

Automatic selection of color constancy algorithms for dark image enhancement by fuzzy rule-based reasoning.

Cepeda-Negrete, J. y Sanchez-Yanez, RE.

Cita:

Cepeda-Negrete, J. y Sanchez-Yanez, RE. (2015). *Automatic selection of color constancy algorithms for dark image enhancement by fuzzy rule-based reasoning*. *Applied Soft Computing*, 28, 1-10.

Dirección estable: <https://www.aacademica.org/jcepedanegrete/9>

ARK: <https://n2t.net/ark:/13683/pa8v/7sC>



Esta obra está bajo una licencia de Creative Commons.
Para ver una copia de esta licencia, visite
<https://creativecommons.org/licenses/by-nc-nd/4.0/deed.es>.

Acta Académica es un proyecto académico sin fines de lucro enmarcado en la iniciativa de acceso abierto. Acta Académica fue creado para facilitar a investigadores de todo el mundo el compartir su producción académica. Para crear un perfil gratuitamente o acceder a otros trabajos visite: <https://www.aacademica.org>.



Automatic selection of color constancy algorithms for dark image enhancement by fuzzy rule-based reasoning



Jonathan Cepeda-Negrete, Raul E. Sanchez-Yanez*

Universidad de Guanajuato DICIS, Carretera Salamanca-Valle de Santiago Km 3.5+1.8, Comunidad de Palo Blanco, Salamanca, Mexico, CP 36885

ARTICLE INFO

Article history:

Received 12 April 2014

Received in revised form 28 October 2014

Accepted 22 November 2014

Available online 8 December 2014

Keywords:

Fuzzy modeling
Rule-based system
Image processing
Color enhancement
Color constancy

ABSTRACT

This work introduces a fuzzy rule-based system operating as a selector of color constancy algorithms for the enhancement of dark images. In accordance with the actual content of an image, the system selects among three color constancy algorithms, the White-Patch, the Gray-World and the Gray-Edge. These algorithms have been considered because of their accurate removal of the illuminant, besides showing an outstanding color enhancement on images. The design of the rule-based system is not a trivial task because several features are involved in the selection. Our proposal consists in a fuzzy system, modeling the decision process through simple rules. This approach can handle large amounts of information and is tolerant to ambiguity, while addressing the problem of dark image enhancement. The methodology consists in two main stages. Firstly, a training protocol determines the fuzzy rules, according to features computed from a subset of training images taken from the SFU Laboratory dataset. We choose carefully twelve image features for the formulation of the rules: seven color features, three texture descriptors, and two lighting-content descriptors. In the rules, the fuzzy sets are modeled using Gaussian membership functions. Secondly, experiments are carried out using Mamdani and Larsen fuzzy inferences. For a test image, a color constancy algorithm is selected according to the inference process and the rules previously defined. The results show that our method attains a high rate of correct selection of the most well-suited algorithm for the particular scene.

© 2014 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/3.0/>).

1. Introduction

Rule-based systems allow representing knowledge, and capturing the personal expertise in a set of IF-THEN rules. In the rules, a set of premises is evaluated for concluding a result. Rule-based systems have shown to be useful in a number of applications [1–3]. Moreover, such systems increase their flexibility, robustness and interpretability, when fused with soft computing techniques [4] like fuzzy logic, which was introduced as an extension of the classical theory of sets [5,6]. Nowadays, fuzzy logic has been recognized as an effective tool for managing information in rule-based systems [7,8] because of its tolerance to imprecision, to ambiguity and to the lack of information, as occurs in tasks such as the extraction of high-level visual information. In our particular case, we determine the algorithm most well-suited to be applied on an image, according to the color content of the scene under evaluation.

Color is an important feature for pattern recognition and computer vision fields. Typical computer vision applications related to this study include feature extraction [9], image classification [10], object recognition [11,12], scene categorization [13,14], human-computer interaction [15] and color appearance models [16]. Color also represents an attribute of visual sensation and appearance of objects. It depends on three components: the reflectance of an object, the sensitivity of the cones in the eyes, and the illuminant. For a robust color-based system, the effects generated by the illumination should be removed.

The ability of a system to recognize the true colors in objects, independently of the illuminant present in a scene, is known as Color Constancy [17]. The human visual system has this innate capability to correct the color effects of the light source. However, the emulation of the same process is not trivial for machine vision systems in an unknown scene [18]. From the computational point of view, color constancy is defined as the transformation of an input image captured under an unknown lighting, to another picture apparently obtained under a known lighting, normally daylight [14]. For this reason, it is required to estimate the color of the light source in the image. The color values, corresponding to the illuminant, are used to transform the input image.

* Corresponding author. Tel.: +52 464 6479940ext2433; fax: +52 464 6479940.
E-mail addresses: j.cepedanegrete@ugto.mx (J. Cepeda-Negrete),
sanchezy@ugto.mx (R.E. Sanchez-Yanez).

In past decades, researchers have tried to solve the color constancy problem using several methods. Nonetheless, and in spite of the wide range of computer vision applications that require color constancy, a general solution to this problem has not been found. A number of algorithms using simple image statistics have been proposed, like the White-Patch [19,20], the Gray-World assumption [21], Shades-of-Gray [22], Gray-Edge [23], Spatio-Spectral [24] and Category by Correlation [25].

The color image enhancement is a very challenging research field, in comparison to the image enhancing in gray scale. A detailed and clear overview about image enhancement issues is the one by Lucchese and Mitra [26]. It is important to mention that color constancy algorithms enhance the chromatic content of images, although they have been originally developed just for the color estimation of a light source. Provenzi et al. [27] already explored the use of two color constancy algorithms for color image enhancement purposes. In particular, their work was oriented to local contrast enhancement using the White-Patch and the Gray-World algorithms in combination with an automatic color equalization technique. Image enhancement using color constancy algorithms seems to be more significant when these algorithms are applied on dark images, a premise taken into consideration in this study.

Some research works have been oriented to the selection of color constancy algorithms according to several features. The use of content-based image analysis for automatic color correction was originally proposed by Schroder and Moser [10]. This study classified the images into a number of classes (e.g. indoor, outdoor, vegetation, mountains), associated to a particular algorithm (White-Patch and Gray-World). Gasparini and Schettini [11] proposed a method for the analysis of a cast index in the images and their classification. This classification (e.g. skin, sky, sea, vegetation) also allows to detect the presence of a possible predominant color. The work by van de Weijer et al. [12] uses high-level visual information in order to model images as a mixture of semantic classes (e.g. skin, road, buildings, grass). This latter study, utilizes the visual information for selecting the best-suited color constancy algorithm. Bianco et al. [13] proposed classifying images into outdoor and indoor categories in order to select the best color constancy algorithm for each scenery. Later, they implemented an automatic selector for color constancy algorithms taking into account low-level properties of the images [28]. Most recently, Faghih and Moghaddam [29] used a classifier in order to determine the best group of color constancy algorithms for an input image and then, some of the algorithms in this group are combined using multi-objective Particle Swarm Optimization (PSO). It is important to note that these research works have been mainly addressed to the estimation of the illuminant. However, to our knowledge, there are no selection systems of color constancy algorithms focused on image enhancement purposes.

In this work, a fuzzy rule-based system is proposed for the selection of one out of the three basic color constancy algorithms aforementioned: the Gray-World, the White-Patch and the Gray-Edge. This framework is a threefold approach to solving the problem. (1) Our study is focused on the color enhancement using color constancy algorithms, at the same time that the remotion of the influence of the illuminant is solved. Besides, it is particularly focused on processing dark images. (2) An important problem for developing the rule-based system is the correct choice of the image features. Twelve low-level features are chosen carefully: seven color features, three texture descriptors, and two lighting-content features. (3) We use a set of fuzzy rules, encoding the knowledge necessary to take a decision about the most well-suited algorithm to be applied to an image under consideration. In order to perform the selection, a test image is submitted to an inference process,

where the best algorithm is chosen if one of its corresponding rules gets the highest firing strength.

