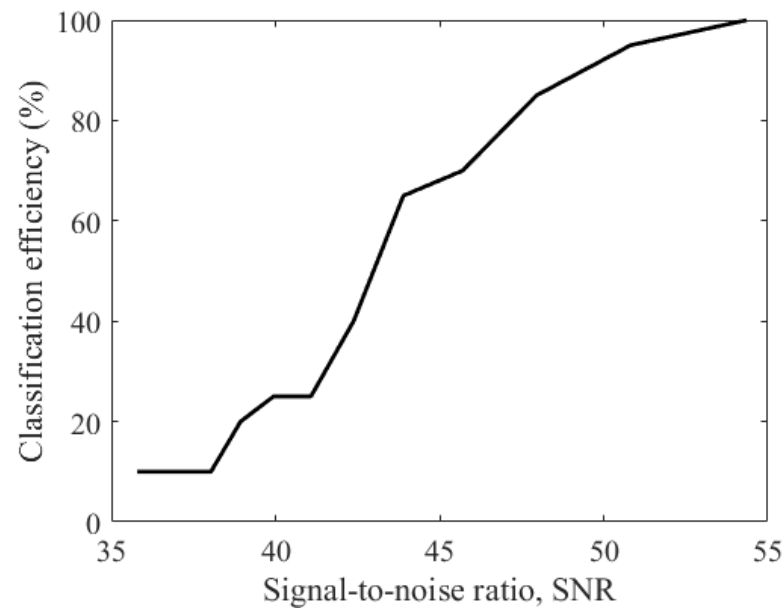


Figure 10. Behavior graphic: classification efficiency vs. signal-to-noise ratio.

In Figure 10, for $SNR = 54.36$ or greater, the VIR-TS transform presents an efficiency of 100%, and this value is equal to the results obtained with images with no added noise, as can be seen in the results in experiment 2. Under these conditions, the classes do not have enough noise to disturb the position of the $\vec{C}_{RC}$, $\vec{C}_{GC}$, $\vec{C}_{BC}$ vectors on the texture space. Contrary to this, if the SNR value diminishes due to an increment of the noise in the class, then the classification decreases given the randomness of the $\vec{C}_{RC}$, $\vec{C}_{GC}$, $\vec{C}_{BC}$ vectors higher on the texture space. Ergo, the vision systems built through the VIR-TS transform yield an efficiency lower than 100%.

Training and Testing Classification Procedure with Noise (Figure 11)

The model of the sub-images without added noise is generated as follows:

1. Take an image of each class S_c .
2. Generate sub-images.
3. Apply a texture descriptor (proposed, LBP or CS_LBP) to generate the feature vectors.
4. Train the model with 80% of the generated sub-images.
5. Perform a classification procedure with the remaining 20% with the proposed classifier, SVM and RF.
6. Generate the trained model.

Evaluation of the classification procedure of sub-images with noise was performed as follows:

7. Take an image from each class S_c and add a Gaussian noise with the function `np.random.normal` of the NumPy library and a parameter of $\sigma = 6$, resulting in $SNR = 38.3947$.
8. Generate sub-images from the images with added noise.
9. Implement the descriptors (proposed, LBP or CS_LBP) to generate the feature vectors.
10. Train the model employing the feature vectors generated with noise as input.
11. Predict which class the studied sub-image belongs to.

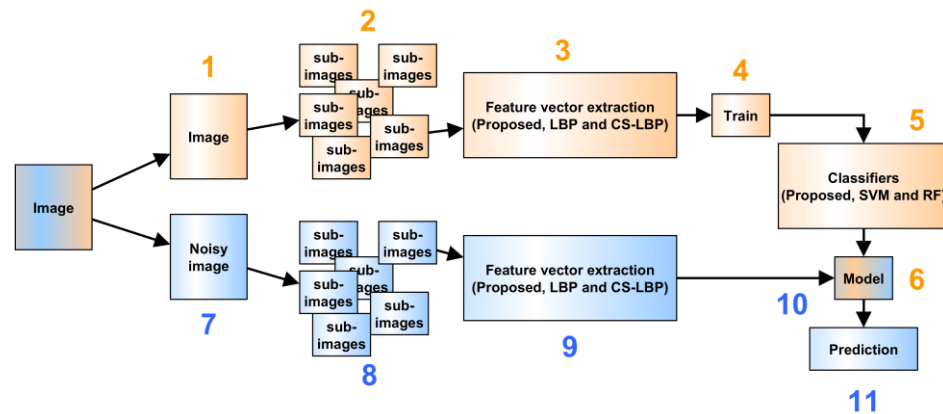


Figure 11. Training and testing procedure.

