Theoretical and experimental comparison of different approaches for colour texture classification

Francesco Bianconi*
Università degli Studi di Perugia
Dipartimento Ingegneria Industriale
Via G. Duranti, 67
06125 Perugia (ITALY)

Richard Harvey, Paul Southam
University of East Anglia
School of Computing Sciences
NR4 7TJ Norwich (UK)

Antonio Fernández
Universidade de Vigo
Escuela Técnica Superior de Ingeniería Industrial
Campus Universitario - 36310 Vigo (SPAIN)

(Dated: July 11, 2011)

Colour texture classification has been an area of intensive research activity. From the very onset, approaches to combining colour and texture have been the subject of much discussion, and, in particular, whether they should be considered joint or separately. In this paper we present a comprehensive comparison of the most prominent approaches both from a theoretical and experimental standpoint. The main contributions of the manuscript are: 1) the establishment of a generic and extensible framework to classify methods for colour texture classification on a mathematical basis, and, 2) a theoretical and experimental comparison of the most salient existing methods. Starting from an extensive set of experiments based on the Outex dataset we highlight those texture descriptors which provide good accuracy along with low dimensionality. The results suggest that separate colour and texture processing is the best practice when one seeks for optimal compromise between accuracy and limited number of features. We believe that the paper may serve as a guide for those who need to choose the appropriate method for a specific application, as well as a basis for the development of new methods.

PACS numbers: 07.05.Pj, 42.30.Sy

I. INTRODUCTION

Texture analysis has emerged as one of the most prominent topics in computer vision. Since the pioneering work of Haralick [25], Julesz [33], and Daugman [15], which date back to the mid 70's and early 80's, numerous solutions have been proposed. This large set of methods has been recently referred to as a "galaxy of texture features" [96]. Originally texture analysis was performed on grayscale images, thus discarding colour information. In the mid 90's various authors started incorporating colour data into texture analysis. That this can improve classification accuracy – at least under steady illumination conditions – has been put in evidence by various authors, most remarkably by Drimbarean and Whelan [17] and, later on, by Mäenpää and Pietikäinen [59]. It has long been debated, however, about which is the best method to combine colour and texture, and, in particular, whether they should be considered jointly or separately [17, 59]. Recent advances in psychophysiology have brought up interesting ideas on this debate. Experiments conducted by Cant *et al.* [10] showed that the form of objects and their surface properties (colour + texture) are processed independently in the brain. Moreover colour and texture seem to be processed independently too, since people who participated in the experiment were shown to be able to ignore colour while making texture classifications and vice versa. These results seem te be corroborated by the work of Cavina-Pratesi *et al.* [12], where they show that colour and texture are processed in different zones of the medial occipitotemporal cortex.

Within texture analysis, classification is concerned with the problem of assigning a sample image to one of a set of known texture classes [25]. This plays an important role in a very ample range of applications. In industry a common use is related to sorting products into lots of similar visual appearance [19, 38, 51], a problem usually referred to as *surface grading*. Medicine is another field where classification comes into its own: breast [55] and skin [97] cancer detection, histopathological image analysis [78], classification of endoscopic images [35, 48] and detection of haematological malignancies [86] are just some examples of possible applications. Other fields of

*Performed this work as a visiting researcher in the School of Computing Sciences, University of East Anglia, UK

application include (but are not limited to): analysis and classification of industrially prepared foods [57, 58], terrain [68, 92] and rock [47] classification, seabed characterization [72], and crowd monitoring [54].

The combination of colour and texture for image classification has been dealt with in several ways, but has so far been lacking a general and comprehensive framework to classify the different methods on a mathematical basis, with the exception of the work of Palm [69], of which we shall discuss in Section II. In this paper we present a theoretical and experimental comparison of these methods that aims at: 1) defining a classification scheme (taxonomy) based on a firm mathematical foundation; 2) guiding scientists and practitioners in choosing the appropriate method for specific applications; 3) giving a base for the development of new methods.

A review of the literature shows that various authors have approached colour texture classification focusing on different aspects of the problem. Drimbarean and Whelan [17] examined two alternatives to extract colour texture features: parallel processing of each colour channel, and disjoint processing of colour and texture through previous separation of luminance from chrominance. In both cases their results show that the use of colour improves classification accuracy. The texture descriptors included in the experiments are Discrete Cosine Transform, Gabor filters and co-occurrence matrices.

Mäenpää and Pietikäinen [59] compared sets of colour indexing methods, grayscale methods and colour texture methods both in static and variable illumination conditions. The results confirm the findings in [17], but only under static illumination conditions. Moreover they show that colour texture descriptors are outperformed by either colour or grayscale texture alone, and thus casting doubts on the use of joint colour texture descriptors in general.

Iakovidis *et al.* [30] comparatively evaluated the performance of four methods for colour texture classification, namely: Opponent Colour Local Binary Patterns; chromaticity moments; wavelet correlation signatures and wavelet covariance features. Their results show that, in general, colour improves texture classification, even if, in some cases, grayscale features alone can give comparable results as well.

Kurmyshev *et al.* [42, 43] evaluated three custom strategies to combine textural and colour information at classification level. In all approaches textural data are extracted from the luminance plane, whereas colour features are computed from the chrominance plane. The experiments, limited to the Coordinated Cluster Representation, show that the three strategies perform the same in colour texture classification, suggesting that the approach of treating texture and color feature separately produces more or less the same results, no matter the way we combine them.

In this paper we wish to approach the problem from a different direction. As a first step we define, on a mathematical basis, a taxonomy for classifying colour texture descriptors (Section II). Subsequently we discuss some of the most prominent approaches and define them within the taxonomy (Section III). Then we present a comparative experiment based on a standard colour texture dataset (Section IV). Finally we discuss the results of the experiments (Section V) and report some concluding considerations (Section VI).

II. COLOUR TEXTURE CLASSIFICATION: A TAXONOMY

The classification scheme proposed by Palm [69] defines three groups: *parallel*, *sequential* and *integrative*. In the parallel approach colour and texture are handled separately; in the sequential approach the colours are processed to produce a single channel image which can then be analysed by conventional texture approaches. The integrative systems attempt to process colour and texture jointly either as a single or as multiple channels. While this taxonomy provides a useful conceptual framework, it would benefit from some new definitions which we now propose. We start with the definition of colour texture function.

Definition 1 *A* colour texture function (CTF) *is a function F which maps a set of vectors $\mathbf{v}_i$ into a vector $\mathbf{f}$:*

$$\mathbf{f} = F\left(\mathbf{v}_1, \cdots, \mathbf{v}_i, \ldots, \mathbf{v}_N\right) \tag{1}$$

where:

$$\mathbf{v}_i = (c_1, \cdots, c_K, x_1, \cdots, x_D) \tag{2}$$

In the pattern recognition jargon $\mathbf{f}$ is usually referred to as the *feature vector* [77]. In Eq. 2 $\{c_1, \cdots, c_K\}$ denote the *spectral* coordinates and $\{x_1, \cdots, x_D\}$ the *spatial* coordinates. They represent the energy content in each spectral band and the position of each vector in a Euclidean space respectively. Consequently K is the dimension of the spectral domain and D that of the spatial domain. In this paper we limit our comparative review to $K = 3$ and $D = 2$, which correspond to the domain of standard colour images. Hence (2) becomes:

$$\mathbf{v}_i = (c_1, c_2, c_3, x_1, x_2) \tag{3}$$

We can now give the definition of a colour texture descriptor.

Definition 2 *A* colour texture descriptor (CTD) *is a non-empty set of colour texture functions.*

It is also useful to define, in this framework, a colour image:

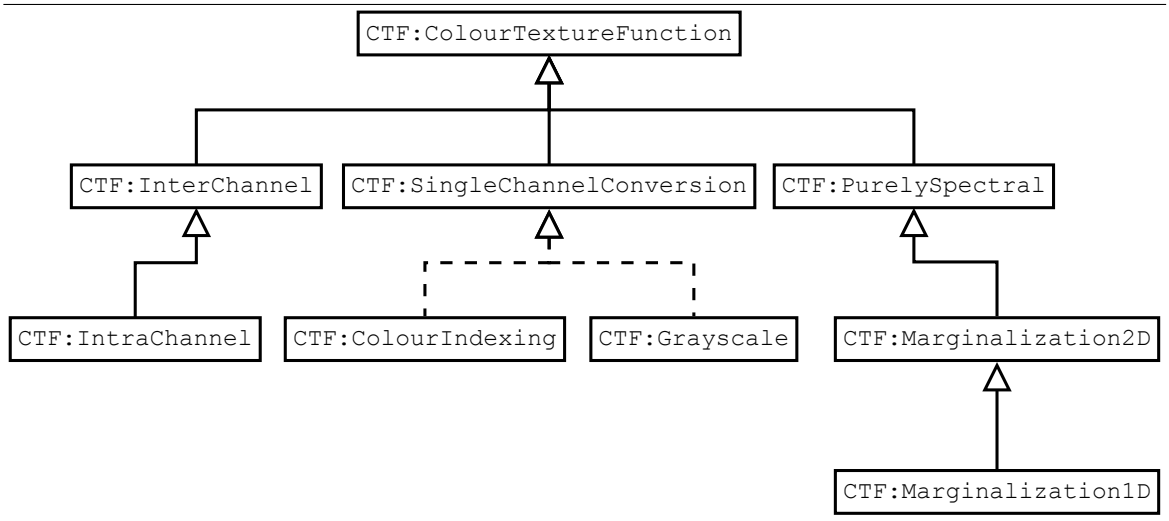


FIG. 1: Taxonomy of colour texture functions

Definition 3 *A* colour image *is a finite set of vectors as they are defined in (3):*

$$\mathbf{I} = \{\mathbf{v}_1, \cdots, \mathbf{v}_i, \cdots, \mathbf{v}_N\} \tag{4}$$

We normally refer to the vectors $\mathbf{v}_i$ as *pixels*. The above expression can be conveniently unfolded and written in the following equivalent form:

$$\mathbf{I} = \begin{bmatrix} c_{11} & c_{21} & c_{31} & x_{11} & x_{21} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ c_{1i} & c_{2i} & c_{3i} & x_{1i} & x_{2i} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ c_{1N} & c_{2N} & c_{3N} & x_{1N} & x_{2N} \end{bmatrix} \tag{5}$$

Different approaches to colour texture analysis can be classified according to the the form assumed by the function F. In the remainder of this section, to simplify the notation, we drop the unnecessary dependence on the index i, so that the above expression becomes:

$$\mathbf{f} = F\left(c_1, c_2, c_3, x_1, x_2\right) \tag{6}$$

with the convention that $(c_1, c_2, c_3, x_1, x_2)$ represents a set of vectors in the five-dimensional spectral-spatial space.