The rest of this paper is organized as follows. Section 2 presents the proposed framework and introduces the image dataset used and the color constancy algorithms. Section 3 details the image features used and, in Section 4 the design of the fuzzy rule-based system is explained. Section 5 shows the results obtained by experimental tests. Finally, the conclusions are presented in Section 6.

2. Selection of a color constancy algorithm

We introduce a rule-based system for the selection of a color constancy algorithm, in order to attain chromatic improvement in a given scene, particularly a dark one. Our approach consists in a MISO (Multiple Input, Single Output) system, including twelve inputs computed from the scene under analysis and a single output, the label of the corresponding algorithm. Inputs and outputs are linguistic terms, linked through a set of *IF-THEN* rules.

Basically, our framework has been divided into two main stages. On one hand, a training protocol determines the fuzzy rules, according to features computed from a set of training images. As said before, we use twelve low-level image features for the selection process: seven color features, three texture descriptors, and two lighting-content descriptors. On the other hand, in a testing mode, given a test image, the best algorithm is chosen according to the rule evaluation in two inference models, Mamdani [30] and Larsen [31]. Fig. 1 shows a flowchart of the methodology in our study.

In order to develop the rule-based system, we use images from the SFU Laboratory dataset [32]. This dataset is conformed by 529 dark images under controlled illuminant. The complete database contains 22 scenes with minimal specularities, 9 scenes with dielectric specularities, 14 scenes with metallic specularities and 6 scenes with at least one fluorescent surface. It is important to note that this

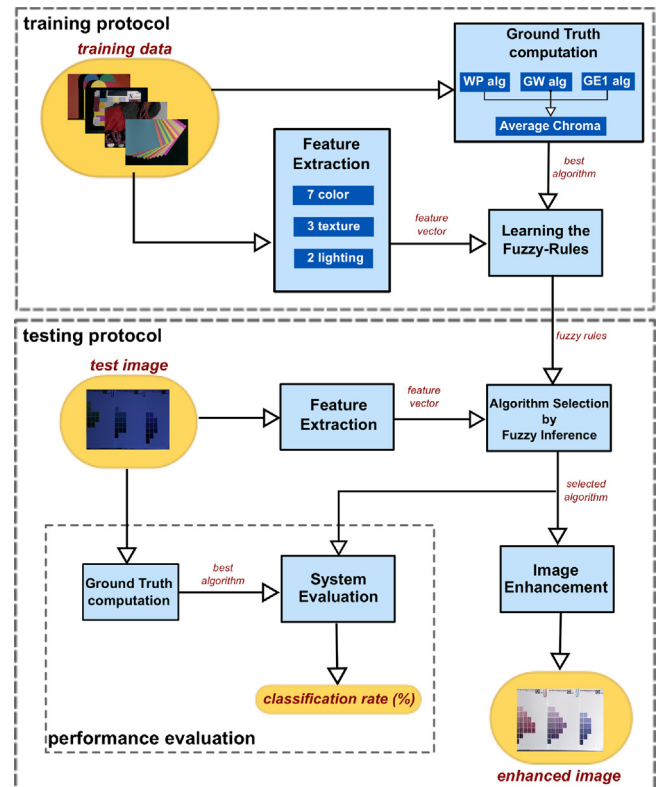


Fig. 1. Flowchart of the methodology carried out in this study.

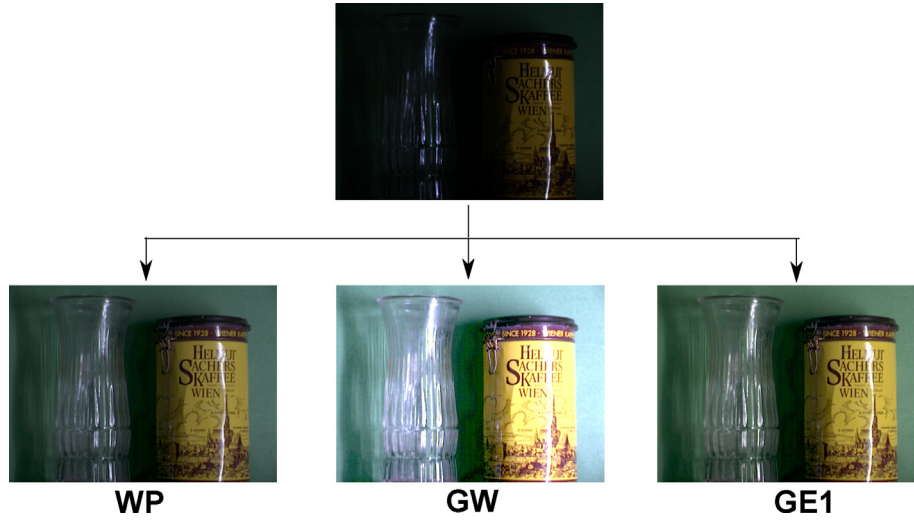


Fig. 2. Image “vase_syl-cwf” from the SFU Laboratory dataset. The original scene is processed using the three color constancy algorithms.

dataset was developed for color constancy purposes. However, it is very suitable for testing the color enhancement in dark images.

The three algorithms that are considered for the selection are

- White-Patch (WP),
- Gray-World (GW),
- 1st order Gray-Edge (GE1).

These three algorithms were considered because of their accurate removal of the illuminant, and outstanding color improvement on dark images. In addition, previously we carried out a study about the performance of color constancy in color enhancement issues. Six color constancy algorithms were compared and the WP, GW, and GE1 resulted the best algorithms in more than 90% of test cases using the SFU Laboratory image dataset. Consequently, the rest of the algorithms tested were excluded in this study. As an example, Fig. 2 shows one image belonging to the SFU Laboratory dataset. In the figure, the original scene and the outcomes after applying the three color constancy algorithms are shown.

The three color constancy algorithms considered assume that the illumination is uniform across the scene. The relationship of the color intensity under a light source in an image is given by

$$f_i(x, y) = G(x, y)R_i(x, y)I_i, \quad (1)$$

where $f_i(x, y)$ is the pixel intensity at the position (x, y) , $G(x, y)$ is a geometry factor, $R_i(x, y)$ is the reflectance of the object, I_i is the illuminant, and i corresponds to the color channel.

Once a color constancy algorithm is applied over an image $f_i(x, y)$, the outcome, $o_i(x, y)$, just depends on $G(x, y)$ and $R_i(x, y)$. Color constancy algorithms assume that the output images $o_i(x, y) = G(x, y)R_i(x, y)I_i'$, are influenced by a white light source, where $I' = \{1, 1, 1\}$ is the illuminant in the output. Then, the relation between the output and the input is

$$o_i(x, y) = G(x, y)R_i(x, y) = \frac{f_i(x, y)}{I_i}. \quad (2)$$

2.1. White-Patch

The Retinex algorithm was proposed by Land and McCann [20]. This algorithm, in its simplest form, is called White-Patch Retinex (WP) [19], which takes into account the highest value in each color channel as the white representation for the image.

Computationally, such values are found from the maximum intensity in each channel

$$I_i = \max\{f_i(x, y)\}. \quad (3)$$

Later, all pixel intensities are scaled according to the illumination computed using Eq. (2). Finlayson et al. [33] improved this algorithm using a 99% of percentage in the histogram for each color channel for the estimation of the illuminant; this improvement is the WP algorithm considered in our study.

2.2. Gray-World assumption

The Gray-World assumption (GW) is the most popular algorithm for color constancy. Proposed by Buchsbaum [21], it is based on the assumption that, on average, the colors found in real world images tend to a gray tone, and hence the illuminant is estimated using the average color of all pixels.

