Comparison of Feature Selection Schemes for Color Texture Classification

A. Porebski[1,2], N. Vandenbroucke[1,2] and L. Macaire[2]
[1] École d'Ingénieurs du Pas-de-Calais - Campus de la Malassise - 62967 Longuenesse Cedex - FRANCE
e-mail: alice.porebski@eipc.fr, nicolas.vandenbroucke@eipc.fr
[2] Laboratoire LAGIS - FRE CNRS 3303 - Université Lille 1 - Sciences et Technologies - 59655 Villeneuve d'Ascq - FRANCE
e-mail: ludovic.macaire@univ-lille1.fr

Abstract— **In this paper, we propose to compare the performances of two sequential feature selection schemes used for supervised color texture classification. We focus this study on the sequential forward selection (SFS) scheme and the more complex sequential forward floating selection (SFFS) scheme which avoids the "nesting effect". These schemes retain Haralick features extracted from chromatic co-occurrence matrices of images coded in different color spaces. We experimentally study the contribution of these two feature selection schemes with three benchmark color texture databases.**

Keywords— **Color texture, feature selection, SFS, SFFS.**

I. Introduction

A lot of papers have shown that there do not exist color texture features which allow to discriminate all kinds of color texture [15]. That is why we propose to compute a high number of color texture features from images coded in several color spaces. The most discriminating color features are then selected thanks to a sequential feature selection procedure which is achieved during a supervised learning scheme [14].

In this paper, we propose to measure the contribution of two sequential feature selection schemes on the quality of color texture classification. For this purpose, we propose to compare the classification results obtained by operating in a feature space selected by the sequential forward selection (SFS) and those obtained by considering the feature space selected by the sequential forward floating selection (SFFS).

A first study hold by Schenk and al. has shown that using the SFFS for feature selection in on-line hand written letter recognition does not lead to better results than using the simple SFS [21]. We experimentally extend this study to the color texture classification by measuring the contribution of these two feature selection schemes with three benchmark color texture databases.

In the second section of this paper, we present the influence of the choice of color spaces on texture classification. Then, in the third section, the color texture are characterized by Haralick features extracted from chromatic co-occurrence matrices of images coded in several color spaces. The fourth section details the two sequential feature selection schemes which determine low dimensional feature spaces. In the fifth section, the set of images coming from the OuTex, VisTex and BarkTex image databases and used in our experiments are presented. Finally, experimental results show the contribution of these two feature selection schemes for color texture classification.

II. Color spaces and texture classification

The color of pixels can be represented in different color spaces which respect different physical, physiologic, and psycho-visual properties [4]. Figure 1 shows several spaces commonly used in color image analysis. They can be classified into four families: the primary spaces, the luminance-chrominance spaces, the perceptual spaces and the independent color component spaces [4].

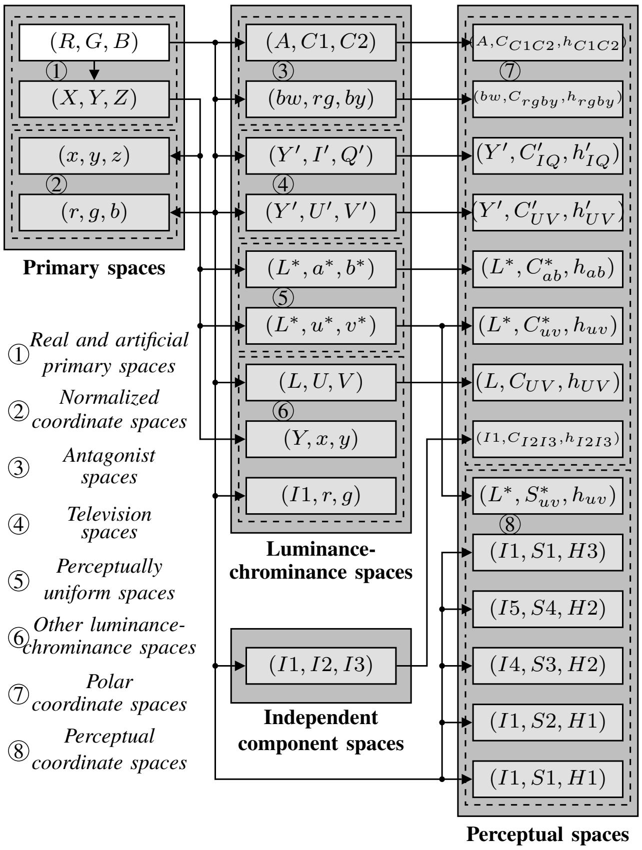

Fig. 1. Color space families.