In the following subsections we define two taxonomies: one for colour texture functions and the other for colour texture descriptors. The former is based on *inheritance* and serves as a basis for the latter, which is based on *composition*. We recall that inheritance implements an "is a" relation (for instance: a `Car` is a `Vehicle`, so we say that the *subclass* `Car` *inherits* from the *super-class* `Vehicle`), whereas composition implements a "has a" relation (for instance: a `Car` has an `Engine`). To this we add that a particular form of inheritance called *restriction inheritance* (or *inheritance by restriction*) is used herein [61]. This means that the sub-class is obtained by imposing some constraints on the variable part of the super-class (e.g: a sub-class `Circle` can be derived by a super-class `Ellipse` by imposing that the two axes of the ellipse are equal). Through this taxonomy we define a set of classes that makes it possible to consider the methods proposed in literature as instances of these classes. Each class is uniquely identified through a label of the following type: `<namespace>:<class_name>`, where `<namespace>` can be either `CTF` for colour texture functions or `CTD` for colour texture descriptors, and `<class_name>` is an acronym descriptive of the class.

A. Taxonomy of colour texture functions

This taxonomy is organized in a hierarchical way. As mentioned in the preceding section, the sub-classes are derived from the parent classes through restriction inheritance. In this specific case the sub-classes are characterized through certain constraints about the form that the function F can assume. So, whereas in the base class the form of the function F is as generic as possible, in the derived classes it is more and more specific as subclassing goes on. An important feature of such a structure, is that, being based on inheritance, it can be easily extended in the future. The taxonomy is illustrated in Figure 1, where we adopted the standard UML specification [85] to represent the classes. This convention is maintained throughout the entire paper. The base class `CTF:ColourTextureFunction` represents a generic function of the type in (6) without any restriction about the form of F.

1. *Inter-channel* (`CTF:InterChannel`)

Inter-channel colour texture functions depend on the spatial coordinates and on a pair of spectral coordinates. The characteristic function of this class can be expressed as follows:

$$\mathbf{f} = F\left(c_h, c_k, x_1, x_2\right) \tag{7}$$

for $h, k \in \{1, \cdots, 3\}, h \neq k$

Intra-channel (`CTF:IntraChannel`) Intra-channel colour texture functions depend on the spatial coordinates and on one spectral coordinate only. They have the following form:

$$\mathbf{f} = F\left(c_h, x_1, x_2\right) \tag{8}$$

for $h \in \{1, \cdots, 3\}$. Therefore this class inherits from `CTF:InterChannel` with the additional constraint that only one spectral coordinate is retained.

2. *Single channel conversion* (`CTF:SingleChannelConversion`)

The colour texture function characteristic of this class is:

$$\mathbf{f} = F\left(s, x_1, x_2\right); \tag{9}$$

where:

$$s = S\left(c_1, c_2, c_3\right) \tag{10}$$

being S a scalar-valued function.

The above equations put in evidence that the methods of this class evaluate the contribution of the spectral coordinates by means of an intermediate function S. The most common implementations of S are grayscale conversion and colour quantization (colour indexing). We refer to these approaches as `CTF:Grayscale` and `CTF:ColourIndexing` respectively. For related formulas and algorithms the interested reader is referred to references [23, 84]. It is worth mentioning that, in the case of colour indexing, for the method to be applied in classification, it is mandatory that the same color be labeled the same way in different images.

3. *Purely spectral* (`CTF:PurelySpectral`)

In this class the colour texture function has the following form:

$$\mathbf{f} = F\left(c_1, c_2, c_3\right) \tag{11}$$

In practice, with respect to (6), we have dropped the dependence on the spatial dimensions x_1 and x_2. Therefore the methods of this class rely on the colour content of the image only, irrespective of the relative spatial distribution. As a consequence they are invariant to any spatial redistribution of the image. In most cases the function F is related to the statistical distribution of the spectral content. If we marginalize the function in (11) we obtain two sub-classes, depending on the number of coordinates that we retain: `CTF:Marginalization2D`, and `CTF:Marginalization1D`.

Marginalization / 2D (`CTF:Marginalization2D`) This group inherits from `CTF:PurelySpectral` with the additional constraint that only two spectral coordinates are considered. Therefore the colour texture function can be expressed as follows:

$$\mathbf{f} = F\left(c_h, c_k\right) \tag{12}$$

for $h, k \in \{1, \cdots, 3\}, h \neq k$

Marginalization / 1D (`CTF:Marginalization1D`) This group inherits from `CTF:Marginalization2D` with the further constraint that only one spectral coordinate is considered. The colour texture function takes the following form:

$$\mathbf{f} = F\left(c_h\right) \tag{13}$$

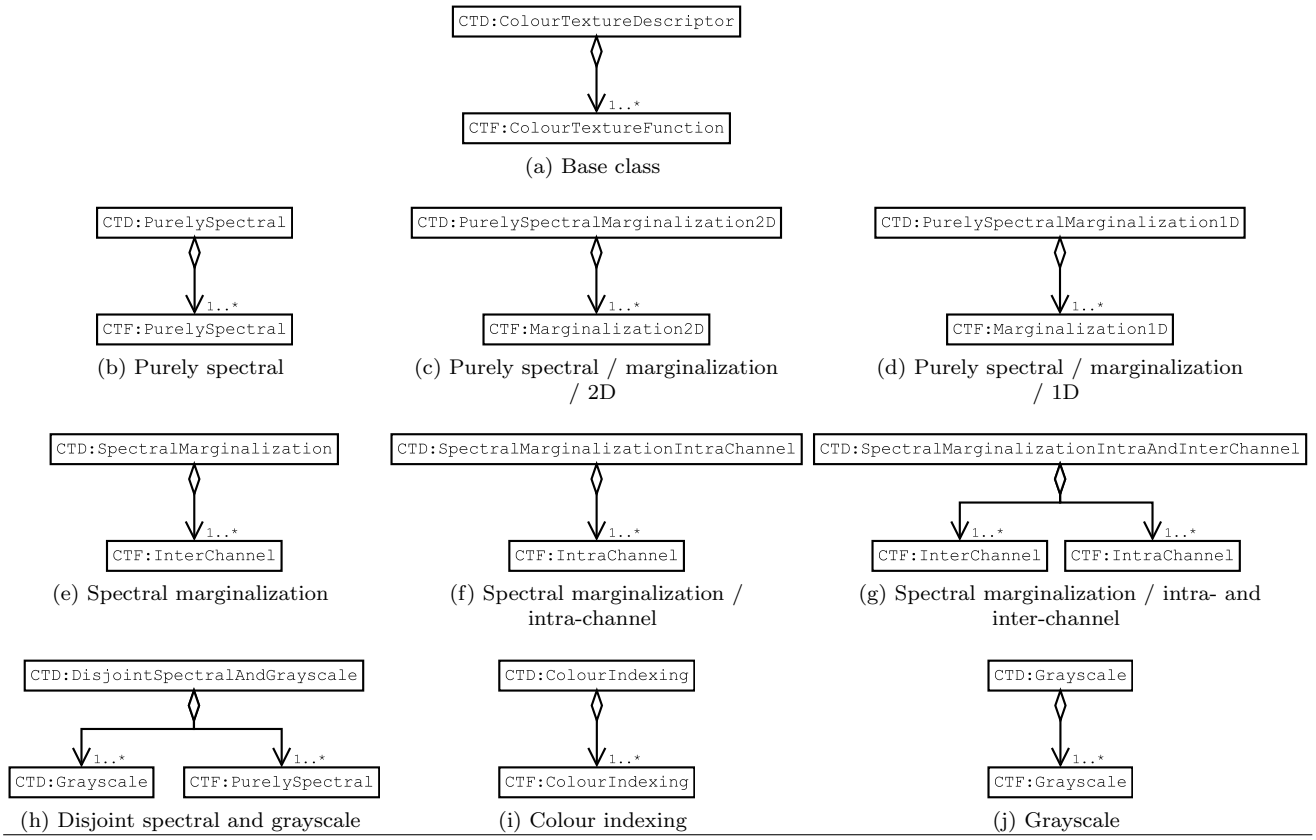
for $h \in \{1, \cdots, 3\}$.

B. Taxonomy of colour texture descriptors

According to Definition 2, colour texture descriptors are obtained through composition of colour texture functions. The base class `CTD:ColourTextureDescriptor` is a composition of instances of `CTF:ColourTextureFunction`, and therefore represents a generic approach to colour texture description (Figure 2a). The other classes of this taxonomy, that we detail in the remainder of this section, include most of the methods described in literature.

1. *Purely spectral* (`CTD:PurelySpectral`)

Purely spectral colour texture descriptors are obtained through composition of one or more purely spectral colour texture functions (Figure 2b). From this class it is convenient to derive two subclasses: `CTD:PurelySpectralMarginalization2D` and `CTD:PurelySpectralMarginalization1D`. The corresponding diagrams are depicted in Figures 2c and 2d.



(a) Base class (b) Purely spectral (c) Purely spectral / marginalization / 2D (d) Purely spectral / marginalization / 1D (e) Spectral marginalization (f) Spectral marginalization / intra-channel (g) Spectral marginalization / intra- and inter-channel (h) Disjoint spectral and grayscale (i) Colour indexing (j) Grayscale

FIG. 2: Taxonomy of colour texture descriptors

2. Spectral marginalization (`CTD:SpectralMarginalization`)

The methods of this class are obtained through composition of one or more instances of `CTF:InterChannel`. The general diagram is shown in Figure 2e. Due to the large number of existing methods, it is convenient to further differentiate between intra-channel and inter- and intra-channel methods, which correspond to the two classes described here below.

3. Spectral marginalization / intra-channel (`CTD:SpectralMarginalizationIntraChannel`)

The methods of this class are obtained through composition of one or more instances of `CTF:IntraChannel`, as shown in Figure 2f.

4. Spectral marginalization / intra and inter-channel (`CTD:SpectralMarginalizationIntraAndInterChannel`)

The methods of this class are obtained through composition of one or more instances of `CTF:IntraChannel` and one or more instances of more instances of `CTF:InterChannel` (Figure 2g).

5. Disjoint spectral and grayscale (`CTD:DisjointSpectralAndGrayscale`)

The instances of this class are usually identified with the expressions "texture and colour features separately" [59] or "parallel" methods, in Palm's classification scheme [69]. They are the result of the composition of instances of `CTF:PurelySpectral` and `CTF:Grayscale`, as shown in Figure 2h.

6. Colour indexing (`CTD:ColourIndexing`)

Colour indexing texture descriptors are obtained through composition of one or more instances of `CTF:ColourIndexing` (Figure 2i). In Palm's classification scheme [69] these methods are referred to as "sequential".

7. *Grayscale* (CTD:Grayscale)

Grayscale texture descriptors are obtained through composition of one or more instances of CTF:Grayscale (Figure 2j). They represent the classic grayscale methods for texture analysis.

III. METHODS FOR COLOUR TEXTURE CLASSIFICATION

In this section we recall the most prominent approaches to colour texture classification and assign each of them to the corresponding places within the taxonomy introduced in the previous section. As a preliminary step we wish to discuss the inclusion/exclusion criteria that we used to select the methods considered in this paper.