The procedure described in Training and Testing Classification Procedure with Noise (Figure 11) was performed to evaluate the classification accuracy. Table 4 presents a comparative study where the evaluation metrics with descriptors proposed in this paper are measured considering the LBP and CS_LBP descriptors and the SVM and Random Forest classifiers. From the experimental results reported in Table 4, VIR-TS is better than LBP-SVM and CS-LBP, but its efficiency is lower than the LBP-RF. Therefore, our proposal can be applied because its efficiency is higher than some texture extraction techniques reported.

Table 4. Comparison of evaluation metrics.

		FEI Face DB	GT Face DB
		Noise	Noise
Proposed	Accuracy	60.00	80.00
	Precision	47.17	72.00
	Recall	60.00	80.00
	F1-Score	51.13	74.67
LBP-SVM	Accuracy	5.64	91.98
	Precision	1.11	92.29
	Recall	5.64	91.98
	F1-Score	1.49	91.94
CS-LBP-SVM	Accuracy	14.54	76.24
	Precision	19.02	77.99
	Recall	14.54	76.24
	F1-Score	10.12	75.87
LBP-RF	Accuracy	17.38	98.64
	Precision	4.33	98.66
	Recall	17.38	98.64
	F1-Score	6.58	98.64
CS-LBP-RF	Accuracy	96.80	98.88
	Precision	96.84	98.89
	Recall	96.80	98.88
	F1-Score	96.80	98.88

4. Discussion

In this paper, facial RGB images are classified on the texture space of the Vectorial Image Representation on the Texture Space (VIR-TS) transform. The RGB images were identified based on the color compositions of their channels given that each channel was

represented by a vector, $\vec{C}_R$, $\vec{C}_G$, $\vec{C}_B$. Here, $\vec{C}_R$, $\vec{C}_G$, $\vec{C}_B$ were employed as characteristic vectors in a multi-class classifier, and the similarity between classes was measured by means of a projection between prototype vectors and test vectors. Experimentally, the classification efficiency of the VIR-TS transform was evaluated. Based on theoretical analysis and experimental results, the following points are worth noting:

- (a) The VIR-TS transform can classify RGB images on texture space, performing a recognition based on the color channels.
- (b) The VIR-TS transform returns a 100% efficiency in image identification when the classes have no added noise and if the signal-to-noise ratio is a greater value.
- (c) By reducing the signal-to noise ratio in the class, the efficiency of the VIR-TS transform decreases due to the increment of the randomness in the position of the $\vec{C}_{Rn}$, $\vec{C}_{Gn}$, $\vec{C}_{Bn}$ vectors on the texture space.
- (d) Efficient artificial vision systems can be built based on the VIR-TS transform if and only if the SNR is high.
- (e) The VIR-TS transform is not invariant to scale or rotation, and these effects on the class could reduce its image classification efficiency.
- (f) The VIR-TS transform can be applied in security systems, medical diagnosis, image reconstruction, and locating people, among other applications.
- (g) The VIR-TS is founded by concepts of linear algebra, as it is a homogeneous system of equations based on a pattern P detected by the observation window $I \times J = 3 \times 3$. On the other hand, other texture extraction techniques (LBP and CCR) are founded in probability and statistic concepts; ergo, the resulting texture unit is employed as an index for discrete histograms.
- (h) The foundation of the VIR-TS transform is based in linear algebra, while other transforms are founded in filtering processes (texture extraction techniques by signal processing), statistical events (statistical texture extraction techniques), fractals and/or polynomials (texture extraction techniques using mathematical models). Thus, the VIR-TS technique is a novel methodology used to represent the texture data, and this representation is, in fact, a texture space.

Comparing the VIR-TS technique against the statistical texture extraction techniques (LBP and CCR), the VIR-TS represents an RGB image through three vectors, $\vec{C}_R$, $\vec{C}_G$, and $\vec{C}_B$, where each vector corresponds to a color channel with three components. Meanwhile, the statistical techniques generate a discrete histogram, $p(k)$ ($k = 0, 1, 2, \dots, K - 1$), with a K length depending on the codification method and the observation window size $I \times J$. Considering the aforementioned, our method offers advantages over the statistical techniques because the computer operations required for classification are significantly reduced when two images are compared. This reduction in computer operations is noticeable when the similarity between two images is measured: VIR-TS measures the similarity comparing three vectors, $\vec{C}_R$, $\vec{C}_G$, $\vec{C}_B$, while the LBP and CCR techniques measure the similarity with discrete histograms [23], rendering the possibility of a buffer overflow.