Many authors have tried to compare the classification performances obtained with different color spaces [5], [10], [8],

[20]. Their goal is to determine the color space which would be the best suited to their color texture classification application. However, the synthesis of these works shows that there is no color space which is adapted to the discrimination of all the textures.

To solve this problem, we have proposed to combine different color spaces in order to improve the performances of color image classification [12]. We associate the informations coming from different color spaces by coding color texture images into several color spaces and by computing some texture features from the so-coded images.

III. Color texture features

For the texture classification purposes, a color texture is described by a set of $\hat{d}$ features and is represented by a point in a $\hat{d}$-dimensional feature space in order to achieve the classification. There exists a large number of color texture descriptors and it is well-known that the performance of the classifier depends on the choice of the color texture features.

In order to represent both the color and the texture informations, three kinds of features are used in the literature:

- "luminance based texture features" mixed with color statistical features [2],
- "within color component texture features" which take into account only the spatial relationships within a single color component (for example within the color component R, G or B) [5], [3],
- "between color component texture features" which consider spatial relationships within and between different color components [10], [1].

Even if the two first approaches seem to be computationally fast, they do not take into account the spatial relationships between color components. So, the information contained in the whole color texture is lost and the classification quality is reduced [1]. That is why several authors privilege features which consider spatial relationships within and between color components. We have shown that the well-known Haralick features extracted from Reduced Size Chromatic Co-occurrence Matrices (RSCCM) are relevant for this purpose [14]. That is why, we choose to consider these features, which are easily implementable, in order to carry out our experimental study.

RSCCMs are $N \times N$ co-occurrence matrices where the quantization level N of the color components is reduced in order to decrease the memory storage cost and so, the time required to extract features from these matrices. In a previous study, we have shown that when N is set to 16, we obtain satisfying classification results while significantly reducing the processing time [14].

Each color image $\mathbf{I}$, coded in a (C_1,C_2,C_3) color space of Fig 1, is characterized by the $N_M = 6$ following RSCCMs:

- three within color component matrices: $m^{C_1,C_1}[\mathbf{I}]$, $m^{C_2,C_2}[\mathbf{I}]$ and $m^{C_3,C_3}[\mathbf{I}]$.
- three between color component ones: $m^{C_1,C_2}[\mathbf{I}]$, $m^{C_1,C_3}[\mathbf{I}]$ and $m^{C_2,C_3}[\mathbf{I}]$.

It has been proved that the choice of the neighborhood shape used to compute RSCCM depends on the analyzed textures [13]. When textures do not present any privileged direction, it is necessary to compute features which take into account all the possible directions. That is why we choose to use the 3x3 isotropic 8-neighborhood.

Usually, RSCCMs are not directly exploited because they contain a large amount of information. To reduce it, while preserving the relevance of these descriptors, we use the first $N_H = 13$ Haralick features denoted f_1 to f_{13} extracted from theses matrices [6].

In order to exploit all the properties of the N_S color spaces, each image is firstly coded in each of these color spaces. Then, for each color space, the $N_M = 6$ RSCCMs are computed and the $N_H = 13$ Haralick features are extracted from each matrix. A color texture is firstly represented by $N_f = N_H \times N_M \times N_S = 13 \times 6 \times 28 = 2184$ candidate color texture features (see. Fig. 2).

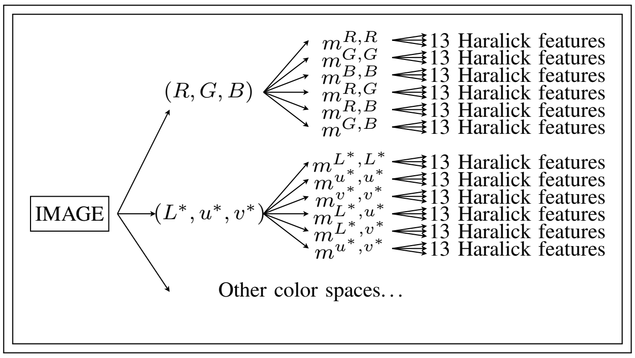

Fig. 2. Candidate color texture features.

