Pattern Recognition 61 (2017) 120–138

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Influence of normalization and color space to color texture classification

E. Cernadas [a,*], M. Fernández-Delgado [a], E. González-Rufino [b], P. Carrión [b]

[a] Centro Singular de Investigación en Tecnoloxías da Información da USC (CiTIUS), Universidade de Santiago de Compostela, Rúa Xenaro de la Fuente Domínguez, 15782 Santiago de Compostela, Spain

[b] Department of Computer Science, Campus Universitario As Lagoas, University of Vigo, Ourense, Spain

ARTICLE INFO

Article history:
Received 17 September 2014
Received in revised form
29 June 2016
Accepted 3 July 2016
Available online 14 July 2016

Keywords:
Color texture classification
Color space
Image normalization
Wavelet transforms
Gabor transforms
Local binary patterns
Learning methods

ABSTRACT

Color texture classification has recently attracted significant attention due to its multiple applications. The color texture images depend on the texture surface and its albedo, the illumination, the camera and its viewing position. A key problem to get an acceptable performance is the ambient illumination, which can vary the perceived structures in the surface. Given a color texture classification problem, it would be desirable to know which is the best approach to solve the problem making the minimal assumptions about the illumination conditions. The present work does an exhaustive evaluation of the state-of-the-art color texture classification methods, considering 5 different color spaces, 12 normalization methods to achieve illumination invariances, 19 texture feature vectors and 23 pure color feature vectors. Our experiments allow to conclude that parallel approaches are better than integrative approaches for color texture classification achieving the first positions in the Friedman ranking. Multiresolution Local Binary Patterns (MLBP) are the best intensity texture features, followed by wavelet and Gabor filters combined with luminance–chrominance color spaces (Lab and $Lab2000_{HL}$), and for pure color classification the best are First Order Statistics (FOS) calculated in RGB color space. For intensity texture features, the learning methods work better on the four smallest datasets, although they could not be tested in other four bigger datasets due to its huge computational cost, nor with color texture classification. Normalization and color spaces slightly increase the average accuracy of color texture classification, although the differences achieved using normalization are not statistically significant in a paired *T*-Test. $Lab2000_{HL}$ and RGB are the best color spaces, but the former is the slowest one. Regarding elapsed time, the best vector features MLBP for intensity texture, Daub4 (Daubechies filters using mean and variance statistics) for color texture and FOS, for pure color are nearly the fastest or are in the middle interval of all the tested methods.

1. Introduction

Texture analysis is an area that has been studied extensively [1]. Image textures are defined as visual patterns appearing in the image. Texture classification is concerned with the problem of assigning a sample image to one in the set of known texture classes. Originally, texture classification was performed on grayscale images, thus discarding color information. Many gray level texture descriptors have been developed and successfully used in numerous domains for image classification: industrial inspections [2], food science [3,4], content-based image retrieval [5], medicine [6], face recognition [7] among others. Nowadays, the cameras register RGB color images and there are proved evidences that color information improves the overall classification performance [8–11]. So, color texture analysis, which uses chromatic and textural properties to characterize an image, has recently attracted significant attention [12–15].

The methods used for texture classification usually consist of the following steps: preprocessing to make suitable the image to the next step, feature extraction to transform the image into a texture feature vector, and classification to assign the feature vector to one of the available texture classes. A texture image is a function of the texture surface and its albedo, and the ambient illumination. Illumination variation is a very important issue in color texture classification because it can change the perceived structures in the image. The variation in ambient illumination can be due to the spectral variations in the illuminant and to the camera and its viewing position. So, the performance of a color texture classification approach can be affected by the illumination conditions. Given a color texture classification problem, it would be desirable to know which is the best approach to solve the

* Corresponding author.
E-mail addresses: eva.cernadas@usc.es (E. Cernadas), manuel.fernandez.delgado@usc.es (M. Fernández-Delgado), nrufino@uvigo.es (E. González-Rufino), pcarrion@uvigo.es (P. Carrión).

http://dx.doi.org/10.1016/j.patcog.2016.07.002
0031-3203/

problem making the minimal assumptions about the illumination conditions. The solutions proposed to achieve illumination invariance can be enclosed in two types: choice of a pertinent color space [16,17,8], or image normalization for achieving illumination invariance [12]. Both solutions are included in the pre-processing step.

The approaches to analyze color and texture information in images can be grouped into parallel and integrative [10]. The parallel approach joins the gray-level or intensity texture features of the image and the pure color features. The integrative methods, in their simplest version, use the union of the gray level texture features for each color channel. The more sophisticated integrative methods imply the collective analysis of color and texture properties extracted from the color images. This analysis requires vectorial computations, that are more complex and less intuitive than their scalar equivalents. Consequently, the majority of published works compute the texture features on gray level images [8,18], or analyze gray level texture for each color channel [14].