- First criterion: *colour*. We include only methods that consider colour in a 'strict' sense, therefore we discard classic grayscale approaches (class CTD:Grayscale). The interested reader is referred to references [5, 13, 49, 76, 79, 82, 89] for comparisons and reviews on these methods. Our investigation is also limited to three-channel colour images, consequently multispectral approaches are not considered here.
- Second criterion: *application in classification*. We consider only methods that can be applied to image classification. This means that features extracted from different images need to be comparable with each other. Indeed many methods exist for colour texture segmentation; in most cases, however, the domain of the colour texture features is limited to the segmented image. As a consequence such methods are not applicable to classification, and therefore they are not considered herein. Up-to-date reviews and comparisons on these methods can be found in references [16, 31, 63, 67, 93, 95].
- Third criterion: *rotation invariance*. Robustness against rotation plays an important role in many applications, since real-world textures can occur, in principle, at any orientation. Based on this consideration we consider only methods that are either intrinsically rotationally-invariant or can easily be converted into rotationally-invariant forms. There are, actually, some notable methods (e.g: multivariate image analysis [50], image tower approach [62]) that we did not include in the comparison since the extension of them to rotationally-invariant versions, assuming that it is possible, is not straightforward.

The following sections are organized in compliance with the taxonomy given in Section II B. Each section corresponds to a class, and for each class we describe, in separate paragraphs, one or more methods that represent instances of that class.

A. *CTD:ColourTextureDescriptor*

This represents the base class. We include here all the methods that do not belong to any of the child classes. This is a common trait of the taxonomy proposed in this paper: a method that does not meet the restrictions imposed by the child classes is classified as belonging to the base class.

Normalized colour space representation This method, described in [90], is based on preliminary dimensionality reduction (from three to two) of the colour space. This makes it possible to represent the resulting pseudo-colours as complex numbers of the form $(P_1 + jP_2)$. The approach to dimensionality reduction leads to three possible and equivalent colour pairs. For each image the choice of the appropriate pair is based on the identification of the least significant colour component of the original image. As suggested by the authors we used the *range/avg* parameter to identify the channel to be discarded. The original image is then converted into a matrix of complex numbers, to which classic filtering methods can be applied. It is worth noticing that in this case filtering can be regarded as a chromato-spatial operation, therefore significantly different from classic grayscale filtering. In the original approach textural features are extracted by computing the normalized energy distribution over eight concentric, diadically-spaced, disks, and 32 uniformly-spaced circular sectors (each one spanning $2\pi/32$ radians). In our experiments, in order to make the method comparable with the other filter-based approaches, we use a Gabor filter bank with the same settings detailed in Section III G, which gives 32 rotationally-invariant features.

Average colour differences The average colour differences [24, 44] are an extension of the semi-variogram [91] to colour images. The method estimates the colour difference as a function of the distance between pixels. Given two generic pixels $\mathbf{v}_i = (c_{1i}, c_{2i}, c_{3i}, x_{1i}, x_{2i})$ and $\mathbf{v}_j = (c_{1j}, c_{2j}, c_{3j}, x_{1i}+nd_1, x_{2i}+nd_2)$, the average colour difference between $\mathbf{v}_i$ and $\mathbf{v}_j$ is defined as follows:

$$\mathrm{AVD}(n, d_1, d_2) = \frac{1}{2} E \left\{ \left[\sum_{h=1}^{3} (c_{hi} - c_{hj})^2 \right]^{\frac{1}{2}} \right\} \qquad (14)$$

where E indicates the expected value. Following the implementation proposed in [24] we considered, in the experiments, four variograms corresponding to the displacements $(d_1, d_2) = \{(1,0), (1,1), (0,1), (-1,1)\}$, for $n = 1, \cdots, 50$. The variograms corresponding diagonal directions $(1,1)$ and $(-1,1)$ have been rescaled by $1/\sqrt{2}$ to ensure that all the variograms span the same length. For rotation invariance we computed the average variogram over the four displacements, which results in a feature vector of dimension 50. As suggested in [24] the average colour difference is computed on the L*a*b* colour space.

Tex-Mex colour features Tex-Mex colour features are based on a morphological operator called *sieve*. The grayscale sieve, originally developed by Bangham *et al.* [3], operates in two steps: 1) identification of the extremal connected sets (regions brighter or darker than their neighbours); 2) union of the $S_{\min}$-scale regions with their most extreme neighbouring region. The process is iterated through a set of scales $\{S_{\min}, \cdots, S_{\max}\}$ and gives, as a result, a set of "sieved" images. Since none of the two steps described above is affected by a rotation of the input image, the method is rotationally-invariant. The sieve has subsequently been extended to the colour domain through the *convex colour sieve* [22]. This is based on convex hulls of the spectral coordinates to identify colour extrema, and on subsequent merging of them to their nearest neighbour found through Euclidean distance measure. Application of the colour sieve to texture classification has been described in [82]. The texture features are the mean and standard deviation of the differences (*granules*) between two successive sieved images. In the experiments we used the same scale sequence proposed in [82]: $\{0, 1, 2, 3, 5, 9, 16, 28, 48, 84, 145\}$, where the values represent areas (in pixels). The mean and standard deviation are computed from each channel of each granule, resulting in a feature vector of $2 \times 3 \times 10 = 60$ features.

B. *CTD:PurelySpectral*

Colour histogram The colour histogram, originally proposed by Swain and Ballard [84], is the estimated probability density function of an image in the colour space. In practice the method consists in dividing the colour space in parts of equal volume and counting how many times each part is represented in the input image. In the implementation described in [84], which is the one used in our experiments, the images are first converted to the *rg-by-wb* colour space. The axes of this colour space are then divided into 16, 16, and 8 sections respectively, giving a feature vector of 2048 elements.

Fuzzy colour histogram Konstantinidis proposed a fuzzy partition of the colour space based on perceptual similarity [37]. The idea is that each colour triple (c_1, c_2, c_3) can be assigned a membership value from 0 to 1, which refers to 10 classes related to the commonsense idea of colour: *black*, *darkgrey*, *red*, *brown*, *yellow*, *green*, *blue*, *cyan*, *magenta*, and *yellow*. The membership value is computed through a set of triangular membership functions and 27 fuzzy rules. The method employs previous conversion to the L*a*b* space.

C. *CTD:PurelySpectralMarginalization2D*

Chromaticity moments The chromaticity moments [70] are statistics extracted from a bidimensional colour distribution. They are defined as follows:

$$M_{D_{hk}}(m,l) = \sum_{h=h_{\min}}^{h_{\max}} \sum_{k=k_{\min}}^{k_{\max}} h^m k^l D(c_h, c_k) \tag{15}$$

$$M_{T_{hk}}(m,l) = \sum_{h=h_{\min}}^{h_{\max}} \sum_{k=k_{\min}}^{k_{\max}} h^m k^l T(c_h, c_k) \tag{16}$$

where D is the estimated bidimensional probability density function in the c_1c_2 plane of an image, and T its trace. The latter is the binary version of the former, meaning that the value of an entry is 1 if the corresponding probability density is positive, 0 otherwise. In the implementation proposed in [70] the images are first converted into the xyY space, then two sets of *D-type* and *T-type* moments are computed from a rescaled and discretized version of the xy plane (in the equations h and k denote the discretized variables). The values of $h_{\min}, k_{\min}$ and $h_{\max}, k_{\max}$ are set to 0 and 99, respectively. In the experiments we consider five D-type moments and five T-type moments (= 10 features) corresponding to the following values of the exponents: $(l, m) \in \{(0,1), (1,0), (1,1), (1,2), (2,1)\}$. It is important to point out that the magnitudes of the moments as they are defined in (15) and (16) can vary significantly as the exponents m and l change. This represents a potential drawback in classifcation. To avoid this problem we normalize the moments value by dividing it for the corresponding maximum attainable value:

$$\max\left[M_{D_{hk}}(m,l)\right] = h_{\max}^m k_{\max}^l \tag{17}$$

$$\max\left[M_{T_{hk}}(m,l)\right] = \sum_{h=h_{\min}}^{h_{\max}} \sum_{k=k_{\min}}^{k_{\max}} h^m k^l \tag{18}$$

D. *CTD:PurelySpectralMarginalization1D*

One-dimensional marginalization, proposed by various authors, is based on the assumption that the colour coordinates are uncorrelated. Theoretically such assumption is not entirely correct for the RGB space. It is more valid for other colour spaces such as Ohta's and HSV. Possible implementations of one-dimensional marginalization are colour histograms or even simpler colour statistics computed on each colour channel separately.

Marginal histograms Pietikainen *et al.* [74] proposed the use of concatenated 1D histograms in the RGB, rgb and Ohta's colour spaces. In each case the one-dimensional histograms are quantized in 256 bins, which results in $256 \times 3 = 768$ features.

Lepistö et al. [46] employed individual H and V histograms in the HSV colour space and a concatenation of

both. In the experiments we maintain 256 bins, giving feature vectors of 256, 256 and 512 components respectively.

Colour statistics: mean Colour statistics, sometimes referred to as *soft colour descriptors* aim at approximating marginal histograms through a reduced set of parameters. These can be as simple as the average value of each channel in the RGB and rgb spaces, as proposed in [38] for surface grading.

Colour statistics: mean + std + moments In a study conducted by López et al. [51] the authors compared various combinations of colour statistics and obtained the best results combining mean, standard deviation and moments from degree two to five for each channel in the RGB and L*a*b* spaces. In their approach the moment of degree n for a spectral dimension h is defined as follows [51]:

$$M_h(n) = \sum_{j=1}^{N} \left(c_{hj} - \mu_{c_h}\right)^n p(c_h) \tag{19}$$

where μ_{c_h} is the mean value of the spectral dimension h, and $p(c_h)$ the discrete probability distribution of c_h, which hs been estimated through a 256-bin histogram. The moments as defined in (19), however, have the disadvantage of being non-dimensionless and therefore inhomogeneous, with potential drawbacks in the classification stage. To avoid this problem we used, in the experiments, the *standardized* moments, which are the moments as defined above divided by $\sigma_{c_h}^n$, where σ_{c_h} is the standard deviation of c_h. As a consequence we take into account degrees from three to five, since the standardized moment of order two is one by definition (and therefore useless), whereas the standardized moments of degree three and four correspond to *skewness* and *kurtosis*, respectively. Consequently the number of features is $5 \times 3 = 15$.

Colour statistics: percentiles The use of percentiles to model the colour distribution is described in [64]. In our experiments we considered the *first*, *second* and *third quartile* of each channel in the RGB space, which result in a feature vector of nine components.