It is well-known that the performances of the classifier - in terms of rates of well-classified textures and computation time - depends on the dimension of the feature space. As the dimension of our candidate color texture feature space is too high, it is necessary to select the most discriminating color texture features during a supervised learning stage. For this purpose, we choose to use a sequential selection procedure associated with a measure of the discriminating power of each candidate color texture feature space.

In this paper, we propose to select the most discriminating features thanks to two different selection approaches (SFS and SFFS, detailed in the next section) and to experimentally compare the performances reached by each of these procedures.

IV. Feature selection

Our supervised classification scheme is divided into two successive stages:

- A supervised procedure selects a low number of discriminating texture features among a set of candidate ones in order to build a low-dimensional feature space. During this stage, the classifier is trained to partition this feature space.
- A decision stage where each examined color texture is represented by a point in the so-selected feature space in order to be classified.

For our experimental study, the examined texture images are equally divided into two sets according to the Holdout method: half of the images are used to build the training set, while the other images are used to test the performance reached by the classifier. These test images are also used to determine the dimension of the selected feature space.

The determination of the most discriminating feature space is achieved thanks to a non-exhaustive iterative selection procedure. In this paper, we propose to compare the performances reached by two sequential procedures: SFS and SFFS.

a) SFS: The SFS algorithm, whose pseudo code is described in Fig. 3, consists in iteratively adding features from a set which is initially empty [16]:

- The initial input data are E, the set of the N_f candidate features X_l ($l = 1, \ldots, N_f$) and J, the discriminating power measure. In this illustration, we choose to maximize this measure to obtain the best discrimination.
- As the iterative selection procedure is a "Forward" type algorithm, the evaluated feature subspace is initially empty ($E_0 = \varnothing$).
- At each step s of the SFS procedure, we consider the $(N_f - s)$ feature subsets $E_s \cup \{X_l\}$, composed of the s features already selected and one of the available features X_l. The available features are those which have not been yet selected, i.e. those which belong to the subset $E \backslash E_s$. The discriminating power $J(E_s \cup \{X_l\})$ of each of these subsets is then evaluated thanks to the discriminating measure J defined by eq. (1). The feature X_l which maximizes $J(E_s \cup \{X_l\})$ is denoted X_+. It is associated with the subspace E_s of the already selected features to form the discriminating $s+1$-dimensional subspace E_{s+1}. The value of s is then incremented ($s = s + 1$).
- Once the stopping criterion is satisfied, the selection procedure stops and E_s is the selected feature space $E_{\hat{d}}$.

A known drawback of the SFS procedure is its monotonic growing feature set. Indeed, once a feature is added, it cannot be removed. This involves the “nesting effect” [17].

b) SFFS: The SFFS algorithm allows to avoid the “nesting effect” by removing at each step s of the procedure one or several selected features while this removal allows to improve the discriminating power [18]:

- First, the initial conditions are the same than the SFS ones.
- The "Forward" step corresponds to a SFS procedure.

Data
$E = \{X_l, l = 1, \ldots, N_f\}$, the N_f candidate features,
J, the discriminating power measure,
$E_d = \{Y_k \in E, k = 1, \ldots, d\}$, the evaluated d-dimensional feature subspace ($d \leq N_f$).

Initialization
$s = 0$
$E_0 = \varnothing$

Do
$X_+ = argmax_{X_l \in E \backslash E_s} J(E_s \cup \{X_l\})$
$E_{s+1} = E_s \cup \{X_+\}$, $Y_{s+1} = X_+$
$s = s + 1$
While $s < d_{max}$
$E_{\hat{d}} = E_s$

Fig. 3. SFS Algorithm.