Comparing our image classification proposal with the one reported in reference [22], both papers performed image classification employing the VIR-TS transform. However, our proposal applies the VIR-TS transform in RGB images where classes received random noise to demonstrate its high classification efficiency. Furthermore, comparing the VIR-TS transform with the techniques reported in references [22,32,33], our proposal analyzes the texture through a transformation, but in the references the texture is studied by its frequency components. Conversely, comparing our proposal with references [34–37], our proposal is founded in linear algebra, and the literature analyzes the texture with statistical fundamentals. Considering the aforementioned, our future work has the following lines

of interest: (a) study the invariance to rotation and scale for the VIR-TS transform; (b) implement artificial vision systems based on the VIR-TS transform; (c) develop a mathematic formality for the VIR-TS transform; (d) develop statistical analysis regarding the efficiency and behavior of classifiers versus noise; (e) improve the classification efficiency of the VIR-TS with techniques based on noise reduction.

5. Conclusions

In this paper, facial recognition was implemented by employing RGB images. Each color channel had its local texture characteristics extracted with the Vectorial Image Representation on the Texture Space (VIR-TS) transform, representing every RGB image by three random vectors, $\vec{C}_R$, $\vec{C}_G$, and $\vec{C}_B$, on the texture space. Experimentally, by deploying a multi-class classifier, the efficiency of the VIR-TS transform was corroborated considering two cases: without noise added to the class and with noise added to the class. In the first case, when an image had no added noise, the VIR-TS transform returned an efficiency of 100%. For the second case, when the face images had noise, the efficiency returned 100% if and only if the $SNR = 50.36$ or greater; otherwise, the efficiency decreased proportionally to the lower value of the SNR.

Based on these results, the VIR-TS transform can be implemented to design artificial vision systems with high efficiency. It can be applied in medical diagnosis, security, access control, medical image analysis, image reconstruction, and locating people, to mention some examples.

Our future work will consist of strengthening and improving this area of research to further our contribution with techniques to allow the descriptor to work with images in uncontrolled environments that present different degrees of rotation, brightness and occlusion.

Author Contributions: Conceptualization, J.T.G.-B., A.G.-B. and H.G.-B.; Methodology, J.T.G.-B., A.G.-B., L.I.J.-A. and H.G.-B.; Software, M.J.-R. and J.A.-S.; Formal Analysis, J.T.G.-B., A.C.Z., H.G.-B. and M.J.-R.; Writing—Original Draft Preparation, J.T.G.-B., A.G.-B. and A.C.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Ethical review and approval were waived for this study because the data used were obtained from publicly available databases.

Informed Consent Statement: Patient consent was waived because the study relied exclusively on data from publicly available databases.

Data Availability Statement: The data related to the results that support our conclusions are available upon request to the authors.