E. *CTD:ColourIndexing*

The methods of this group are based on preliminary conversion of the three spectral coordinates into a single scalar value. In the classification scheme proposed by Palm [69] these methods are referred to as *sequential approaches*. Meaningful implementations of the function S –Eq. (9)– are grayscale conversion, which –as mentioned at the beginning of this section– is not considered here, and colour indexing. For all the methods presented in this section, colour indexing is obtained through uniform partition of the RGB colour space with a number of colours $N_c = \{8, 27, 64\}$.

Co-occurrence of colour indices Co-occurrence of colour indices [1, 26] is a co-occurrence matrix applied to the colour labels of an indexed image. For the method to be applied in classification, colour indexing needs to be image-independent, in order to assure that the same color is labeled the same way in different images. To this end, as suggested in [1], we used a uniform partition of the RGB space. In the experiments we compute eight co-occurrence matrices corresponding to the following displacement vectors: $\{(1,0),(1,1),(0,1),(-1,1),(-1,0),(-1,-1),(0,-1),(1,-1)\}$. The matrices are averaged for rotation invariance and the following five features are computed:
1) *contrast*:

$$CN = \frac{1}{(G-1)^2} \sum_{u=0}^{G-1} \sum_{v=0}^{G-1} |u-v|^2 p(u,v) \tag{20}$$

2) *correlation*:

$$CR = \frac{1}{2} \sum_{u=0}^{G-1} \sum_{v=0}^{G-1} \frac{(u-\mu_u)(v-\mu_v)}{\sigma_u^2 \sigma_v^2} p(u,v) + 1 \tag{21}$$

3) *energy*:

$$EN = \sum_{u=0}^{G-1} \sum_{v=0}^{G-1} p(u,v)^2 \tag{22}$$

4) *entropy*

$$ET = -\frac{1}{2\log_2(G)} \sum_{u=0}^{G-1} \sum_{v=0}^{G-1} p(u,v) \log_2 \left[p(u,v)\right] \tag{23}$$

5) *homogeneity*

$$HM = \sum_{u=0}^{G-1} \sum_{v=0}^{G-1} \frac{p(u,v)}{1+|u-v|} \tag{24}$$

where u,v are the coordinates of the co-occurrence matrix, G is the number of gray levels, μ_u, μ_v, σ_u and σ_v are the mean values and the standard deviations of the u-th row and of the v-th column of the co-occurrence matrix respectively. The above definitions ensure that all the features have range $[0, 1]$. Throughout this paper we maintained this settings (displacements and features) wherever co-occurrence matrices appear.

Multilayer CCR The multilayer Coordinated Clusters Representation (multilayer CCR) [8] is an extension of the Coordinated Cluster Representation (CCR) [41, 81] to colour images. It is based on preliminary colour indexing and subsequent subdivision of the image plane into a stack of binary layers, each one corresponding to a colour class. In each layer the value of a pixel is one if its colour label matches that of the layer, zero oth-

erwise. The method consist of computing the histograms of occurrence of the 512 binary patterns that can be defined in a 3×3 square window. This process is repeated for each layer, and the resulting histograms are concatenated. Since colour labels are mutually exclusive, one layer is redundant and can be discarded. In the experiments we employ uniform partition of the RGB space and applied the $CCR^{ri}_{8,1}$ rotationally-invariant operator [19] to each layer. This gives $72 \times (N_c - 1)$ features.

F. *CTD:SpectralMarginalizationInter-Channel*

Inter- and intra-channel descriptors, of which we discuss in this section and in the following two sections, are related to two different theories of colour vision, namely the *opponent-colours theory* and the *trichromatic theory*, respectively. The opponent-colours theory, originally proposed by the German physiologist K.E. Hering around 1872-1874, states that colours are processed in three pairs: red-green, yellow-blue and white-black. The trichromatic theory, of which T. Young and H. von Helmholtz are credited to be the main contributors, affirms that the signals (approximatively blue, red and green) coming from the three cone types of the retina are transmitted to the brain where they are processed directly. Even if for many years the opponent-colours theory competed with the thricromatic theory of colour, subsequently it became accepted that either are adequate to model different processes that in fact occur in two separate but sequential zones of the brain [94].

In computer vision intra-channel features can be either considered alone (Section III G) or together with inter-channel features (Section III H). Inter-channel features are seldom used alone. An exception is represented by the approach of Gevers and Smeulders [21], which is summarized below.

Histograms of colour ratios Colour ratios, described in [21], are inter-channel features computed between neighbouring pixels. Given two neigbouring pixels $\mathbf{v}_i = (c_{1i}, c_{2i}, c_{3i}, x_{1i}, x_{2i})$ and $\mathbf{v}_j = (c_{1j}, c_{2j}, c_{3j}, x_{1j}, x_{2j})$, colour ratios are:

$$m_1 = \frac{c_{1i}c_{2j}}{c_{1j}c_{2i}}, \quad m_2 = \frac{c_{1i}c_{3j}}{c_{1j}c_{3i}}, \quad m_3 = \frac{c_{2i}c_{3j}}{c_{2j}c_{3i}} \tag{25}$$

Therefore, for a given displacement we can compute three colour ratios. The image features are the histograms of the colour ratios. Particular attention must be paid to the histogram partition, since the ratios defined in (25) cannot be considered uniformly distributed, as correctly stated by Funt and Finlayson [20]. Therefore we used a quantization scheme in which each bin divides the ratios' domain into equally probable parts under the assumption of uniform distribution of c_1, c_2 and c_3. In our experiments we considered eight unit displacements, and for each displacement we computed the corresponding histograms of m_1, m_2 and m_3. To obtain rotationally-invariant features the histograms are averaged over the displacements; this results in three histograms, one for each colour ratio. As recommended in [21], the number of bins is set to 32, giving a vector of $32 \times 3 = 96$ features. To avoid division by zero in (25), the colour ratios are computed only when the denominator is not zero.

G. *CTD:SpectralMarginalizationIntra-Channel*

Inter-channel features are computed by applying classic grayscale methods to each colour channel separately.

Gabor features The use of Gabor features to extract orientation and scale information from different bands is described in [71, 73]. In our experiments we use a Gabor filter bank with the following parameters: number of frequencies = four, number of orientations = six, maximum frequency = 0.327, frequency ratio = half-octave and smoothing parameters $\eta, \gamma = 0.5$. The filter bank design is based on the considerations reported in [6]. For each colour plane we extract the mean and standard deviation of the absolute value of the transformed images as texture features. This gives $4 \times 6 \times 3 \times 2 = 144$ monochrome features, where the first three factors represent the number of frequencies, number of orientations, and number of channels respectively. Rotation-invariant features are obtained through Discrete Fourier Transform (DFT) normalization [9, 45], which results in $4 \times 4 \times 3 \times 2 = 96$ rotationally-invariant features. Throughout the paper we maintain the same configuration for Gabor filtering.

Gabor features on Gaussian colour model Hoang *et al.* [29] proposed an improvement of the approach described in the preceding paragraph, based on previous conversion of the RGB data into the Gaussian colour model. The rationale behind the method is that a colour image can be regarded as an energy density function $E(x, y, \lambda)$ in the spectral-spatial domain, and that such function can be *observed* through a suitable probe function $p(x, y, \lambda)$. In the implementation proposed in [29] the probe is decomposable, and it is actually decomposed into a spectral and a spatial part. Texture measurement is carried out in two sequential steps: 1) colour measurement with the Gaussian colour models, and 2) spatial measurement through Gabor filtering. The first step has been implemented using the same linear transform reported in [29]; the second through the same Gabor filter bank described in the previous paragraph.

Granulometry Hanbury *et al.* [24] suggested applying morphological operators (granulometries) to each colour channel separately. Thus they compute the normalized sum of the pixel values of an image when transformed with a family of openings and closings, as a function of the size of the structuring elements. In formulas, for the k-th colour channel we have:

$$G_k(l) = \frac{Vol_k\left[\phi_l\left(\mathbf{I}\right)\right]}{Vol_k\left(\mathbf{I}\right)} \tag{26}$$

where l is the dimension of the structuring element, ϕ the morphological operator and, using the same notation introduced in (5), Vol_k can be expressed as follows:

$$Vol_k\left(\mathbf{I}\right) = \sum_{j=1}^{N}(c_{kj}) \tag{27}$$

In the experiments we use the same settings described in [24], namely four linear structuring elements with orientations $\{0°, 45°, 90°, 135°\}$ for both opening and closing. The dimension of the elements ranges from 2 to 50 pixels by step 4. To obtain rotationally-invariant features the four granulometry vectors corresponding to each direction are averaged. The total number of features is $13 \times 2 \times 3 = 78$.

Wavelets and co-occurrence histograms Hiremath *et al.* [28] described a method for colour texture characterization based on wavelets and co-occurrence histograms applied to each colour channel separately. The first step consists in computing the 1-level wavelet decomposition of each colour channel and its complement (i.e.: each pixel value is replaced by the complement to the maximum intensity level). Let's denote the approximation, vertical, horizontal, and diagonal wavelet coefficients with A, V, H, and D respectively. The second step consists in computing joint co-occurrence histograms between the following couples of wavelet coefficients: (A_1, H_1), (A_1, V_1), (A_1, D_1), $(A_1, |D_1 - H_1 - V_1|)$, $(\bar{A}_1, \bar{H}_1), \cdots, (\bar{A}_1, |\bar{D}_1 - \bar{H}_1 - \bar{V}_1|)$, where $\bar{}$ represents the complement operator. For each couple of wavelet coefficients and a given displacement, two histograms of co-occurrence probability are constructed based on a max-min composition rule [28]. Afterward the co-occurrence histograms are converted into cumulative probability histograms and normalized as detailed in [39]. Three features are then extracted from each normalized cumulative histograms: slope of the regression line, mean and mean deviation. In the experiments we considered the same displacement described in Section III E. To obtain rotationally-invariant features, the histograms are averaged over the eight displacements, giving $8 \times 2 \times 3 \times 3 = 144$ features, where the factors indicate the number of wavelet couples, the number of histograms for each couple, the number of features for each histogram and the number of channels.

Complex Wavelet Features Barilla and Spann [4] applied the Dual Tree Complex Wavelet Transform (DT-CWT) to each colour channel. If compared to the Discrete Wavelet Transform (DWT), the DT-CWT has potential advantages, such as moderate redundancy and directional selectivity [36]. Whereas the DWT can only separate sub-bands into vertical, horizontal, and diagonal, the DT-CWT can provide a reasonable amount of directions, namely ±15°, ±45°, and ±75. To maintain the same number of scales and orientations used with Gabor filtering, we consider four scales and the six orientations cited above. For each sub-band the mean and the standard deviation of the absolute value of the CWT coefficients are used as texture features. For each scale the features are averaged over the six orientations to give rotationally-invariant features. We believe that DFT normalization, though adopted by some authors, is not recommendable here due to the different sensitivity of the complex wavelets (the ±15° and ±75° components are more sensitive than the ±45° [2]). This setting results in $4 \times 2 \times 3 = 18$ features, which are the number of scales, the number of features for each scale (averaged over the orientations), and the number of channels.