- The selected feature X_+ influences the rest of the procedure. To avoid local optima, it is necessary to apply one or several "Backward" steps. This step consists in removing a selected feature Y_k from the subspace E_s. The corresponding discriminating power $J(E_s \backslash \{Y_k\})$ is then measured. This operation is done for each feature Y_k ($k = 1, \ldots, s$) of the subspace E_s. We thus denote X_-, the feature Y_k which maximizes $J(E_s \backslash \{Y_k\})$. If $J(E_s \backslash \{X_-\})$ is higher than $J(E_{s-1})$, the subspace selected at the step $s - 1$ becomes $E_s \backslash \{X_-\}$ and we start a new "Backward" step until the feature removal does not improve the discrimination, i.e. until $J(E_s \backslash \{X_-\}) \leq J(E_{s-1})$. When the backward steps are finished, the stopping criterion is evaluated. If it is not satisfied, we start a new "Forward" step else the selection procedure stops and E_s is the selected feature space $E_{\hat{d}}$.

The SFFS algorithm is summarized in Fig. 4.

In order to only select color texture features which are not correlated, we measure, at each “Forward” step, the correlation level between each of the available color texture features and each of the $(s-1)$ other color texture features constituting the previously selected $(s-1)$-dimensional space. The considered features are selected as candidate ones only if their correlation level with the already selected color texture features is lower than a threshold fixed by the user [19].

We assume that the more the clusters associated with the different texture classes are well-separated and compact in the candidate feature space, the higher the discriminating power of the selected color texture features is. That leads us to use measures of class separability and class compactness as measures of the discriminating power. That is the reason why we choose to minimize the Wilks’ criterion:

$$J = \frac{|\Sigma_W|}{|\Sigma_W + \Sigma_B|}, \tag{1}$$

Data
$E = \{X_l, l = 1, \ldots, N_f\}$, the N_f candidate features,
J, the discriminating power measure,
$E_d = \{Y_k \in E, k = 1, \ldots, d\}$, the evaluated d-dimensional feature subspace ($d \leq N_f$).

Initialization
$s = 0$
$E_0 = \varnothing$

Step 1 (Forward) **:**
$X_+ = argmax_{X_l \in E \setminus E_s} J(E_s \cup \{X_l\})$
$E_{s+1} = E_s \cup \{X_+\}$, $Y_{s+1} = X_+$
$s = s + 1$

Step 2 (Backward) **:**
$X_- = argmax_{Y_k \in E_s} J(E_s \setminus \{Y_k\})$
If $J(E_s \setminus \{X_-\}) > J(E_{s-1})$ **Then**
$E_{s-1} = E_s \setminus \{X_-\}$
$s = s - 1$
Go to **Step 2**
Else
If the stopping criterion is satisfied **Then**
$E_{\hat{d}} = E_s$
Else
Go to **Step 1**

Fig. 4. SFFS Algorithm.

where Σ_W is the the within-class dispersion matrix which characterizes the compactness of each texture class, Σ_B is the between-class dispersion matrix which measures the class separability and $|M|$ is the determinant of the matrix M.

The selection procedure builds d_{max} d-dimensional pre-selected color texture feature spaces. In order to determine the dimension $\hat{d}$ of the most discriminating feature space, we propose to measure the rate T_d ($1 \leq d \leq d_{max}$) of well-classified test images obtained with each d-dimensional pre-selected space.

We choose to classify the images of the test set thanks to the simple and widely used 1-NN classifier. The selected color texture feature space is thus the pre-selected space for which T_d is maximum:

$$\hat{d} = \operatorname*{argmax}_{d=1}^{d_{max}} T_d. \quad (2)$$

V. Texture databases

In this section, we present three well-known and largely used benchmark color texture databases used in our experiments and more precisely, three different benchmark texture sets, coming from OuTex, VisTex and BarkTex databases, respectively [9], [11], [7].

A. OuTex database

Outex contains a very large number of surface textures acquired under controlled conditions by a 3-CCD digital color camera. To build the Outex set, each of 68 textures of this database is split up from an original image into disjoint sub-images whose size is 128×128 pixels. Since the original image size is 746×538 pixels, this makes a total of 20 sub-images by texture. Figure 5 illustrates each texture class by one image.