Basically, this algorithm consists in computing the illuminant based on $a_i = \text{mean}\{f_i(x, y)\}$. Some assumptions are proposed in [34] for simplifying the method, and the illuminant is adopted as

$$I_i \approx 2a_i. \quad (4)$$

Because $o_i(x, y) = f_i(x, y)/I_i$, the outcome image for this algorithm is given by

$$o_i(x, y) = \frac{f_i(x, y)}{2a_i}. \quad (5)$$

2.3. Gray-Edge hypothesis

The Gray-Edge hypothesis (GE) was proposed as an alternative to the Gray-World assumption [23]. This theory considers that the illuminant is a p th Minkowski norm of the reflectance differences in a scene. This method is based on the Shades-of-Gray algorithm [22], which is adopted adding the spatial derivative of an image and is given by

$$I_i = \left(\frac{\sum_x \sum_y h_i^\sigma(x, y)^p}{MN} \right)^{1/p} \quad (6)$$

where $h_i^\sigma = ((\partial^n f_{ix})^2 + (\partial^n f_{iy})^2)^{1/2} \otimes \mathbf{G}^\sigma$. M and N are the columns and rows, respectively, and \mathbf{G}^σ is a local smoothing with a Gaussian filter with standard deviation σ .

Eq. (6) describes a method producing different estimations for the illuminant color, based on three variables, (1) the order n of the

Table 1
Examples of label assignment in the pre-selected training images according to the average chroma. The best results are shown in bold.

Index	WP	GW	GE1	Class
0	22.24	23.43	21.79	2
1	21.03	22.04	19.83	2
2	30.12	22.39	25.86	1
3	33.25	23.84	27.56	1
4	14.43	14.79	12.17	2
5	13.53	13.88	12.11	2
⋮	⋮	⋮	⋮	⋮
97	27.29	19.16	25.36	1
98	23.85	12.75	22.30	1
99	25.11	13.90	22.37	1

derivative in the image, (2) the Minkowski norm p , and, (3) the scale of the local smoothing σ . In our study, we use $n=1$, $p=7$ and $\sigma=4$, according to the recommendation made by van de Weijer et al. [23].

2.4. Ground truth

A quality measure is fundamental in order to pre-select the images, in the training process for learning the rules, and in the testing protocol in order to compare and assess the performance of our system. The measure used allows us to find the outcome image with the best chromatic adaptation. Shortly after, we can conclude the algorithm that achieves the best enhancement on a particular image.

There does not exist a well-established measure without reference for evaluating image color enhancement. Human beings have a natural predisposition to qualify positively an image with intense colors [35]. To understand this, philosophical and psychological aspects of the human mind have been studied. For example, Joshi et al. [35] described aspects about aesthetics and emotions present in images, including references to understand the human mind. Some research works have tried to establish simple measures in this field. Also, we carried out experimental tests, and it was observed that the measure of the Average Chroma [36] provides meaningful information about the quality of color images. This measure is also an image feature in this study and will be discussed in the following section.

For this study, 100 images randomly taken from the SFU database are used as training data and the rest 429 for testing purposes. The three algorithms under consideration are applied to the training images. Thereby, the quality measure is computed from the outcomes for each image and thereafter, the three measures are compared. The highest value corresponds to the best algorithm for the particular image. Thus, it is possible to know the best algorithm for each image. This pre-selection will be our ground truth from now on, and we start the learning process for the selector. Table 1 illustrates with examples the correct class determined in the training process.

The a priori probability of each algorithm for being the best choice was estimated. 48% of the cases, the GW assumption was the best option. In more than one third, 36% of the images, the WP algorithm corresponds to the best choice. The remaining 16% of the images was best processed by the GE1 algorithm.

3. Image feature extraction

Features are used to describe images numerically and, they should provide information to represent a scene. If the features extracted are chosen carefully, it is expected that the feature set will extract the relevant information from the image in order to perform the desired task using this reduced representation instead of using the full image. The features considered for this work are mainly

related to the description of color content. Some of these features have been used in similar studies [28,13]. Furthermore, additional features related to textural description and lighting-content in a scene are considered. We use twelve features in this work. Seven associated with color, three of textural content, and two of lighting-content.

3.1. Color features

Different color descriptors have been used in previous works for measuring the color of an image. We can mention statistical moments of color components in a color space and color histograms [28], cast indices [11] and spectral information [37], among others.

The color features used in our system as image descriptors are the following:

- Number of colors.
- Average power spectrum value (APSV).
- Cast indices (σ , D and D_σ).
- Average chroma (μ_{cR}).
- Probability of dominant color (PDC).

In addition, it is important to note that the RLAB color space is used for the extraction of five out of the seven color features. This space is an extension of the CIELAB perceptual color space [39,38], which incorporates a more accurate model of chromatic adaptation, especially under extreme light conditions. This space is perceptually uniform, and the chromatic components, a^* and b^* , are not correlated with the lightness component L^* . The three cast indices, the average chroma and the percent of dominant color are the features computed in the RLAB color space. The transformation equations from RGB into RLAB can be seen in [16,39]. The cylindrical coordinates of RLAB are given by

$$h^R = \tan^{-1} \left(\frac{b^R}{a^R} \right), \quad (7)$$

$$c^R = \sqrt{(a^R)^2 + (b^R)^2}, \quad (8)$$

where a^R and b^R are the chromatic components in RLAB. Also, h^R and c^R are the Hue and Chroma, respectively. Thereby, some features are computed according to these components.

3.1.1. Number of colors

The number of different colors is our first feature and is related to the color range of the image. Since a number of color constancy algorithms are based on the Gray-World assumption, the number of colors is a referent of whether this assumption holds true or not. The actual colors of the pixels may cancel this assumption, but if an image contains diverse colors, then the average color is likely to be a gray value. For the RGB space, and considering 8 bits per channel, it is possible to quantize a number of 16 million colors ($2^8 \times 2^8 \times 2^8$). Bianco et al. [13] used a similar approach to obtain the number of colors in an image.

3.1.2. Average power spectrum value

The Power Spectrum Metric [37] has been used for quality image assessment and for determining the chromatic information in the image. Transforming an image into the Fourier domain enables signal analysis in the frequency domain, allowing the analysis of information that is not evident in the spatial domain. In this work, the classical Discrete Fourier Transform is used for the conversion.

The Average Power Spectrum Value (APSV) is given by

$$APSV = \frac{1}{3} \sum_i |F_i|^2, \quad (9)$$

where i stands for the color component in RGB space and $|F|^2$ is given by $|F|^2 = (1/MN) \sum_u \sum_v |F(u, v)|^2$, where M and N are the number of columns and rows, respectively. $F(u, v)$ is the image in the Fourier domain. It is important to note that the APSV indicates the quality of each image. That is, the higher the APSV is, the better the quality is. Moreover, the APSV is a no-reference image quality metric. The APSV tends to be higher when the image shows a high chromatic content under a natural illumination.

3.1.3. Cast indices

The cast indices are features for the analysis of the color distribution in the image. These indices were proposed by Gasparini and Schettini [11] for image classification (e.g. skin, sky, sea, vegetation). Also, these indices have been used in similar works [28,13].

The computation of the cast indices should be made in a color space where the luminance and the chroma components are independent. Firstly, the image is transformed into the RLAB space and the procedure proposed in [11] is carried out. Thus, the means and the variances of the a^R and the b^R color components in RLAB (μ_{a^R} , μ_{b^R} , $\sigma_{a^R}^2$ and $\sigma_{b^R}^2$) are used for computing the first cast index

$$\sigma = \sqrt{\sigma_{a^R}^2 + \sigma_{b^R}^2}, \quad (10)$$

that corresponds to the radius of an Equivalent Circle (EC) with center in $C = (\mu_{a^R}, \mu_{b^R})$. The other two cast indices are given by

$$D = \mu - \sigma, \quad (11)$$

$$D_\sigma = D/\sigma, \quad (12)$$

where $\mu = \sqrt{\mu_{a^R}^2 + \mu_{b^R}^2}$. D is a measure of how far the color distribution of the EC lies from the neutral axis ($a^R = 0$, $b^R = 0$). D_σ quantifies the strength of the cast.