H. *CTD:SpectralMarginalizationIntraAnd-InterChannel*

Opponent Gabor features Inter- and intra-channel Gabor features have been described by Jain and Healey [32]. In their work they consider monochrome features –extracted from each colour plane separately– and opponent features –extracted from couples of colour planes. The second group can be further subdivided in two subgroups: 1) features computed from the normalized difference of the Gabor transforms at same scale and orientation on couples of different colour channels, and 2) features computed from the normalized difference of the Gabor transforms at different scale and orientation on couples of different colour channels. In [32] a unit difference in scale is considered at most (for instance if the number of scales is three, for each orientation the following opponent features are computed: 1/1, 2/2, 3/3, 1/2, and 2/3, where the fractions indicate the scale ratio). Using the same filter bank detailed in Section III G we get $4 \times 6 \times 3 \times 2 = 144$ monochrome features, $4 \times 6 \times 3 \times 2 = 144$ opponent and homo-scale features and $3 \times 6 \times 3 \times 2 = 108$ opponent and shifted-scale features, all these features being not rotationally-invariant. The factors in the preceding expressions indicate the number of frequencies, orientations, number of channels and features per filter respectively. DFT normalization is used to obtain the following set of rotationally-invariant features: $4 \times 4 \times 3 \times 2 = 96$ monochrome features, $4 \times 4 \times 3 \times 2 = 96$ opponent and homo-scale features and $3 \times 4 \times 3 \times 2 = 72$ opponent and shifted-scale features. This gives a feature vector of which dimension is 264.

Opponent Colour Local Binary Patterns (OCLBP) The Opponent Colour LBP [60] is an extension of the standard Local Binary Patterns (LBP) [65] to colour images. In the OCLBP the LBP operator is applied on each color channel separately and on three couples of colour channels jointly: (c_1, c_2), (c_1, c_3) and (c_2, c_3). Inter-channel (opponent) features are computed by picking the threshold value from the central pixel of the first

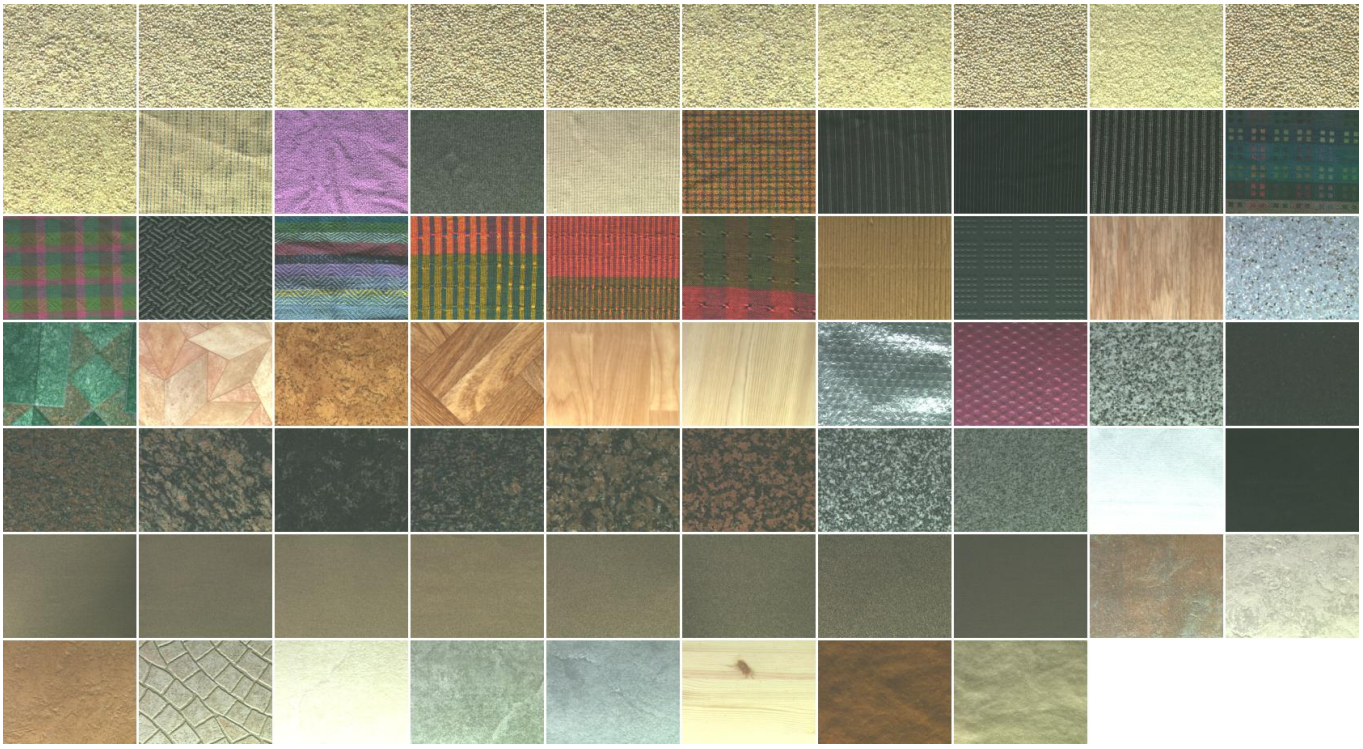


FIG. 3: Dataset: 68 colour textures of the OUTEX_TC_00013 test suite

plane and applying it to the peripheral pixels of the second plane. In the experiments we use the $LBP^{ri}_{8,1}$ rotationally-invariant operator, which provides 36 features. The total number of features of the OCLBP is therefore $36 \times 6 = 216$.

Integrative co-occurrence matrices Integrative co-occurrence matrices have been proposed, almost at the same time, by Palm [69] and by Arvis *et al.* [1]. The method includes both intra- and inter-channel features. The first are computed by applying the grayscale method to each colour channel separately. The second consider the co-occurrence of inter-channel pixel values among couples of colour channels: (c_1, c_2), (c_1, c_3) and (c_2, c_3). In the experiments we use the same settings reported in Section III E, which give, in this case, $5 \times 3 = 15$ monochrome features and $5 \times 3 = 15$ opponent features.

Colour ranklets Colour ranklets [7] extend the ranklet method to the colour domain. Original ranklets [56, 80] are non parametric texture descriptors. They are based on splitting a window in two subsets of predefined geometry and on checking whether there is significant difference between the pixel values of the two subsets. The original version of the method considers grayscale images and vertical, horizontal and diagonal subdivisions only. Colour ranklets extend it in two different directions: orientational selectivity and rotation-invariance on the one hand, and colour on the other hand. Colour is taken into account through intra-channel features (ranklets of each colour channel separately) and inter-channel features computed among couples of colour channels (opponent features). Subdivisions of the window are defined at different scales and orientations. For each subdivision the mean and standard deviation of the statistical test are retained as texture features. In the experiments we adopted the same settings used in [7], namely: three window sizes (4×4, 6×6 and 8×8 pixels), six linear subdivisions ($\theta = \{0°, 30°, 60°, 90°, 120°, 150°\}$) and three diagonal subdivisions ($\theta = \{0°, 30°, 60°\}$). This gives $3 \times (6+3) \times 2 \times 3 = 162$ monochrome features and the same number of opponent features, all these being not rotationally-invariant. After DFT normalization for rotation invariance, we get $3 \times (4+2) \times 2 \times 3 = 108$ monochrome features and the same number of opponent features, for a total of 216 features.

I. *CTD:DisjointSpectralAndGrayscale*

Disjoint spectral and grayscale analysis involves previous conversion into a colour spaces which separates intensity and chrominance (e.g.: HSV, L*a*b*). We used, for the first three methods included in this section, the HSV colour space, which is common to all. In all cases the chromatic features are the mean and standard deviation of the H and S channels. In the last two method presented in this section (LBP + colour percentiles and co-occurrence matrices + colour centiles) the colour features are computed from the RGB space. As a prelim-

inary step both HSV and RGB values are rescaled to interval $[0, 1]$. The feature vector is obtained by concatenating texture and colour features without any further normalization.

Gabor filters and chromatic features Combination of Gabor features from the luminance plane with chromatic features has been studied by Drimbarean and Whelan [17]. In the experiments we maintain the same Gabor settings detailed in Section III G. This gives 32 monochrome rotationally-invariant features + four chromatic features (which are intrinsically rotationally-invariant).

Co-occurrence matrices and chromatic features Combinations of co-occurrence features from the luminance plane and chromatic features have been proposed by Arvis et al. [1]. In the experiments we use the same co-occurrence features detailed in Section III E, which give five monochrome rotationally-invariant features and four chromatic features.

Complex Wavelet features and chromatic features In a similar way Barilla and Span considered the use of DT-CWT features [4] extracted from the luminance plane in combination with chromatic features. With the same settings used in Section III G this gives 16 monochrome rotationally-invariant features and four chromatic features.

Local Binary Patterns and colour centiles Niskanen *et al.* [64] suggested a combination of LBP features from the luminance plane and colour centiles computed on each R, G and B channel for wood inspection. In the experiments we used the $LBP^{ri}_{8,1}$ operator to extract features from the luminance plane and RGB colour centiles the with the same settings described in Section III D. This gives 36 monochrome rotationally-invariant features and nine colour features.

Co-occurrence matrices and colour centiles In [64] the authors consider also a combination of co-occurrence features from the luminance plane and RGB colour centiles. With the same settings used with co-occurrence features and RGB centiles respectively we obtain five monochrome rotationally-invariant features and nine colour features.

IV. EXPERIMENTS

A. Dataset

The dataset is composed of the 68 texture classes (Figure 3) of the OUTEX_TC_00013 test suite [66]. This suite, which provides hardware-rotated texture images acquired under known and controlled illumination conditions, is particularly suitable for comparative purposes, and has been used in a number of related papers [30, 59, 75, 82]. Other databases provide hardware-rotated colour textures too, such as CUReT [14] and the KTH-TIPS group (KTH-TIPS [27] and KTH-TIPS2 [11]). The reasons why we decided to stick to the Outex dataset even in presence of these alternatives are: 1) both CUReT and KTH-TIPS/KTH-TIPS-2 present the inconvenience that the original images contain not only the sample but also some background, making the usable part of the picture rather small; 2) whereas in Outex both the sensitivity of the camera and the spectrum of the illuminant are available as .txt files, enabling the colour calibration procedure described below, none of the alternative datasets come with such data.

As suggested in the Outex website[98], the images, the original size of which is 746×538, have been cropped by centering the sampling grid so that equally many pixels are left on each side. Each trimmed image has then been subdivided into 20 non-overlapping sub-images, with resolution 128×128, giving a database of 1360 texture samples for each rotation angle.