Fig. 5. Example of OuTex color textures: each image represents a class of texture.

The examined dataset is available at the OuTex web site (`http://www.outex.oulu.fi/temp/`) as test suite Contrib-TC-00013. Among the 1360 images of the Outex set, 680 images are used for the training set and 680 for the test set.

The illuminant which have been considered to code the color components of the different color spaces of Fig. 1 is the illuminant CIE A [8].

B. VisTex database

VisTex is a large set of natural color textures acquired under non-controlled conditions by different color cameras. To build the VisTex set, each of 54 textures of this database is split up from an original image into disjoint sub-images whose size is 128×128 pixels. Since the original image size is 512×512 pixels, this makes a total of 16 sub-images by a texture (see Fig. 6).

This set is available at the OuTex web site (`http://www.outex.oulu.fi/temp/`) as test suite Contrib-TC-00006. Among the 864 images of the VisTex set, 432 images are used for the training set and 432 for the test set.

Here, the illuminant which have been considered to code the color components of the different color spaces of Fig. 1 is the illuminant D65 [8].

Fig. 6. Example of VisTex color textures: each image represents a class of texture.

C. BarkTex database

Color images of the BarkTex database are equally divided into six tree bark classes, with 68 images by class. The size of each image is 256×384 pixels.

To build the BarkTex set, a region of interest, centered on the bark and whose size is 128×128 pixels, is firstly defined. Then, four sub-images whose size is 64×64 pixels are extracted from each region. We thus obtain a set of $68 \times 4 = 272$ sub-images by a class (see Fig. 7).

Fig. 7. Example of BarkTex color textures: each column represents a class of texture.

Among the 1632 images of the BarkTex set, 816 images are used for the training set and 816 for the test set.

As the natural color textures of the BarkTex database are acquired under non-controlled conditions, the illuminant which have been considered to code the color components of the different color spaces is also the illuminant D65.

VI. Experimental results

The SFS procedure is known to be less expensive in computation time than the SFFS one. However, the SFFS procedure avoids the "nesting effect" as it allows "backtracking" during the selection. It thus seems to be more efficient than the SFS procedure. That is what we propose to verify in this section by comparing the classification results obtained with these two sequential selection procedures.

Figures 8, 9 and 10 show the rates of well-classified test images reached by considering the SFS and SFFS schemes. These rates T, obtained according to the dimension d of the pre-selected feature space, are presented with the OuTex, VisTex and BarkTex databases, respectively.

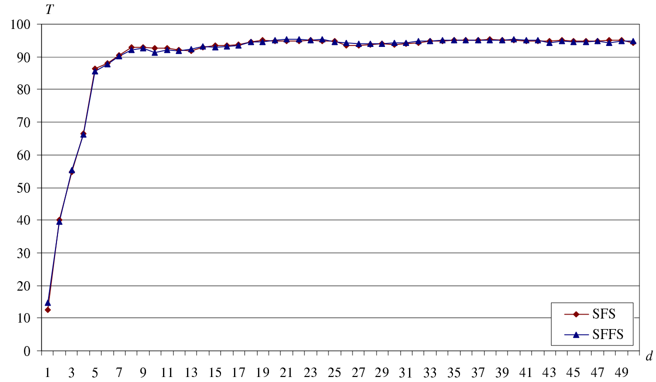

Fig. 8. Rates T of well-classified OuTex test images, according to the dimension d of the pre-selected feature space, considering the SFS and SFFS procedures.

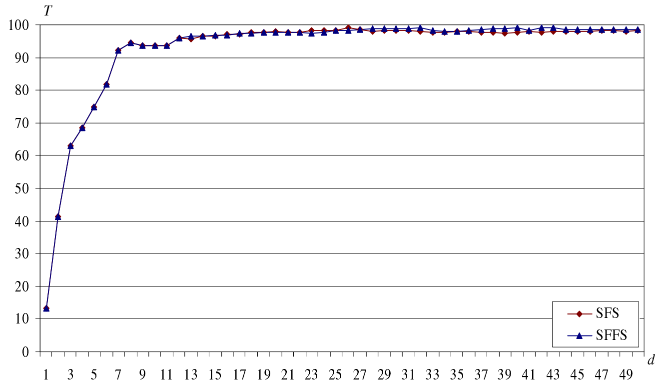

Fig. 9. Rates T of well-classified VisTex test images, according to the dimension d of the pre-selected feature space, considering the SFS and SFFS procedures.