3.1.4. Average chroma

Chroma is an attribute related to the intensity of the colors. This cue corresponds to a component of the RLAB color space in cylindrical coordinates. Also, Chroma is a vector representing the magnitude between a^R and b^R components in this color space, $c^R = \sqrt{(a^R)^2 + (b^R)^2}$. The higher the chroma value is, the more intense the color is. A study about the Chroma as a measure of quality color images is given in [36]. We use this measure as a simple indicator of the quality perceived by humans, and also as a color descriptor. The Average Chroma is also used in the training stage for the evaluation of the images

$$\mu_{c^R} = \text{mean}\{c^R(x, y)\}. \quad (13)$$

3.1.5. Probability of dominant color

The Probability of Dominant Color (PDC) can be extracted from a color histogram. We compute a histogram from the Hue component of RLAB. This histogram has 12 bins, where each bin registers 30 successive degrees of Hue. For example, the first bin has the occurrence of pixels in Hue between 0 and 29 degrees, the second bin registers between 30 and 59 degrees, etc. Afterwards, the probability of each bin is computed as the ratio between the number of occurrences and the total number of pixels in the image. P_{HR} corresponds to the probability distribution function of the Hue. The maximum probability registered into a bin is given by

$$\text{PDC} = \max\{P_{HR}\}. \quad (14)$$

3.2. Texture

The use of texture descriptors is inspired by the study of Bianco et al. [28]. They used some features related to texture analysis and considered that these features could describe the composition of

the image, regardless of whether the color issues were studied. Nonetheless, we take into account other texture features proposed by Haralick et al. [40], but computed using the Sum and Difference Histograms (SDH), the faster alternative proposed by Unser [41]. The relative displacement vector (\mathbf{V}) between two picture elements is a SDH parameter, and in our study it is defined as the composition of the Cartesian product $R \times \theta$, where $R = \{1, 2\}$ and $\theta = \{0, \pi/4, \pi/2, 3\pi/4\}$.

The three texture features considered are the following

$$\begin{aligned} \text{Entropy} = & -\sum_i P_s(i) \cdot \log(P_s(i)) \\ & -\sum_j P_d(j) \cdot \log(P_d(j)) \end{aligned} \quad (15)$$

$$\text{Contrast} = \sum_j j^2 \cdot P_d(j) \quad (16)$$

$$\text{Homogeneity} = \sum_j (1 + j^2)^{-1} \cdot P_d(j) \quad (17)$$

where P_s and P_d are the normalized SDH. It is important to mention that these texture features are computed using just the lightness component L^* of the RLAB space.

3.3. Lighting

Two lighting features are considered in this study.

- Average lighting (μ_{LR}).
- Probability of Specularity (PS).

These features also are obtained from the RLAB perceptual color space. It is important to note that the luminance component measures the luminous intensity or the lighting in the image. The average lighting is computed as

$$\mu_{LR} = \text{mean}\{L^R(x, y)\} \quad (18)$$

Specularity refers to the amount of light that is emitted by reflection of an object. Commonly a dark image has low Probability of Specularity. It is obtained from the luminance histogram (0–100 in RLAB) and normalized. Thereby the probability density function (pdf) of the luminance is obtained. We considered the cumulative distribution function (cdf) between the 91st and 100th bins,

$$\text{PS} = \sum_{i=91}^{100} P_{LR}(i), \quad (19)$$

where P_{LR} is the luminance histogram and i is the index of the bin in the histogram.

4. Learning the fuzzy rules

The determination of a rule database is a key aspect of the proposed system. These rules are significantly important because they contain the necessary information for taking a decision about the correct algorithm to be applied. In a training protocol, we need to define the linguistic terms and fuzzy sets. Later, the rules are formulated and tuned for improving the selection. Posteriorly, for an image under test they are used in an inference model in order to choose the best algorithm.

The first step for the development of our system consists in the definition of the functional and operational characteristics. Such characteristics include the linguistic terms, the formulation of the rules and the inference model. One set of i inputs $\mathbf{X} = [x_1, x_2, \dots, x_i]^T \in \mathfrak{X}^i$, is considered for the selection, corresponding

Table 2
The twelve features used in the input vector.

Feature	Name	Category
x_1	Number of colors	Color
x_2	Average power spectrum value	Color
x_3	Radius of the EC	Color
x_4	Distance to the EC	Color
x_5	Strength of the EC	Color
x_6	Average chroma	Color
x_7	Probability of a dominant color	Color
x_8	Entropy	Texture
x_9	Contrast	Texture
x_{10}	Homogeneity	Texture
x_{11}	Average lighting	Lighting
x_{12}	Probability of specularity	Lighting

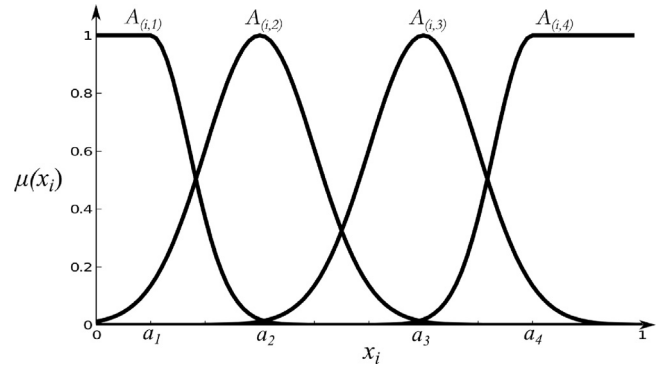


Fig. 3. Distribution of the Gaussian fuzzy sets in a feature domain.

to a feature vector with i elements, where a given element is a quantitative feature. Twelve features (see Table 2) were included in the vector \mathbf{X} : Number of colors (x_1), APSV (x_2), σ of the EC (x_3), D of the EC (x_4), D_σ of the EC (x_5), Average Chroma (x_6), Probability of a dominant color (x_7), Entropy (x_8), Contrast (x_9), Homogeneity (x_{10}), Average lighting (x_{11}), and Probability of Specularity (x_{12}). The output of the system (Ω) takes a value from a set $\Omega = \{\omega_1, \omega_2, \omega_3\}$ corresponding to labels of 3 known classes. The output classes represent the color constancy algorithm considered in the selection: WP (ω_1), GW (ω_2) and GE1 (ω_3).

Once all the features are extracted for the whole training set, these are normalized between 0 and 1, according to the minimal and maximal values registered for each feature. For each training image, its feature vector was extracted. This feature vector was normalized between 0 and 1, according to the minimum and maximum values of each feature during the training stage (see Table 3).

4.1. Definition of linguistic terms

The next step in the learning process, is the definition of the linguistic terms and their corresponding fuzzy sets. From the feature vector \mathbf{X} , each input feature x_i is a value on the domain of the linguistic term, partitioned in a number of fuzzy sets. Next, the definition of the fuzzy sets will be needed.

The definition of fuzzy sets represents one of the most important steps in the design process. According to our experience, linguistic terms should be separated into 4 fuzzy sets in order to obtain a good resolution and achieve an adequate decision capability avoiding an intermediate linguistic term. Fig. 3 shows how four Gaussian fuzzy sets are distributed across the feature domain, $x_i \rightarrow \{0, 1\}$, considering a standard deviation of $\sigma = 0.1$.