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

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

References

1. Nguyen-Tat, B.T.; Bui, M.Q.; Ngo, V.M. Automating attendance management in human resources: A design science approach using computer vision and facial recognition. *Int. J. Inf. Manag. Data Insights* **2024**, *4*, 100253. <https://doi.org/10.1016/j.jjime.2024.100253>.
2. Kaman, S.; Javeriya, S.B.; Makandar, A. Hybrid CNN-based detection of forged facial images using Gabor filters and local binary patterns. *Discov. Comput.* **2025**, *28*, 244. <https://doi.org/10.1007/s10791-025-09731-x>.
3. Radad, M.; Enab, A.E.; Elagooz, S.S.; El-Fishawy, N.A.; El-Rashidy, M.A. Face Anti-spoofing Detection Based on Novel Encoder Convolutional Neural Network and Texture's Grayscale Structural Information. *Int. J. Comput. Intell. Syst.* **2025**, *18*, 175. <https://doi.org/10.1007/s44196-025-00757-z>.
4. Opanasenko, V.M.; Fazilov, S.K.; Radjabov, S.S.; Kakharov, S.S. Multilevel Face Recognition System. *Cybern. Syst. Anal.* **2024**, *60*, 146–151. <https://doi.org/10.1007/s10559-024-00655-w>.
5. Khayyat, M.M.; Zamzami, N.; Zhang, L.; Nappi, M.; Umer, M. Fuzzy-CNN: Improving personal human identification based on IRIS recognition using LBP features. *J. Inf. Secur. Appl.* **2024**, *83*, 103761. <https://doi.org/10.1016/j.jisa.2024.103761>.
6. Karanwal, S. Discriminative binary pattern descriptor for face recognition. *Pattern Anal. Appl.* **2024**, *27*, 78. <https://doi.org/10.1007/s10044-024-01293-w>.
7. Yan, M.; Meng, X.; Wu, Z. A facial expression recognition method based on the combination of LBP operator and layer normalized ResNet model. In *International Conference on Computer Graphics, Artificial Intelligence, and Data Processing (ICCAID 2024)*; Xu, X., Zain, A.b.M., Eds.; SPIE: Bellingham WA, USA, 2025; p. 135601R. <https://doi.org/10.1117/12.3061194>.
8. Sedaghatjoo, Z.; Hosseinzadeh, H.; Bigam, B.S. Local Binary Pattern(LBP) Optimization for Feature Extraction. *arXiv* **2024**, arXiv:2407.18665. <https://doi.org/10.48550/arXiv.2407.18665>.
9. Karanwal, S.; Arora, N. A Comprehensive Comparative Study of Handcrafted Descriptors in Face and Palmprint Recognition. *Int. J. Image Graph.* **2026**, *26*, 2650026. <https://doi.org/10.1142/S0219467826500269>.
10. Mukhopadhyay, M.; Dey, A.; Kahali, S. A deep-learning-based facial expression recognition method using textural features. *Neural Comput. Appl.* **2023**, *35*, 6499–6514. <https://doi.org/10.1007/s00521-022-08005-7>.
11. Bouchene, M.M. Bayesian optimization of histogram of oriented gradients (HOG) parameters for facial recognition. *J. Supercomput.* **2024**, *80*, 20118–20149. <https://doi.org/10.1007/s11227-024-06259-7>.
12. Karanwal, S. Median Block-ZigZag LBP: An Efficient Face Descriptor Under Illumination Variations. In *Advances in Distributed Computing and Machine Learning*; Kar, B., Tripathy, A.K., Teng, W.-C., Sahoo, J.P., Obaidat, M.S., Eds.; Springer Nature: Singapore, 2025; pp. 79–91.
13. Alrjebi, M.M.; Aribi, H.A. Robust Face Recognition using Multi-Resolution Multi-Threshold Local Binary Patterns. *Univ. Zawia J. Nat. Sci.* **2025**, *2*, 41–47. <https://doi.org/10.26629/uzjns.2025.06>.
14. Dahmouni, A.; Abdelouahad, A.A.; Silkan, H. A noise-robust descriptor: Applications to face recognition. *J. Theor. Appl. Inf. Technol.* **2024**, *102*, 4361–4373.
15. Zhao, Y.; Jin, X.; Gao, S.; Wu, L.; Yao, S.; Jiang, Q. TAN-GFD: Generalizing face forgery detection based on texture information and adaptive noise mining. *Appl. Intell.* **2023**, *53*, 19007–19027. <https://doi.org/10.1007/s10489-023-04462-2>.
16. Salih, M.S.; Zebari, N.A.; Masoud, R.; Zebari, D.A. Deep Transfer Learning and Feature Fusion for Improving Facial Expression Recognition on JAFFE Dataset. *Appl. Comput. J.* **2025**, *5*, 439–451. <https://doi.org/10.52098/acj.20255466>.
17. Samy, S.; Gedam, V.V.; Karthick, S.; Sudarsan, J.S.; Anupama, C.G.; Ghosal, S.; Samuel, A.A. The drowsy driver detection for accident mitigation using facial recognition system. *Life Cycle Reliab. Saf. Eng.* **2025**, *14*, 299–311. <https://doi.org/10.1007/s41872-025-00314-9>.
18. Karanwal, S. Multiscale Discriminative Binary Pattern (MDBP): Novel Descriptor for Face Recognition in Pose and Illumination Variations. *IETE J. Res.* **2025**, *71*, 3057–3067. <https://doi.org/10.1080/03772063.2025.2503344>.
19. Zhang, J.; Wang, Q.; Li, J.; Yin, Y. A noise-robust and generalizable framework for facial expression recognition. *Inf. Sci.* **2025**, *719*, 122457. <https://doi.org/10.1016/j.ins.2025.122457>.
20. Chen, G.Y. An experimental study for the effects of noise on face recognition algorithms under varying illumination. *Multimed. Tools Appl.* **2019**, *78*, 26615–26631. <https://doi.org/10.1007/s11042-019-07810-y>.
21. Kobuhumure, B.; Sadiq, B.O.; Paul, A. Noise-resilient face recognition system: A review of denoising approaches and their impact on accuracy. *KIU J. Sci. Eng. Technol.* **2025**, *4*, 201–208. <https://doi.org/10.59568/KJSET-2025-4-1-21>.
22. Sánchez-Morales, M.-E.; Guillen-Bonilla, J.T.; Guillen-Bonilla, H.; Guillen-Bonilla, A.; Aguilar-Santiago, J.; Jiménez-Rodríguez, M. Vectorial Image Representation for Image Classification. *J. Imaging* **2024**, *10*, 48. <https://doi.org/10.3390/jimaging10020048>.