We notice that the rates of well-classified test images reached with each of these two selection procedures are quite similar whatever the value of d, particularly for the OuTex and VisTex images. This conclusion concurs with the fact that the first selected features are the same whatever the used selection scheme: for the OuTex database, the three first selected features are the same and for the VisTex images, the five first selected features are identical.

For the BarkTex database, only the single first selected feature is shared by the two selection approaches. This concurs with the classification results: when d is lower than 5, the SFS procedure allows to obtain better classification results and when d is higher or equal to 5, the results obtained by

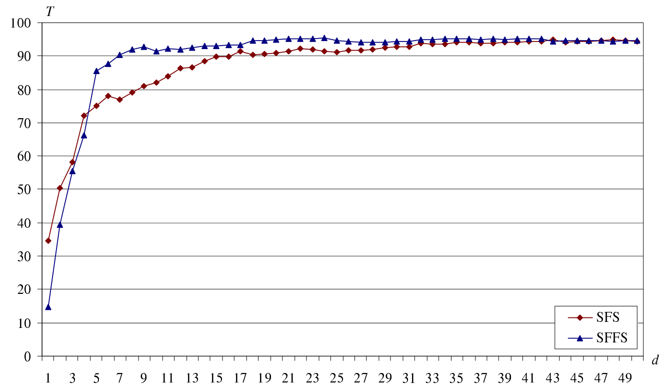

Fig. 10. Rates T of well-classified BarkTex test images, according to the dimension d of the pre-selected feature space, considering the SFS and SFFS procedures.

considering the SFFS procedure are better than those reached with SFS. However, this difference tends to decrease when d increases.

We have shown that these conclusions about the comparison between the SFS and SFFS approaches are quite similar when the Trace criterion is considered instead of the Wilk's ones.

For the OuTex database, the best classification rates reach 95.44% whether the SFS ($\hat{d} = 38$) or the SFFS ($\hat{d} = 24$) procedure is considered. The SFFS procedure selects a discriminating feature space with a lower dimension than the one selected thanks to the SFS procedure, while keeping the same performances.

With the VisTex images, the best classification rates are also the same. They reach 99.07% with the SFS ($\hat{d} = 26$) and the SFFS ($\hat{d} = 32$) procedures. In this case, it is the dimension of the subspace selected by considering the SFS approach which is lower than that selected by SFFS.

Finally, for the BarkTex database, the best classification rates reach 93.87% ($\hat{d} = 50$) by considering the SFS procedure and 93.38% ($\hat{d} = 50$) with the SFFS ones.

VII. Conclusion

The using of the complex SFFS procedure does not improve the classification results with regard to the basic SFS algorithm. This conclusion concurs with that provided by Schenk and al., who compare the handwritten recognition performances reached by considering each of these two selection procedures [21].

Moreover, the SFS approach does not allow "backtracking" during the selection. It is thus less expensive in processing time: the time required to select the $d_{max} = 50$ discriminating features from the color textures of the OuTex database is 576 s when the SFS procedure is considered, and 1 391 s when the SFFS procedure is performed. The sequential SFS procedure allows thus to offer a satisfying compromise between classification results and processing time.