The intensity texture extraction methods can be categorized into: statistical, spectral, structural, model-based and learning approaches [1]. Statistical and spectral techniques are the most popular in the literature of texture classification. Structural approach usually considers the texture as a composition of texture primitives [19] and it only performs well on very regular textures. Model-based approaches, such as the use of Markov Random Fields (MRF) [20], are not widely extended for image classification [21]. The bag-of-features (BoF) framework [22] and texton dictionary-based methods [23–26] can be considered as learning methods that need a learning and representation stage to extract the feature vector that represents the image.

The literature provides several experiments comparing several aspects of color texture classification. Drimbarean and Whelan [8] conducted experiments on 16 images of VisTex dataset to conclude that the use of color improves the performance of gray level texture classification. They use five color spaces (RGB, HSI, CIE-XYZ, CIE-LAB and YIQ) and three texture features: Discrete Cosine Transform (DCT), Gabor filters and Co-occurrence approach, concluding that the best is YIQ color space using integrative approaches and DCT features. Paschos [27] experimentally analyzed the impact of color space (RGB, Lab and HSV) on color texture classification using a dataset of 50 color images and as texture descriptors the Gabor filter. He concluded that HSV is the best, followed by Lab and RGB. Mäenpää and Piatikäinen [17] experimentally compared integrative and parallel approaches on the datasets VisTex, Outex13 and Outex14 using the color spaces RGB, HSV, Lab and $I_1I_2I_3$. They use Local Binary Patterns and Gabor filter as gray level texture features, and color and color ratio histograms as color features. Kandaswamy et al. [9] analyzed the performance of nine different texture features when varying illumination conditions and degrees of affine transformation to classify color texture images in RGB color space.

All these works are limited in the number of datasets, feature descriptors, color spaces and image normalizations used. The objective of the current paper is to develop a wider comparison of the state-of-art color texture features for different combinations of image normalizations and color spaces. We use a wide range of color texture datasets acquired under variable illumination conditions, composed by the public benchmark datasets CUReT, Outex, Vistex, USPTex and ALOT. Sections 2 and 3 briefly describe the techniques used for image normalization, color space transformation and color texture feature extraction. Section 4 describes the datasets and experimental setup, and discusses the results. The main conclusions are summarized in the last section.

2. Image pre-processing

The pre-processing transforms the original texture images into a more suitable form to be used in color and texture feature extraction. This process includes: (1) the image normalization to standardize the image color range, in such a way that the extracted properties from the images are comparable; and (2) the transformation of the image to the working color space.

2.1. Image normalization

The RGB image recorded by a camera changes significantly under different imaging settings, depending on the illumination conditions in the environment, the sensing device and the physical properties of the materials. In general, the texture information of an observed object is severely affected by changes in the illuminant color, and these variations are different among color texture surfaces [12]. Normalization can be defined as any technique that aims to produce a description of an image that is invariant to the illumination conditions under which the image is taken. Some previous gray level texture classification research applied normalization before computing texture descriptors in order to minimize the effect of intensity variations [23,24,9]. More recently, normalization is also applied to color images before other processings [12,28]. The image normalization literature assumes Lambertian surfaces to model the image formation process, in which the response of the k-th sensor is given by the following equation:

$$q_k = \int_\omega E(\lambda)S(\lambda)Q_k(\lambda)d\lambda, \quad k = 1, \ldots, m \tag{1}$$

where $E(\lambda)$ is the spectral power distribution of the light source which specifies how much energy the source emits at each wavelength (λ) of the electromagnetic spectrum; $S(\lambda)$ characterizes the reflectance properties of the surface, which defines what proportion of light incident upon it is reflected by the surface; and $Q_k(\lambda)$ characterizes the sensor, which specifies its sensitivity to light energy at each wavelength of the spectrum. The integral is taken over the range of wavelength λ. Normally, the acquisition devices have three sensors, which are commonly denoted by R, G and B, corresponding to colors red, green and blue respectively.

This image formation model has inspired a number of different techniques for achieving color normalization [29], by finding a procedure that cancels out all variables in the model that are dependent on illumination. In this work we use the following invariant color representations (normalizations): Chroma, GWN, CGWN, HEQ, CLAHE, RGBcb, RGBib, Retinex, MV and L_{max}. Besides, the image is labelled as WN when any normalization is applied.

Chroma: One of the simplest invariants is a *Chromaticity* representation of the image data, derived from a RGB image by:

$$(R', G', B') = \left(\frac{R}{RGB}, \frac{G}{RGB}, \frac{B}{RGB}\right) \tag{2}$$

where $RGB = R + G + B$. A chromaticity vector (R', G', B') is invariant to a change in the intensity of the illuminant.

GWN: An invariant to changes in illumination color is achieved with the transformation *GrayWorld Normalization*:

$$R' = \frac{R}{R_{avg}}, \quad G' = \frac{G}{G_{avg}}, \quad B' = \frac{B}{B_{avg}} \tag{3}$$

where the triplet $(R_{avg}, G_{avg}, B_{avg})$ denotes the mean of all RGBs in the image.

CGWN: Finlayson et al. [30] have shown that successive and repeated application of Eqs. (2) and (3) converges to an image representation (they call a *Comprehensive Gray World Normalized*