B. Colour spaces

As a preliminary step we converted the images from the device-dependent RGB space to the device-independent sRGB space. A detailed description of the colour calibration procedure is available in [8]. This decision is motivated by the following considerations: 1) with colour calibration we aim at eliminating (or, at least, reducing) any bias related to the imaging system; 2) some of the methods included in the comparison require previous conversion to device-independent spaces (e.g.: xyY, L*a*b*) which is straightforward when the data are already available in another device-independent space; 3) it is fair play to compare all the methods on device-independent colour spaces. When other device-dependent colour spaces are used (e.g.: rgb, rg-by-wb, HSV, I1I2I3), their values are obtained from the sRGB values through the standard conversion formulas RGB/rgb, RGB/HSV, etc.

After this pre-processing operation, the images can be further converted into other colour spaces (see Table I), depending on the implementation proposed by the authors. As a general strategy we decided to maintain the same colour spaces used in the cited references. This approach is motivated by the following reasons. Firstly, there are some methods that only make sense within certain colour spaces (e.g: intra- and, even more, inter-channel methods make little sense in spaces that are not RGB-like). When considering intra-channel features, for instance, it isn't particularly sound to compute them on channels that are not intensity-based. Likewise, when it comes to inter-channel features, it is sensible to compute ratios - or differences - of homogeneous quantities (i.e.: intensity in different channels, R/G, R/B, etc), whereas it sounds quite odd to do the same with, for instance, hue and intensity or saturation and intensity. In summary we believe that the choice of a particular colour space is inherent to the method, and cannot be separated from it. Secondly, it is important to recall that the most significant difference among colour spaces is that they can be either device-dependent or device-independent. This difference is essential, since device-dependent spaces are

TABLE I: Experimental results. Along with the name and references of each method, every single row reports: colour space (*Col.*), computational time (*Time*) over a three-level qualitative scale: low (L), medium (M) and high (H), dimension of the feature space (*Dim.*), mean (*Avg*) and standard deviation (*Std*) of the accuracy over the nine rotation angles, and accuracy for each rotation angle.

Method (instance)	Ref.	Col.	Time	Dim.	**Avg**	*Std*	Rotation angle								
							0°	5°	10°	15°	30°	45°	60°	75°	90°
`CTD:ColourTextureDescriptor`															
Normalized color space repres.	[90]	P1P2	M	32	**61,4**	*6,7*	70,6	69,1	66,6	64,2	62,2	56,8	57,5	52,6	53,0
Average colour differences	[24, 44]	L*a*b	M	50	**45,3**	*6,9*	56,2	47,0	41,7	40,4	42,3	54,6	37,8	38,2	49,8
Tex-Mex colour features	[82]	sRGB	M	60	**68,0**	*1,2*	69,0	68,5	69,2	67,8	69,2	68,0	66,5	65,6	68,1
`CTD:PurelySpectral`															
Colour histogram	[84]	rg-by-wb	M	2048	**84,0**	*1,9*	85,7	85,2	85,0	84,8	85,4	84,7	82,8	81,6	80,4
Fuzzy colour histogram,	[37]	L*a*b	M	10	**57,6**	*2,8*	59,9	61,4	59,0	59,6	58,9	56,4	55,4	54,9	52,8
`CTD:PurelySpectralMarginalization2D`															
Chromaticity moments (CM55)	[70]	xyY	M	10	**64,6**	*2,4*	67,7	66,9	65,8	65,3	66,0	63,8	63,4	61,4	60,8
`CTD:PurelySpectralMarginalization1D`															
Three marginal histograms	[74]	sRGB	L	768	**86,6**	*1,3*	88,1	88,1	87,2	86,5	87,6	86,5	85,8	84,9	84,6
Three marginal histograms	[74]	rgb	L	768	**80,3**	*2,5*	82,4	82,2	82,2	81,9	81,6	80,6	78,7	76,9	75,7
Three marginal histograms	[74]	I1I2I3	L	768	**87,8**	*1,5*	89,0	88,8	88,7	88,8	88,7	88,2	87,3	85,8	85,0
One marginal histogram (H)	[46]	HSV	L	256	**75,6**	*3,9*	79,6	79,1	78,5	78,3	76,9	75,1	72,9	70,5	69,1
One marginal histogram (V)	[46]	HSV	L	256	**65,3**	*1,7*	68,5	66,5	64,3	63,4	65,2	65,6	64,6	63,2	66,5
Marginal histogram (H+V)	[46]	HSV	L	512	**85,3**	*2,4*	87,8	87,3	86,9	86,8	86,7	85,3	83,6	82,2	81,4
Colour statistics (mean)	[38]	sRGB	L	3	**66,7**	*3,4*	71,7	70,1	68,1	68,0	67,4	66,7	63,5	61,2	63,3
Colour statistics (mean)	[38]	rgb	L	3	**52,9**	*1,5*	54,0	53,8	53,3	54,0	53,7	54,5	51,4	50,7	50,7
Col. stat. (mean + std + 3-5th)	[51]	sRGB	L	15	**84,2**	*1,4*	85,9	86,1	84,6	84,2	85,3	84,1	83,5	82,2	82,2
Col. stat. (mean + std + 3-5th)	[51]	L*a*b	L	15	**82,3**	*1,2*	83,3	83,3	82,4	82,6	83,3	82,8	82,0	80,9	79,9
Colour statistics (percentiles)	[64]	sRGB	L	9	**80,8**	*2,4*	83,9	83,4	82,0	81,5	82,0	80,9	78,7	77,5	77,5
`CTD:ColourIndexing`															
Co-occurrence mat., 8 levels	[1]	sRGB	M	5	**39,2**	*1,7*	41,4	40,9	39,3	39,5	39,9	39,9	38,4	36,1	37,1
Co-occurence mat., 27 levels		sRGB	M	5	**52,2**	*2,6*	54,9	53,9	52,3	54,1	54,7	52,6	50,3	48,7	47,8
Co-occurence mat., 64 levels		sRGB	M	5	**48,3**	*2,0*	50,7	50,5	48,4	48,3	49,9	48,8	46,8	45,6	45,5
Multilayer CCR, 8 levels	[8]	sRGB	M	504	**61,9**	*1,7*	64,6	63,3	62,4	61,9	63,5	61,8	60,5	59,3	59,9
Multilayer CCR, 27 levels		sRGB	H	1872	**80,7**	*1,7*	82,4	82,2	81,9	81,9	81,4	80,3	79,9	78,5	77,6
Multilayer CCR, 64 levels		sRGB	H	4536	**82,3**	*1,8*	84,5	84,0	83,4	83,4	83,2	82,3	81,1	79,9	79,4
`CTD:SpectralMarginalizationIntraChannel`															
Gabor features	[53, 71]	sRGB	M	96	**85,9**	*1,3*	86,7	86,5	86,4	85,6	87,9	84,3	86,3	83,8	85,1
Granulometry	[24]	sRGB	M	78	**78,1**	*2,2*	81,1	80,4	77,9	76,9	77,7	80,8	76,2	74,6	77,7
Gabor feat. on Gauss. col. mod.	[29]	sRGB	M	96	**83,9**	*3,5*	87,2	87,2	86,8	86,2	86,3	81,0	82,1	78,2	80,0
Wavelets + Co-occurrence hist.	[28]	sRGB	M	144	**50,1**	*1,1*	52,2	51,2	49,8	49,6	50,8	49,9	49,3	48,3	49,9
DT-CWT (4 scales, 6 ornts)	[4]	sRGB	M	48	**79,9**	*0,9*	80,2	80,5	81,3	80,9	80,2	78,8	79,1	79,5	78,9
`CTD:SpectralMarginalizationInterChannel`															
Histograms of colour ratios	[21]	sRGB	M	96	**42,0**	*3,0*	44,5	44,8	45,1	44,1	43,6	40,9	39,6	37,8	37,9
`CTD:SpectralMarginalizationIntraAndInterChannel`															
Opponent Gabor features	[32]	sRGB	M	264	**86,0**	*2,5*	88,8	88,8	88,2	87,7	85,3	85,0	84,9	83,4	81,6
Opponent colour LBP	[60]	sRGB	M	216	**86,0**	*3,3*	89,7	89,5	89,2	88,7	86,1	84,1	82,4	82,2	82,1
Integrative co-occurrence mat.	[1, 69]	sRGB	M	30	**86,0**	*1,0*	86,8	86,7	86,9	86,4	87,4	85,4	85,1	84,8	84,7
Colour ranklets	[7]	sRGB	H	216	**88,3**	*1,8*	89,5	89,5	89,5	89,7	89,6	88,5	87,4	85,9	84,7
`CTD:DisjointSpectralAndGrayscale`															
Gabor and chromatic features	[17]	sHSV	M	36	**89,3**	*1,2*	90,0	90,1	90,3	89,9	90,8	89,2	88,8	87,8	87,2
DT-CWT + chromatic features	[4]	sHSV	M	20	**74,7**	*1,8*	76,1	76,1	76,2	76,1	75,2	75,6	72,8	72,2	71,8
Co-occurrence mat. + chroma	[1]	sHSV	L	9	**86,2**	*1,1*	87,2	86,9	86,6	86,7	87,1	86,9	85,2	84,8	84,4
Co-occurrence mat. + percentiles	[64]	sRGB	L	14	**85,7**	*1,8*	87,9	87,9	86,4	85,9	86,7	85,6	84,4	83,2	83,3
LBP + percentiles	[64]	sRGB	L	45	**86,1**	*1,6*	88,0	87,8	86,9	86,9	86,9	85,6	84,7	83,6	84,1

affected by a bias, which is related to the acquisition system and is likely to have significant effects on classification. As mentioned above, this source of uncertainty has been previously removed by a calibration procedure. As a results all the spaces considered in the experiments are, in fact, device-independent, which implies fair play in the experiments.

We wish to underline that the experiments have been conducted under steady illumination conditions. Problems related to illumination invariance are not considered in this paper. Readers should be aware that under variable illumination conditions in principle any colour-based method is likely to suffer a significant loss of performance, as remarkably evidenced by Mäenpää and Pietikäinen [59].

C. Classification and estimation of accuracy

To compare the performance of the methods described in the previous sections, we carried out a classification experiment using a nearest-neighbour (1-NN) classifier with L_1 distance. Of the various classification methods that have been proposed in literature, the 1-NN has some desirable features that make it particularly suitable for feature comparison. Among these are: absence of tuning parameters, easiness of implementation and the asymptotical property that the error rate is bounded from above by twice the Bayes error as the number of samples tends to infinity [18].

The procedure for accuracy estimation is based on split-half validation with stratified sampling [83]. This method generates a set of problems (100 in our setting). In each problem the overall texture set is randomly split into two non-overlapping subsets, one for training and the other one for validation. Stratified sampling means that the proportion of samples in the training and validation set is the same for all the classes. This proportion is 0.5 here, meaning that in each problem half of the samples of each class is used for training and the other half for validation. For each problem the estimated accuracy is the percentage of samples of the validation set that has been classified correctly. The overall accuracy is the average of this over the 100 problems. The procedure is described step-by-step here below.