References

[1] V. Arvis, C. Debain, M. Berducat and A. Benassi, Generalization of the cooccurrence matrix for colour images: application to colour texture classification, Image Analysis and Stereology, vol. 23, pp. 63-72, 2004.
[2] F. Lopez, J.M. Valiente, J.M. Prats and A. Ferrer, Performance evaluation of soft color texture descriptors for surface grading using experimental design and logistic regression, Pattern Recognition, vol. 41, nb. 5, pp. 1761-1772, 2008.
[3] M.A. Akhloufi, X. Maldague and W.B. Larbi, A New Color-Texture Approach for Industrial Products Inspection, Journal of Multimedia, vol. 3, nb. 3, pp. 44-50, 2008.
[4] L. Busin, N. Vandenbroucke, and L. Macaire, Color spaces and image segmentation, Advances in Imaging and Electron Physics, vol. 151, ch. 2, pp. 65-168, 2008.
[5] A. Drimbarean and P.-F. Whelan, Experiments in colour texture analysis, Pattern Recognition Letters, vol. 22, pp. 1161-1167, 2001.
[6] R. Haralick, K. Shanmugan and I. Dinstein, Textural features for image classification, IEEE Transactions on Systems, Man and Cybernetics, vol. 3, pp. 610-621, 1973.
[7] R. Lakmann, BarkTex benchmark database of color textured images, Koblenz-Landau University, ftp://ftphost.uni-koblenz.de/outgoing/vision/Lakmann/BarkTex.
[8] T. Mäenpää and M. Pietikäinen, Classification with color and texture: jointly or separately?, Pattern Recognition, vol. 37, pp. 1629-1640, 2004.
[9] T. Ojala, T. Mäenpää, M. Pietikäinen, J. Viertola, J. Kyllönen and S. Huovinen, Outex new framework for empirical evaluation of texture analysis algorithms, In Proceedings of the 16th International Conference on Pattern Recognition (ICPR'02), Quebec, Canada, vol. 1, pp. 701-706, 2002, http://www.outex.oulu.fi/temp/.
[10] C. Palm, Color texture classification by integrative co-occurrence matrices, Pattern Recognition, vol. 37, pp. 965-976, 2004.
[11] R. Picard, C. Graczyk, S. Mann, J. Wachman, L. Picard and L. Campbell, VisTex benchmark database of color textured images, Media Laboratory, Massachusetts Institute of Technology (MIT), Cambridge, http://vismod.media.mit.edu/pub/VisTex/VisTex.tar.gz.
[12] A. Porebski, N. Vandenbroucke and L. Macaire, Iterative feature selection for color texture classification, Proceedings of the 14th IEEE International Conference on Image Processing (ICIP'07), San Antonio, Texas, USA, pp. 509-512, 2007.
[13] A. Porebski, N. Vandenbroucke and L. Macaire, Neighborhood and Haralick feature extraction for color texture analysis, In Proceedings of the 4th European Conference on Colour in Graphics, Imaging and Vision (CGIV'08), Terrassa, Spain, pp. 316-321, 2008.
[14] A. Porebski, N. Vandenbroucke and L. Macaire, Selection of color texture features from Reduced Size Chromatic Co-occurrence Matrices, In Proceedings of the IEEE International Conference on Signal and Image Processing Applications (ICSIPA'09), Malaysia, 2009.
[15] E.L. Van Den Broek and E.M. Van Rikxoort, Evaluation of color representation for texture analysis, In Proceedings of the Belgium-Dutch Conference on Artificial Intelligence (BNAIC'04), Groningen, The Netherlands, pp. 35-42, 2004.
[16] M. Kudo and J. Sklansky, Comparison of algorithms that select features for pattern classifiers, Pattern Recognition, vol. 33, nb. 1, pp. 25-41, 2000.
[17] A. Jain and D. Zongker, Feature selection: evaluation, application and small sample performance, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 19, nb. 2, pp. 153-158, 1997.
[18] P. Pudil, J. Novovicova and J. Kittler, Floating search methods in feature selection, Pattern Recognition Letters, vol. 15, nb. 11, pp. 1119-1125, 1994.
[19] N. Vandenbroucke, L. Macaire and J.-G. Postaire, Color image segmentation by pixel classification in an adapted hybrid color space. Application to soccer image analysis, Computer Vision and Image Understanding, vol. 90, pp. 190-216, 2003.
[20] Q. Xu, J. Yang and S. Ding, Color texture analysis using the wavelet-based hidden Markov model, Pattern Recognition Letters, vol. 26, pp. 1710-1719, 2005.
[21] J. Schenk, M. Kaiser and G. Rigoll, Selecting Features in On-Line Handwritten Whiteboard Note Recognition: SFS or SFFS?, In Proceedings of the 10th IEEE International Conference on Document Analysis and Recognition, ICDAR 2009, Barcelona, Spain, pp. 1251-1254, 2009.