23. Aguilar-Santiago, J.; Guillen-Bonilla, J.T.; García-Ramírez, M.A.; Jiménez-Rodríguez, M. Identification of Lacerations Caused by Cervical Cancer through a Comparative Study among Texture-Extraction Techniques. *Appl. Sci.* **2023**, *13*, 8292. <https://doi.org/10.3390/app13148292>.
24. Fernández, A.; Álvarez, M.X.; Bianconi, F. Texture Description Through Histograms of Equivalent Patterns. *J. Math. Imaging Vis.* **2013**, *45*, 76–102. <https://doi.org/10.1007/s10851-012-0349-8>.
25. An, Y.; Li, H.; Su, T.; Wang, Y. Determining Uncertainties in AI Applications in AEC Sector and their Corresponding Mitigation Strategies. *Autom. Constr.* **2021**, *131*, 103883. <https://doi.org/10.1016/j.autcon.2021.103883>.
26. Hu, J.; Szymczak, S. A review on longitudinal data analysis with random forest. *Brief. Bioinform.* **2023**, *24*, bbad002. <https://doi.org/10.1093/bib/bbad002>.
27. Ara Nefian Face Recognition Page. Available online: https://www.anefian.com/research/face_reco.htm (accessed on 9 June 2024).
28. FEI. [En Línea]. Available online: <https://fei.edu.br/~cet/facedatabase.html> (accessed on 13 June 2026).
29. Graetzer, S.; Hopkins, C. Intelligibility prediction for speech mixed with white Gaussian noise at low signal-to-noise ratios. *J. Acoust. Soc. Am.* **2021**, *149*, 1346–1362. <https://doi.org/10.1121/10.0003557>.
30. Al-Dabbas, H.M.; Azeez, R.A.; Ali, A.E. Two Proposed Models for Face Recognition: Achieving High Accuracy and Speed with Artificial Intelligence. *Eng. Technol. Appl. Sci. Res.* **2024**, *14*, 13706–13713. <https://doi.org/10.48084/etasr.7002>.
31. Arshed, M.A.; Mumtaz, S.; Ibrahim, M.; Dewi, C.; Tanveer, M.; Ahmed, S. Multiclass AI-Generated Deepfake Face Detection Using Patch-Wise Deep Learning Model. *Computers* **2024**, *13*, 31. <https://doi.org/10.3390/computers13010031>.
32. Stuchi, J.A.; Boccato, L.; Attux, R. Frequency learning for image classification. *arXiv* **2020**, arXiv:2006.15476. <https://doi.org/10.48550/arXiv.2006.15476>.
33. Hong, D.; Wu, X.; Ghamisi, P.; Chanussot, J.; Yokoya, N.; Zhu, X.X. Invariant Attribute Profiles: A Spatial-Frequency Joint Feature Extractor for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2020**, *58*, 3791–3808. <https://doi.org/10.1109/TGRS.2019.2957251>.
34. Medelytė, S.; Rzhano, Y.; Šiaulys, A.; Lowell, K. Evaluating textural descriptors for automated image classification of stony reefs in turbid temperate waters. *Ecol. Inform.* **2025**, *90*, 103236. <https://doi.org/10.1016/j.ecoinf.2025.103236>.
35. Daradkeh, Y.I.; Gorokhovatskyi, V.; Tvoroshenko, I.; Gadetska, S.; Al-Dhaifallah, M. Statistical Data Analysis Models for Determining the Relevance of Structural Image Descriptions. *IEEE Access* **2023**, *11*, 126938–126949. <https://doi.org/10.1109/ACCESS.2023.3332291>.
36. Rahman, M.A.; Hossain, M.F.; Hossain, M.; Ahmmed, R. Employing PCA and t-statistical approach for feature extraction and classification of emotion from multichannel EEG signal. *Egypt. Inform. J.* **2020**, *21*, 23–35. <https://doi.org/10.1016/j.eij.2019.10.002>.
37. Romero-Carmona, A.P.; Rangel-Magdaleno, J.J.; Renero-Carrillo, F.J.; Ramirez-Cortes, J.M.; Peregrina-Barreto, H. Lightweight Statistical and Texture Feature Approach for Breast Thermogram Analysis. *J. Imaging* **2025**, *11*, 358. <https://doi.org/10.3390/jimaging11100358>.

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.