- *Step one*: randomly split the 20 samples of each texture class into two groups of 10 samples each: one for training and the other for validation. Pick the training textures from the 0° group and the test textures from the α-degree group, where $\alpha = \{0^\circ, 5^\circ, 10^\circ, 15^\circ, 30^\circ, 45^\circ, 60^\circ, 75^\circ, 90^\circ\}$.
- *Step two*: classify the samples of the validation set through the nearest neighbour rule with L_1 distance using the training set as labelled examples.
- *Step three*: compute and store the percentage of correct classification.
- *Step four*: repeat steps from *one* to *three* 100 times.

The output of this procedure is an array of 100 values whose average is the estimated accuracy. In order to obtain reproducible results, we computed and stored the 100 random subdivisions (which are available online - see Sec. IV E) at the beginning of the experiment, and used them in all the calculations.

D. Pairwise comparison

We defined the following criterion for pairwise statistical comparison of the methods considered in the experiments: we consider one method significantly better than another if, for each rotation angle, it significantly outperforms the other. To assess whether the performance of two methods is significantly different (at a given rotation angle) we used a t-test for two population means with variances unknown and unequal [34], where mean and variance of the samples are computed as described in the preceeding section. The tests have been conducted at a significance level $\alpha = 0.05$. The results are reported in Table II.

E. Implementation, execution and reproducible research

The methods have been implemented within the MATLAB®package, release 2007a. Gabor features are based on the SIMPLEGABOR©toolbox [88] developed at the Lappeenranta University of Technology, Finland. Dual-Tree Complex Wavelet Features are based on the Matlab implementation of Wavelet Transforms [87] available at the Polytechnic Institute of New York University, USA. The classification experiments were executed using the *High Performance Computing Cluster*[99] available within the University of East Anglia, UK.

For reproducible research purposes we made all the information (code and data) required to replicate the results available online[100].

V. RESULTS AND DISCUSSION

The experimental results are summarized in Table I. Each column of the table reports the following data: name of the method, related references, dimension of the feature space, computational time over a three-level qualitative scale (*low*, *medium* and *high*), mean and standard deviation of the classification accuracy over the nine rotation angles and classification accuracy for each rotation angle.

Table II reports the results of the pairwise comparisons according to the criterion defined in Section IV D. For each entry (i, j) the arrow points in the direction of the method that performs best. The symbol = indicates that there is no significant difference between the two methods. By definition the matrix is antisymmetric.

Based on the results shown in Table I we can establish a ranking by counting the number of "wins", "losses" and "ties" of each method. The ranking is reported in Table III, where we included the methods of which the balance (difference between wins and losses) is non negative.

A potentially useful method for colour texture classification should yield good classification accuracy with relatively few features. Based on this consideration we believe that it is recommendable to compare the methods with respect to both accuracy and dimension of the feature space. To this end we report, in a scatter plot (Figure 4), the number of features of each method listed in the ranking (in logarithmic scale) versus its accuracy. In the graph the two blue dash-dot lines represent the second quartile (median) of the accuracy and of the number

TABLE II: Pairwise comparison. In each entry the direction of the arrow indicates which of the two methods (row or column) performs best. The symbol "=" stands for a tie. The matrix is antisymmetric by definition.

	Normalized color space repres.	Average colour differences	Tex-Mex colour features	Colour histogram	Fuzzy colour histogram	Chromaticity moments (CM55)	Three marginal histograms (sRGB)	Three marginal histograms (rgb)	Three marginal histograms (I1I2I3)	One marginal histogram (H)	One marginal histogram (V)	Two marginal histograms (H+V)	Colour statistics (mean, sRGB)	Colour statistics (mean, rgb)	Colour statistics (mean, std + 3-5th, RGB)	Colour statistics (mean, std + 3-5th, Lab)	Colour statistics (centiles, sRGB)	Co-occurrence mat., 8 levels (sRGB)	Co-occurrence mat., 27 levels (sRGB)	Co-occurrence mat., 64 levels (sRGB)	Multilayer CCR., 8 levels (sRGB)	Multilayer CCR., 27 levels (sRGB)	Multilayer CCR., 64 levels (sRGB)	Gabor features	Granulometry	Gabor feat. On Gauss. Col. mod	Wavelets + Co-occurrence hist.	DT-CWT	Histograms of colour ratios	Opponent Gabor features	Opponent colour LBP	Integrative co-occurrence mat.	Colour ranklets	Gabor + chrom. feat.	DT-CWT + chrom. feat.	Co-occurrence mat. + chrom. feat.	Co-occurrence mat. + percentiles	LBP + percentiles
Normalized color space repres.		←	=	↑	=	=	↑	↑	↑	↑	=	↑	↑	↑	↑	↑	↑	←	←	←	=	↑	↑	↑	↑	↑	←	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
Average colour differences	↑		↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	=	=	=	↑	↑	↑	↑	↑	↑	=	↑	=	↑	↑	↑	↑	↑	↑	↑	↑	↑
Tex-Mex colour features	=	←		↑	←	←	↑	↑	↑	↑	←	↑	=	←	↑	↑	↑	←	←	←	←	↑	↑	↑	↑	↑	←	↑	←	↑	↑	↑	↑	↑	↑	↑	↑	↑
Colour histogram	←	←	←		←	←	↑	←	↑	←	←	↑	←	←	=	=	←	←	←	←	←	←	←	←	=	←	=	←	←	←	=	=	↑	↑	←	↑	↑	↑
Fuzzy colour histogram,	=	←	↑	↑		↑	↑	↑	↑	↑	↑	↑	↑	←	↑	↑	↑	←	←	←	↑	↑	↑	↑	↑	↑	←	↑	←	↑	↑	↑	↑	↑	↑	↑	↑	↑
Chromaticity moments (CM55)	=	←	↑	↑	←		↑	↑	↑	↑	=	↑	=	←	↑	↑	↑	←	←	←	←	↑	↑	↑	↑	↑	←	↑	←	↑	↑	↑	↑	↑	↑	↑	↑	↑
Three marginal histograms (RGB)	←	←	←	←	←	←		←	↑	←	←	=	←	←	←	←	←	←	←	←	←	←	←	←	=	←	←	←	←	←	=	=	=	↑	↑	←	=	←	=
Three marginal histograms (rgb)	←	←	←	↑	←	←	↑		↑	←	←	↑	←	←	↑	↑	=	←	←	←	←	=	↑	↑	=	↑	←	=	←	↑	↑	↑	↑	↑	←	↑	↑	↑
Three marginal histograms (I1I2I3)	←	←	←	←	←	←	←	←		←	←	←	←	←	←	←	←	←	←	←	←	←	←	←	=	←	←	←	←	←	=	=	←	=	↑	←	←	←	←
One marginal histogram (H)	←	←	←	↑	←	←	↑	↑	↑		←	↑	←	←	↑	↑	↑	←	←	←	←	↑	↑	↑	=	↑	←	↑	←	↑	↑	↑	↑	↑	=	↑	↑	↑
One marginal histogram (V)	=	←	↑	↑	←	=	↑	↑	↑	↑		↑	=	←	↑	↑	↑	←	←	←	←	↑	↑	↑	↑	↑	←	↑	←	↑	↑	↑	↑	↑	↑	↑	↑	↑
Marginal histogram (H+V)	←	←	←	←	←	←	=	←	↑	←	←		←	←	=	=	←	←	←	←	←	←	←	←	=	←	←	←	←	←	=	=	=	↑	↑	←	=	=	↑
Colour statistics (mean, RGB)	←	←	=	↑	←	=	↑	↑	↑	↑	=	↑		←	↑	↑	↑	←	←	←	←	↑	↑	↑	↑	↑	↑	↑	←	↑	←	↑	↑	↑	↑	↑	↑	↑
Colour statistics (mean, rgb)	↑	=	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	←	=	←	↑	↑	↑	↑	↑	↑	↑	↑	←	↑	↑	↑	↑	↑	↑	↑	↑	↑
Col. stat. (mean + std + 3-5th, RGB)	←	←	←	=	←	←	↑	←	↑	←	←	=	←	←		=	←	←	←	←	←	←	←	←	↑	←	=	←	←	←	=	=	↑	↑	↑	↑	↑	↑
Col. stat. (mean + std + 3-5th, Lab)	←	←	←	=	←	←	↑	←	↑	←	←	=	←	←	=		←	←	←	←	←	←	←	←	↑	←	=	←	←	←	=	=	↑	↑	↑	←	↑	↑	↑
Colour statistics (percentiles, RGB)	←	←	←	↑	←	←	↑	=	↑	←	←	↑	←	←	↑	↑		←	←	←	←	=	↑	↑	=	↑	←	=	←	↑	↑	↑	↑	↑	←	↑	↑	↑
Co-occurrence mat., 8 levels	↑	=	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
Co-occurence mat., 27 levels	↑	=	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	=	↑	↑	↑	←		←	↑	↑	↑	↑	↑	↑	=	↑	←	↑	↑	↑	↑	↑	↑	↑	↑	↑
Co-occurence mat., 64 levels	↑	=	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	←	↑		↑	↑	↑	↑	↑	↑	↑	↑	←	↑	↑	↑	↑	↑	↑	↑	↑	↑
Multilayer CCR, 8 levels	=	←	↑	↑	←	↑	↑	↑	↑	↑	↑	↑	↑	←	↑	↑	↑	←	←	←		↑	↑	↑	↑	↑	←	↑	←	↑	↑	↑	↑	↑	↑	↑	↑	↑
Multilayer CCR, 27 levels	←	←	←	↑	←	←	↑	=	↑	←	←	↑	←	←	↑	↑	=	←	←	←	←		↑	↑	=	=	←	=	←	↑	↑	↑	↑	↑	←	↑	↑	↑
Multilayer CCR, 64 levesl	←	←	←	↑	←	←	↑	←	↑	←	←	↑	←	←	↑	↑	←	←	←	←	←	←		↑	←	=	←	←	←	↑	↑	↑	↑	↑	←	↑	↑	↑
Gabor features	←	←	←	=	←	←	=	←	=	←	←	=	←	←	←	←	←	←	←	←	←	←	←	←		←	=	←	←	←	=	=	=	=	↑	←	=	=
Granulometry	←	←	←	↑	←	←	↑	=	↑	=	←	↑	←	←	↑	↑	=	←	←	←	←	=	↑	↑		↑	←	=	←	↑	↑	↑	↑	↑	←	↑	=	↑
Gabor feat. on Gauss. col. mod.	←	←	←	=	←	←	↑	←	↑	←	←	↑	←	←	=	=	←	←	←	←	←	=	=	=	←		←	=	←	=	=	=	↑	↑	←	=	=	↑
Wavelets + Co-occurrence hist.	↑	=	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	←	=	←	↑	↑	↑	↑	↑	↑		↑	←	↑	↑	↑	↑	↑	↑	↑	↑	↑
DT-CWT (4 scales, 6 ornts)	←	←	←	↑	←	←	↑	=	↑	←	←	↑	←	←	↑	↑	=	←	←	←	←	=	↑	↑	=	=	←		←	↑	↑	↑	↑	↑	←	↑	↑	↑
Histograms of colour ratios	↑	=	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	←	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑	↑	↑	↑	↑	↑
Opponent Gabor features	←	←	←	=	←	←	=	←	=	←	←	=	←	←	=	=	←	←	←	←	←	←	←	←	=	←	=	←	←	←		=	=	↑	↑	←	=	=	=
Opponent colour LBP	←	←	←	=	←	←	=	←	=	←	←	=	←	←	=	=	←	←	←	←	←	←	←	←	=	←	=	←	←	←	=		=	=	↑	←	=	=	=
Integrative co-occurrence mat.	←	←	←	←	←	←	=	←	↑	←	←	=	←	←	←	←	←	←	←	←	←	←	←	←	=	←	=	←	←	←	=	=		=	↑	←	=	=	=
Colour ranklets	←	←	←	←	←	←	←	←	=	←	←	←	←	←	←	←	←	←	←	←	←	←	←	←	=	←	←	←	←	←	←	=	=		↑	←	←	←	←
Gabor + chrom. feat.	←	←	←	←	←	←	←	←	←	←	←	←	←	←	←	←	←	←	←	←	←	←	←	←	←	←	←	←	←	←	←	←	←	←		←	←	←	←
DT-CWT + chrom. feat.	←	←	←	↑	←	←	↑	↑	↑	=	←	↑	↑	↑	↑	↑	↑	←	←	←	↑	↑	↑	↑	↑	↑	←	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑
Co-occurrence mat. + chrom. feat.	←	←	←	←	←	←	=	←	↑	←	←	=	←	←	←	←	←	←	←	←	←	←	←	←	=	←	=	←	←	←	=	=	=	↑	↑	←		=	=
Co-occurrence mat. + percentiles	←	←	←	←	←	←	↑	←	↑	←	←	=	←	←	←	←	←	←	←	←	←	←	←	←	=	←	=	←	←	←	=	=	=	↑	↑	←	=		=
LBP + percentiles	←	←	←	←	←	←	=	←	↑	←	←	=	←	←	←	←	←	←	←	←	←	←	←	←	=	←	=	←	←	←	=	=	=	↑	↑	←	=	=	