We use Gaussian functions, with a_m centers and $\sigma = 0.1$ with $m = \{1, 2, 3, 4\}$. The fuzzy set conformation is done using 4 clusters per feature over each of the linguistic variables. The a_m cluster center, is the maximum degree of membership in the $A_{(i,m)}$ fuzzy set. For specific cases, the $A_{(i,m)}$ labels are replaced by intuitive terms as $A_{(i,1)} = \text{"VERY LOW"}$, $A_{(i,2)} = \text{"LOW"}$, $A_{(i,3)} = \text{"HIGH"}$ and $A_{(i,4)} = \text{"VERY HIGH"}$. For the first Gaussian function $A_{(i,1)}$, its left side remains with membership value of 1 for lower values in the domain of x_i . Similarly occurs for the last function $A_{(i,4)}$, but now to the right side. Table 4 shows the nucleus with maximum degree of membership (center of Gaussians) for each feature.

Table 3
Features and their range computed during the training protocol. The feature vector in the test protocol is normalized according to these values.

Feature	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}
min	1083.00	0.03	5.96	22.78	0.95	1.79	0.08	2.73	1.31	0.26	11.79	0.00
max	62166.00	60.19	32.46	38.97	4.99	49.72	0.88	12.31	434.34	0.88	71.71	0.01

4.2. IF-THEN rules

An important step in the learning process is the knowledge acquisition, also known as rule formulation. In this approach, inputs and outputs are represented by linguistic terms related by an IF-THEN rule. These rules are expressions used in an inference process such that if a set of facts is known (antecedents), an algorithm (consequent) can be inferred. The generic form of the n th of these rules is:

$$R^n : \text{IF } x_1 \text{ is } A_{(1,m)} \text{ AND } x_2 \text{ is } A_{(2,m)} \text{ AND } \dots \\ \text{AND } x_i \text{ is } A_{(i,m)} \text{ THEN } \Omega \text{ is } \omega_p,$$

where, $A_{(i,m)}$ is the m_i fuzzy set in the i th feature for the n th rule, and ω_p represents the p th consequent fuzzy set in the n th rule. R^n is the n th rule that belongs to the set of rules \mathbf{R} .

The methodology for the formulation of the n th rule is the following. Firstly, the corresponding images are identified in the p th class in the training set. Afterwards, the average of each i th feature is calculated for the subset of images previously identified. Thus, a vector with twelve average values is obtained. The definition of the fuzzy set $A_{(i,m)}$ considered in the antecedent of the rules is given by

$$A_{(i,m)} = \max\{\mu_{(i,1)}(x_i), \mu_{(i,2)}(x_i), \dots, \mu_{(i,m)}(x_i)\} \quad (20)$$

where $\mu_{(i,m)}(x_i)$ is the degree of membership of x_i in the fuzzy set. Then, the rule is evaluated in the corresponding m fuzzy set. As an example, a rule for the WP algorithm (ω_1) is stated as:

$$R^1 : \text{IF } x_1 \text{ is } A_{(1,1)} \text{ AND } x_2 \text{ is } A_{(2,1)} \text{ AND } \dots \\ \text{AND } x_{12} \text{ is } A_{(12,2)}, \text{ THEN } \Omega \text{ is } \omega_1,$$

where ω_1 is the consequent fuzzy set in the rule R^1 .

4.3. Fuzzy inference model for algorithm selection

Algorithm selection is the process of mapping a test image into a known color constancy algorithm. In this part of our approach, test images are submitted to the system to compute their feature vectors. Each feature vector is used for the determination of the best algorithm from the already formulated knowledge base through an inference process. This process starts with the estimation of a firing strength $\tau_n(\mathbf{x})$ in each rule generated. Thus, the firing strength represents to what extent \mathbf{x} satisfies the whole set of antecedents.

Table 4
Maximum membership values ($A_{(i,m)}$).

Feature	$A_{(i,1)}$	$A_{(i,2)}$	$A_{(i,3)}$	$A_{(i,4)}$
x_1	0.09	0.20	0.33	0.52
x_2	0.05	0.20	0.37	0.80
x_3	0.06	0.25	0.38	0.49
x_4	0.20	0.25	0.35	0.50
x_5	0.06	0.15	0.25	0.50
x_6	0.05	0.27	0.40	0.50
x_7	0.03	0.18	0.35	0.70
x_8	0.12	0.47	0.65	0.80
x_9	0.06	0.12	0.20	0.39
x_{10}	0.30	0.45	0.60	0.92
x_{11}	0.05	0.30	0.50	0.75
x_{12}	0.00	0.01	0.04	0.86

Using the Mamdani inference model [30], the firing strength $\tau_n(\mathbf{x})$ of a n th rule is computed using

$$\tau_n(\mathbf{x}) = \min[\mu_{(1,m)}^n(x_1), \mu_{(2,m)}^n(x_2), \dots, \mu_{(i,m)}^n(x_i)] \quad (21)$$

or, if the Larsen product inference [31] is used, the firing strength $\tau_n(\mathbf{x})$ is calculated using

$$\tau_n(\mathbf{x}) = \prod_i \mu_{(i,m)}^n(x_i). \quad (22)$$

Finally, the label assigned ω_p to the corresponding output Ω is the one that corresponds to the maximum firing strength from all the rules

$$\Omega = \max[\tau_1(\mathbf{x}), \tau_2(\mathbf{x}), \dots, \tau_n(\mathbf{x})]. \quad (23)$$

Each inference model is considered independently in the experiments, in order to test its performance in the selection, and thus consider it as the best inference model for this task.

4.4. Rules and feature selection

At the beginning of the training process, the automatic selector was designed using three rules, each one for an output algorithm. Images from the corresponding algorithm were included to generate the corresponding rule. However, results showed a very low selection performance, achieving a value close to 30%. This result has two possible explanations. First, features selected are not sufficiently discriminant. Second, there are images with different characteristics that correspond to the same color constancy algorithm, that is, a specific algorithm is suited for a variety of non-similar images. We experimented with a large number of features. However, the twelve before mentioned were chosen due to their higher discrimination capability. For this reason, we considered the second explanation.

Numerous experimental tests were conducted using a heuristic search for determining the number of appropriate rules. As a result, we found that, on one hand, using fewer than five rules provided deficient performance. On the other hand, the inclusion of more than five rules resulted on a negligible improvement in the performance. Consequently, we consider that five rules per algorithm are adequate, enabling us to have a reasonably intuitive control over the information, and exploiting the advantage of handling large amounts of data in a few rules.

The k -means algorithm is used for generating five clusters of features per class in a 12-dimensional space. Thus, a rule of each subset is generated using the rule formulation process. As result, a total of 15 rules are formulated in the learning process.

A sample out of the 15 rules formulated is

$$R^1 : \text{IF } x_1 \text{ is } A_{(1,1)} \text{ AND } x_2 \text{ is } A_{(2,1)} \text{ AND } \dots \text{ AND } x_{12} \text{ is } A_{(12,3)}, \\ \text{THEN } \Omega \text{ is } \omega_1.$$

Table 5
A more practical manner of representing the 15 rules. All premises are included in each antecedent.

R^n	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}	Ω
1	1	1	2	3	2	2	2	2	1	3	2	3	WP
2	1	1	2	4	3	3	4	2	1	2	2	1	WP
3	2	2	4	2	2	4	3	3	1	3	3	1	WP
4	3	4	1	4	4	4	4	4	4	2	4	2	WP
5	4	1	4	2	2	3	2	4	2	1	3	2	WP
6	3	1	4	2	1	2	2	3	2	2	2	1	GW
7	1	1	3	1	1	2	1	2	3	3	2	4	GW
8	1	3	3	3	2	2	2	2	1	3	2	1	GW
9	1	1	2	2	1	1	3	1	1	4	1	2	GW
10	2	1	4	1	1	3	2	3	2	2	2	2	GW
11	2	4	2	4	3	4	4	4	2	3	4	1	GE1
12	3	2	3	4	3	4	2	3	1	2	3	1	GE1
13	1	1	4	3	2	3	3	2	1	3	2	2	GE1
14	4	3	3	1	1	2	2	4	3	1	4	1	GE1
15	3	1	1	4	4	3	3	4	1	1	3	1	GE1

Analyzing the 15 rules with this notation could be a hard task. Table 5 shows a more practical manner of representing the 15 rules, where $A_{(i,j)}$ is shown only with a j index. Each label is represented by a number that can be associated to an intuitive label: 1 to “VERY LOW”, 2 to “LOW”, 3 to “HIGH”, and finally, 4 to “VERY HIGH”. The output set is written out using the algorithm acronym.

4.5. Tuning the rules

The feature space is very large, and the possible number of fuzzy rules covering the whole space is 4^{12} , approximately seventeen million rules. This means that if only 15 rules are considered by the expert system, a huge portion of the feature space is not taken into account. However, if each rule is tuned, with the intention to cover a larger section of the space, the selection performance should increase. In order to extend the coverage of the space by each rule, it is necessary to exclude one or several premises in the antecedent. Thus, the rule formulated is more robust and implicitly includes other rules.

The tuning process was performed through an empirical experimentation. All possible combinations were incorporated into the knowledge base generated by the 15 rules. Finally, the best combination was adopted and, such 15 rules with tuned premises are shown in Table 6. Also, as we can appreciate in Table 6, the included rules show the symbol ‘-’ which represents the excluded premises. A hyphen denotes the exclusion of such fuzzy set from the premises

Table 6
A more practical manner of representing the 15 tuned rules. Some premises were excluded for increasing the rule coverage in the space.

R^n	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}	Ω
1	1	-	2	3	-	2	2	2	-	3	2	3	WP
2	-	-	2	-	-	3	-	-	-	-	2	-	WP
3	-	2	-	2	-	4	-	-	-	-	-	-	WP
4	-	4	-	4	4	-	-	-	4	-	-	-	WP
5	-	1	-	2	2	-	2	4	2	1	3	-	WP
6	3	-	4	2	-	-	-	3	-	2	2	-	GW
7	-	1	3	1	-	2	-	2	-	-	-	-	GW
8	-	3	3	-	-	2	-	2	-	-	2	-	GW
9	-	-	-	-	-	-	-	1	-	4	1	-	GW
10	-	1	4	1	-	-	-	-	-	-	2	-	GW
11	2	4	-	4	3	4	-	4	-	-	-	-	GE1
12	3	-	-	4	-	4	2	3	-	-	3	-	GE1
13	1	1	4	3	-	-	3	2	-	3	2	-	GE1
14	4	3	-	-	-	2	-	-	-	1	-	-	GE1
15	-	1	1	-	4	-	-	-	-	-	3	-	GE1

Table 7
Performance using five “non-optimized” rules per algorithm. Using Larsen inference provides the best selection rate.

Inference	Selection (%)
Mamdani	60.8
Larsen	62.4

of the rule. This “do not care” situation results in the coverage of the whole space of the feature. That is, whatever the specific value of the feature is, the rule is not affected. An example out of the 15 tuned rules is the following

$$R^2 : \text{IF } x_3 \text{ is } A_{(3,2)} \text{ AND } x_6 \text{ is } A_{(6,3)} \text{ AND } x_7 \text{ is } A_{(7,2)}, \text{ THEN } \Omega \text{ is } \omega_1$$

4.6. Performance evaluation

In the evaluation, we used confusion matrices. A confusion matrix is a visualization tool, used widely in supervised learning. Each column of the matrix represents the number or percentage of predictions per class, whereas each row represents the instances of the actual class. The main advantage of this tool is that shows the confusion between classes. An ideal confusion matrix must keep all the information in the main diagonal.

A “success” is given when the system has correctly selected the algorithm that corresponds to an image. Also, this hit must be registered in the main diagonal of the confusion matrix. A “mistake” is one in which the system chooses incorrectly the algorithm. Thus, this selection is recorded in the row of the correct algorithm and the column of the algorithm selected by the system. The success rate is given by $(N_{correct}/N) \times 100$, where $N_{correct}$ is the number of times that an algorithm was correctly chosen, and N is the total number of samples.

5. Experimental results

In this section we present the results obtained from our fuzzy rule-based system. In the testing protocol 429 images from the SFU Laboratory are used. On previous sections it was mentioned that, in random manner, 100 images were taken from the whole set for the training process. It is important to note that the experiments were carried out using two inference models: Mamdani and Larsen. The selection rate obtained in the results using the 15 “non-optimized” rules, is shown in Table 7. Using the Mamdani inference, the system obtains 60.8% correct selection rate. Larsen inference provides a marginal improvement to 62.4%.

Later, experiments were performed using five “optimized” rules per algorithm. Table 8 presents the confusion matrix obtained using Mamdani inference. The confusion matrix using Larsen inference is presented in Table 9. The selection rate of our approach increased to 69.2% using Mamdani inference and to 77.5% using Larsen inference. Now, the difference between them is higher.

Overlapping features is an indicator of high similarity in the images, even among those belonging to different classes. In the

Table 8
Confusion matrix of the selection results using Mamdani inference in the test dataset.

Best algorithm	Predicted algorithm		
	WP	GW	GE1
WP	70.1	15.3	14.6
GW	17.5	74.1	8.5
GE1	36.4	7.8	55.8
Total			69.2%

Bold values highlight the data in the main diagonal in the confusion matrix.

Table 9
Confusion matrix of the selection results using Larsen inference in the test dataset.

Best algorithm	Predicted algorithm		
	WP	GW	GE1
WP	79.6	14.6	5.7
GW	13.2	83.1	3.7
GE1	31.2	9.1	59.7
Total			77.5%

Bold values highlight the data in the main diagonal in the confusion matrix.

evaluation, several test images provide at least one feature that is not included in the feature space created by the rules. It is important to note that, if a feature is outside the range established by the rule, the membership degree tends to zero in the premise. Therefore, the final firing strength, is also almost null. By applying the same process for each rule, the maximum firing strength remains close to zero. Consequently, the selector cannot take properly a decision for choosing an algorithm. For this case of uncertainty, the strongest algorithm (GW) could have been chosen according to the probabilities in the training protocol. However, in this situation, any algorithm chosen provides a very low selection rate. We chose GE1 for being the last and weakest algorithm in the rule evaluation.

5.1. Comparison with other selection methods

The main goal of this work was the development of a fuzzy rule-based selector in a practical and novel task. However, there do not exist specific methods applied to this problem for comparison purposes. For this reason, we only use classical methods for the evaluation of the correct classification/selection rate.

Some classical methods for classification were taken into account for comparison purposes. The k -Nearest Neighbor algorithm (knn) is a non-parametric method used for classification and regression. The input consists of the k closest training examples in the feature space. k -means is another basic method for cluster analysis in data mining. This method aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean, being a prototype of the cluster. A Linear Discriminant Classifier (LDC) is a parametric method based on the value of a linear combination of the characteristics. A Quadratic Discriminant Classifier (QDC) is used to separate measurements of two or more classes of objects or events by a quadric surface. It is a more general version of the LDC method. More details about these methods, can be found in [42,43].

Table 10 shows the results obtained for each selection method. The results show that our approach provides the best rate of selection. The knn algorithm presents a slightly better performance if just one neighbor is used in the analysis instead of three, with 62.4% and 60.5%, respectively. Our selector uses 15 rules (5 per algorithm), thereby the k -means also considers $k = 15$. This method shows a better performance with 64.0%. Moreover, the QDC and the LDC methods achieve a higher performance, with 66.9% and

Table 10
Comparison with other selection methods.

Approach	Selection rate (%)
knn ($nn = 3$)	60.5
knn ($nn = 1$)	62.4
k -means	64.0
QDC	66.9
Fuzzy rule-based system (Mamdani)	69.2
LDC	72.3
Fuzzy rule-based system (Larsen)	77.5

Bold values highlight the data in the main diagonal in the confusion matrix.

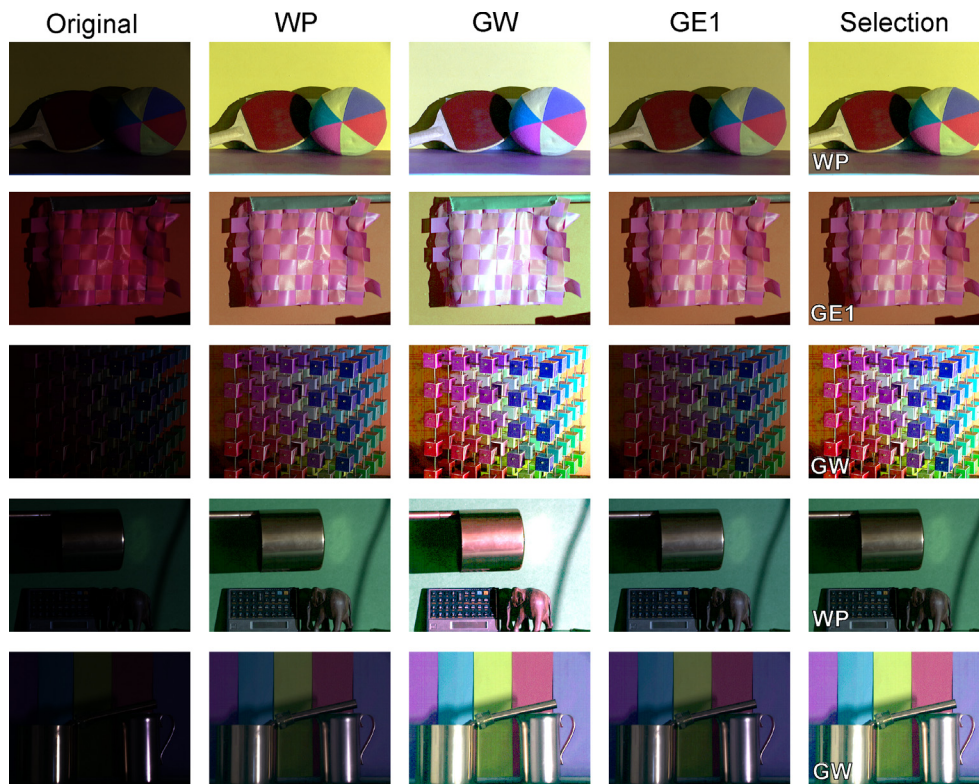


Fig. 4. Five samples out of the 429 used in experiments. These were correctly selected. Top-down: “ball_solux-4100”, “weave_syl-50MR16Q”, “clr_cubes3-solux-4700”, “elephant_solux-4100+3202” and “flcooper_syl-wwf” images. Left to right: Original image, WP outcomes, GW outcomes, GE1 outcomes and the image selected by the system.

72.3%, respectively. Even though the last methods provide a better selection rate than *knn* and *k-means*, our approach using Larsen inference, is the best method, achieving 77.5%. According to this selection rate, we can say that our proposal is reasonably better than the rest of the classifiers. Fig. 4 shows five sample images correctly selected.

Classification or image selection using color constancy algorithms is a hard and challenging task. For example, in an analogous work extracting high-level visual content [28] the success rate was approximately 40% using 87 image features. The purpose of this study was slightly different, oriented to image enhancement issues instead of the illuminant estimation. However, a significant 77.5% selection rate was achieved, with a 95% confidence interval of [74.1, 81.9]. In spite of the fact that we approximately use twenty percent of the images in the training process, instead of the sixty or seventy percent that is usually considered for training data. Another advantage of our approach consists in the use of only twelve features with an intuitive meaning. We think that these features are very important for attaining the acceptable selection rate.

If Table 6 is reviewed, we can draw some conclusions about the contribution of the features in the rules. All features were considered in the rules for the WP algorithm. Although the Number of colors (x_1) or Probability of Specularity (x_{12}) are only taken into account in the first rule. The rules for the GW algorithm do not need the inclusion of Strength of the EC (x_5), Probability of a Dominant Color (x_7), Contrast (x_9) and Probability of Specularity (x_{12}). However, Radius of the EC (x_3) and the Entropy (x_8) are the most significant for this algorithm. Finally, in the case of the GE1 algorithm, the Number of colors (x_1) is apparently important for the rules.

Summarizing, the features with major occurrence in the rules are the Average Power Spectrum Value (x_2), Distance to the EC (x_4), Entropy (x_8) and the Average lighting (x_{11}). We can conclude that

these features are very important for the performance of our system. From the four features, we can highlight two that have not been used in previous studies, the APSV and the Entropy.

6. Conclusions

We presented a framework for developing an automatic selector system, which is oriented to the choice of the best color constancy algorithm for enhancement of dark images. In this study we used three color constancy algorithms: the White-Patch, the Gray-World and the Gray-Edge. These algorithms have been widely used in color constancy tasks due to their simplicity and acceptable performance. However, they have also shown a significant image color enhancement, especially on images under low-light conditions.

The design of an automatic system is not a trivial task when diverse image features are involved in the selection. For this reason, we developed a fuzzy rule-based system, in order to model the information through simple rules. The methodology carried out was divided into two main stages. First, a training protocol for the determination of the fuzzy rules according to features computed from a subset of training images. An important advantage of our approach, consists in the use of twelve image features. Thus, we think that these features are very important for the significant selection rate. Secondly, for a given test image the best algorithm was chosen according to the rule evaluation. Moreover, experiments were conducted in separate manner using Mamdani and Larsen fuzzy inferences. Specifically, the Larsen inference model provided an outstanding selection rate in this specific task.

The main goal of this work has been the development of a fuzzy rule-based selector in a practical and novel task. Besides, our framework is an adequate tool for solving two problems at the same

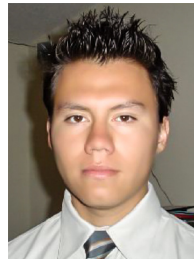
time: color constancy and image color enhancement. Specifically this system should be applied to images or video frames under low light conditions. The defined framework is easy to be replicated for possible posterior issues. Future works could be oriented to the implementation of the automatic selector in practical engineering applications like mobile devices (e.g. cameras and tablets), video surveillance and security tasks.

Acknowledgments

Jonathan Cepeda-Negrete thanks the Mexican National Council on Science and Technology (CONACyT), scholarship 290747 (Grant No. 388681/254884); to the University of Guanajuato (PIFI-2013 program) and the DAIP FJI, for the financial support provided.

References

- [1] J.A. Bernard, Use of a rule-based system for process control, *IEEE Control Syst. Mag.* 8 (5) (1988) 3–13.
- [2] M. Nilashi, O. Ibrahim, A model for detecting customer level intentions to purchase in B2C websites using TOPSIS and fuzzy logic rule-based system, *Arab. J. Sci. Eng.* 39 (3) (2013) 1–16.
- [3] G. Shobha, J. Gubbi, K.S. Raghavan, L.K. Kaushik, M. Palaniswami, A novel fuzzy rule based system for assessment of ground water potability: A case study in South India, *Magnesium (Mg)* 30 (2013) 35–41.
- [4] O. Cordon, M.J. del Jesus, F. Herrera, A proposal on reasoning methods in fuzzy rule based classification systems, *Int. J. Approx. Reason.* 20 (1) (1999) 21–45.
- [5] L.A. Zadeh, Outline of a new approach to the analysis of complex systems and decision processes, *IEEE Trans. Syst. Man Cybern.* 3 (1) (1973) 28–44.
- [6] L.A. Zadeh, Fuzzy sets, *Inf. Control* 8 (1965) 338–353.
- [7] H. Ishibuchi, T. Nakashima, M. Nii, *Classification and Modeling with Linguistic Information Granules*, Springer-Verlag, Secaucus, NJ, USA, 2004.
- [8] K. Trawinski, O. Cordon, L. Sanchez, A. Quirin, A genetic fuzzy linguistic combination method for fuzzy rule-based multiclassifiers, *IEEE Trans. Fuzzy Syst.* 21 (5) (2013) 950–965.
- [9] T. Gevers, A.W.M. Smeulders, PicToSeek: Combining color and shape invariant features for image retrieval, *IEEE Trans. Image Process.* 9 (1) (2000) 102–119.
- [10] M. Schroder, S. Moser, Automatic color correction based on generic content-based image analysis, in: *Proceedings of Color Imaging Conference*, 2001, pp. 41–45.
- [11] F. Gasparini, R. Schettini, Color balancing of digital photos using simple image statistics, *Pattern Recognit.* 37 (6) (2004) 1201–1217.
- [12] J. van de Weijer, C. Schmid, J. Verbeek, Using high-level visual information for color constancy, in: *Proceedings of the International Conference on Computer Vision*, 2007, pp. 1–8.
- [13] S. Bianco, G. Ciocca, C. Cusano, R. Schettini, Improving color constancy using indoor-outdoor image classification, *IEEE Trans. Image Process.* 17 (12) (2008) 2381–2392.
- [14] A. Gijssenij, T. Gevers, Color constancy using natural image statistics and Scene semantics, *IEEE Trans. Pattern Anal. Mach. Intell.* 33 (4) (2011) 687–698.
- [15] J. Yang, R. Stiefelwagen, U. Meier, A. Waibel, Visual tracking for multimodal human computer interaction, in: *Proceedings of the Conference on Human Factors in Computing Systems*, 1998, pp. 140–147.
- [16] M.D. Fairchild, *Color Appearance Models*, 2nd ed., John Wiley & Sons, 2005.
- [17] S. Zeki, *A Vision of the Brain*, John Wiley & Sons, New York, 1993.
- [18] V. Agarwal, B.R. Abidi, A. Koshan, M.A. Abidi, An overview of color constancy algorithms, *J. Pattern Recognit. Res.* 1 (2006) 42–54.
- [19] E.H. Land, The retinex theory of color vision, *Sci. Am.* 237 (6) (1977) 108–128.
- [20] E.H. Land, J.J. McCann, Lightness and retinex theory, *J. Opt. Soc. Am.* 61 (1) (1971) 1–11.
- [21] G. Buchsbaum, A spatial processor model for object colour perception, *J. Frankl. Inst.* 310 (1980) 1–26.
- [22] G.D. Finlayson, E. Trezzi, Shades of gray and colour constancy., in: *Color and Imaging Conference*, vol. 2004, 2004, pp. 37–41.
- [23] J. van de Weijer, T. Gevers, A. Gijssenij, Edge-based color constancy, *IEEE Trans. Image Process.* 16 (9) (2007) 2207–2214.
- [24] A. Chakrabarti, K. Hirakawa, T. Zickler, Color constancy with spatio-spectral statistics, *IEEE Trans. Pattern Anal. Mach. Intell.* 34 (8) (2012) 1509–1519.
- [25] J. Vazquez-Corral, M. Vanrell, R. Baldrich, F. Tous, Color constancy by category correlation, *IEEE Trans. Image Process.* 21 (4) (2012) 1997–2007.
- [26] L. Lucchese, S.K. Mitra, A new class of chromatic filters for color image processing, *IEEE Trans. Image Process. Theor. Appl.* 13 (4) (2004) 534–548.
- [27] E. Provenzi, C. Gatta, M. Fierro, A. Rizzi, A spatially variant white-patch and gray-world method for color image enhancement driven by local contrast, *IEEE Trans. Pattern Anal. Mach. Intell.* 30 (10) (2008) 1757–1770.
- [28] S. Bianco, G. Ciocca, C. Cusano, R. Schettini, Automatic color constancy algorithm selection and combination, *Pattern Recognit.* 43 (3) (2010) 695–705.
- [29] M.M. Faghih, M.E. Moghaddam, Multi-objective optimization based color constancy, *Appl. Soft Comput.* 17 (0) (2014) 52–66.
- [30] E.H. Mamdani, Advances in the linguistic synthesis of fuzzy controllers, *Int. J. Man Mach. Stud.* 8 (6) (1976) 669–678.
- [31] M. Jamshidi, N. Vadiee, T. Ross, *Fuzzy Logic and Control: Software and Hardware Applications*, 2nd ed., Prentice Hall, Englewood Cliffs, NJ, 1993.
- [32] K. Barnard, L. Martin, B. Funt, A. Coath, A data set for colour research, *Color Res. Appl.* 27 (3) (2002) 148–152.
- [33] G.D. Finlayson, S.D. Hordley, C. Lu, M.S. Drew, Removing shadows from images, in: *European Conference on Computer Vision (ECCV 2002)*, 2002, pp. 823–836.
- [34] M. Ebner, *Color Constancy*, John Wiley & Sons, New York, 2007.
- [35] D. Joshi, R. Datta, E. Fedorovskaya, Q.T. Luong, J.Z. Wang, J. Li, et al., Aesthetics and emotions in images, *IEEE Signal Process. Mag.* 28 (5) (2011) 94–115.
- [36] V. Tsagaris, G. Ghistoulas, V. Anastassopoulos, A measure for evaluation of the information content in color images, in: *IEEE International Conference on Image Processing*, vol. 1, 2005, pp. 417–420.
- [37] Y. Zhang, P. An, Q. Zhang, L. Shen, Z. Zhang, A no-reference image quality evaluation based on power spectrum, in: *3DTV Conference: The True Vision: Capture, Transmission and Display of 3D Video (3DTV-CON)*, 2011, pp. 1–4.
- [38] M.D. Fairchild, R.S. Berns, Image color-appearance specification through extension of CIELAB, *Color Res. Appl.* 18 (3) (1993) 178–190.
- [39] M.D. Fairchild, Refinement of the RLAB color space, *Color Res. Appl.* 21 (5) (1996) 338–346.
- [40] R.M. Haralick, K. Shanmugam, I. Dinstein, Textural features for image classification, *IEEE Trans. Syst. Man Cybern.* 3 (6) (1973) 610–621.
- [41] M. Unser, Sum and difference histograms for texture classification, *IEEE Trans. Pattern Anal. Mach. Intell.* 8 (1) (1986) 118–125.
- [42] L.I. Kuncheva, *Fuzzy Classifier Design*, Springer-Verlag, Heidelberg, 2000.
- [43] R.O. Duda, P.E. Hart, D.G. Stork, *Pattern Classification*, John Wiley & Sons, New York, 2012.



Jonathan Cepeda-Negrete received his MEng in Electrical Engineering in 2012 and his BEng in Communications and Electronics in 2011 from the Universidad de Guanajuato, Mexico. He is currently working toward his DEng degree also at the Universidad de Guanajuato. His current research interests include computer vision applications, mainly dark image enhancement using soft computing techniques.



Raul E. Sanchez-Yanez is a Doctor of Science (Optics), concluding his studies at the Centro de Investigaciones en Óptica (CIO), Leon, Mexico, in 2002. He received his MEng Electrical Engineering and BEng in Electronics, from the Universidad de Guanajuato at Salamanca, where he has been a full-time professor since 2003. His main research interests include color and texture analysis for computer vision tasks and the computational intelligence applications in feature extraction and decision making.