of features of the methods reported in Table III. These values ideally divide the methods into four zones: *A*) high accuracy and low number of features; *B*) high accuracy and high number of features; *C*) low accuracy and low number of features and *D*) low accuracy and high number of features. Ideally the group *A* represents the "optimum", in a Pareto sense, since it combines good accuracy and limited number of features.

As a general trend the good performance of the class `CTD:DisjointSpectralAndGrayscale` emerges. Four out of five methods of this class appear in the ranking (Table III); of the four that appear in the ranking three methods belong to the group *A*; and, last but not least, the most accurate method pertains to this class too.

The methods of the class `CTD:Spectral-MarginalizationIntraAndInterChannel` provide good classification accuracy too, but at the cost of a greater number of features. They tend to cluster in the zone B of the diagram (a remarkable exception is represented by integrative co-occurrence matrices, which

TABLE III: Ranking of the methods. *W*, *L* and *T* stand for wins, losses and ties, respectively. *Bal.* is the difference between the first two. *Rnk* reports the final standing. The table includes only those methods the balance of which is non negative.

	W	L	T	Bal.	Rnk
Gabor + chromatic features	37	0	0	37	1
Three marginal histograms (I1I2I3)	32	1	4	31	2
Colour ranklets	32	1	4	31	3
Three marginal histograms (RGB)	27	3	7	24	4
LBP + percentiles	27	3	7	24	5
Gabor features	24	1	12	23	6
Integrative co-occurrence matrices	25	2	10	23	7
Co-occurrence matrices + chromatic features	25	3	9	22	8
Opponent colour LBP	22	1	14	21	9
Co-occurrence matrices + percentiles	25	4	8	21	10
Marginal histogram (H+V)	24	4	9	20	11
Opponent Gabor features	22	2	13	20	12
Colour histogram	22	9	6	13	13
Col. stat. (mean + std + 3-5th, RGB)	22	9	6	13	14
Col. stat. (mean + std + 3-5th, Lab)	22	9	6	13	15
Gabor feat. on Gauss. col. mod.	19	6	12	13	16
Multilayer CCR, 64 levesl	21	15	1	6	17
Multilayer CCR, 27 levels	16	16	5	0	18
DT-CWT (4 scales, 6 ornts)	16	16	5	0	19

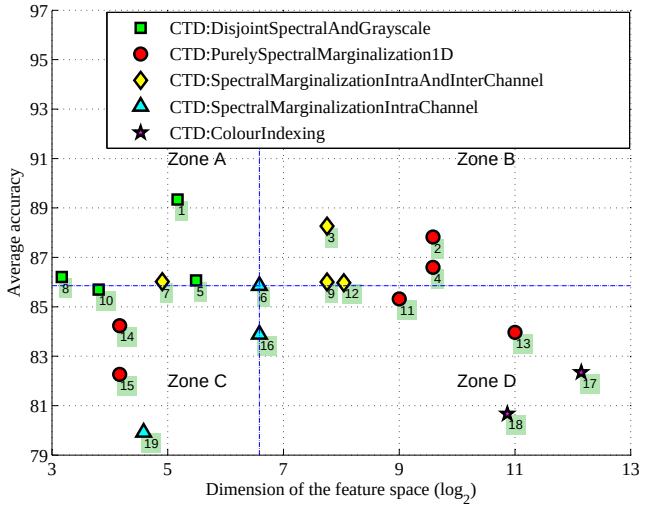


FIG. 4: Accuracy vs dimension of the feature space. Numbers by each symbol indicate position in the ranking (Table III, column "Rnk").

perform well with a low number of features).

Some methods of the class `CTD:PurelySpectral-Marginalization1D` attain good classification accuracy too, but in this case the number of features required is even higher (two colour histograms methods occupy the second and fourth position in the placings, but with a number of features equal to 768). If we ignore the methods of this class that fall in zone *D*, we notice that the remaining methods split into zones *C* and *B*, namely either methods that provide good accuracy with a high number of features (e.g.: three marginal histograms) or leaner methods that provide inferior accuracy (e.g.: colour statistics). The results show that purely spectral features alone remain outside the optimal zone (A), suggesting that combination of textural and colour features is needed to achieve the optimum.

Of the six methods of the class `CTD:ColourIndexing` considered in the experiment, only the Multilayer CCR with a number of levels 27 and 64 appear in the ranking. This method, however, is disadvantaged by the high dimensionality, which makes end it up in zone D.

VI. CONCLUSIONS AND FUTURE DIRECTIONS

The combination of colour and texture into joint descriptors has received significant attention in literature. Nevertheless, despite the good number of methods that have been proposed, little attention has been devoted to the definition of a comprehensive framework to classify them in a unifying scheme. Comparative evaluation of the methods has not been a field of particularly intensive activity either. Based on these motivations with this paper we aimed at providing an overview of the existing methods, defining and appropriate taxonomy and comparing the performance of the methods over a standard dataset. As a preliminary step we presented a taxonomy based on a firm mathematical basis. In establishing this we introduced two different concepts: colour texture function and colour texture descriptor, where the second is defined as a composition of the first. Subsequently we selected an ample set of methods published in literature and gave them the correct ubication inside the taxonomy. We submitted these methods to a comparative experiment using a standard colour texture dataset (OUTEX_TC_00013) and a standard classification approach (nearest-neighbour rule and L_1 distance).

The results point in the direction of separate colour and texture analysis as the best practice when one seeks for optimal compromise between accuracy and limited number of features. This conclusion seems to be in accordance with some recent psychophysiological findings (see Section I) that suggest independent processing of colour and texture in object perception. Good classification accuracy can also be obtained through intra- and inter-channel features, but at the cost of longer feature vectors. Purely spectral methods seem to need even more features to provide comparable results in terms of accuracy.

The above considerations suggest that integrating colour and texture features from parallel processing seems to be the most promising strategy. In the approaches presented in this manuscript this was obtained in the most straightforward way, namely through con-

catenation of feature vectors. Future research could be focussed on smarter approaches to integrate colour and texture data from parallel processing, for instance through fusion of classifiers [40]. Related work in Content-Based Image Retrieval [52] indeed confirms that improved accuracy can be achieved when colour and texture feature vectors are combined through suitable fusion schemes.

VII. NOTE TO THE REVIEWERS

The credentials to access the website http://dismac.dii.unipg.it/ctc are: username = colour, password = texture.

[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*, 23:63–72, 2004.

[2] D. B. Aydogan, M. Hannula, T. Arola, J. Hyttinen, and P. Dastidar. Texture based classification and segmentation of tissues using DT-CWT feature extraction methods. In *Proc. of the 21st IEEE International Symposium on Computer-Based Medical Systems*, pages 614–619, Jyväskylä, Finland, June 2008.

[3] J.A. Bangham, R. Harvey, and P.D. Ling. Morphological scale-space preserving transforms in many dimensions. *Journal of Electronic Imaging*, 5:283–299, 1996.

[4] M. E. Barilla and M. Spann. Colour-based texture image classification using the Complex Wavelet Transform. In *Proc. of the 5th International Conference on Electrical Engineering, Computing Science and Automatic Control*, Mexico City, Mexico, November 2008.

[5] M.H. Bharati, J.J. Liu, and J.F. MacGregor. Image texture analysis: methods and comparisons. *Chemometrics and intelligent laboratory systems*, 72:57–71, 2004.

[6] F. Bianconi and A. Fernández. Evaluation of the effects of Gabor filter parameters on texture classification. *Pattern Recognition*, 40:3325–3335, 2007.

[7] F. Bianconi, A. Fernández, E. González, and J. Armesto. Robust color texture features based on ranklets and discrete Fourier transform. *Journal of Electronic Imaging*, 18:043012–1–8, 2009.

[8] F. Bianconi, A. Fernández, E. González, D. Caride, and A. Calviño. Rotation-invariant colour texture classification through multilayer CCR. *Pattern Recognition Letters*, 30:765–773, 2009.

[9] F. Bianconi, A. Fernández, and A. Mancini. Assessment of rotation-invariant texture classification through Gabor filters and Discrete Fourier Transform. In *Proc. of the 20th International Congress on Graphical Engineering*, Valencia, Spain, June 2008.

[10] J.S. Cant, M.E. Large, L. McCall, and M.A. Goodale. Independent processing of form, colour, and texture in object perception. *Perception*, 37:57–78, 2008.

[11] B. Caputo, E. Hayman, and P. Mallikarjuna. Class-specific material categorisation. In *Proceedings of the 10th International Conference on Computer Vision*, volume 2, pages 1597–1604, Beijing, China, 2005.

[12] C. Cavina-Pratesi, R.W. Kentridge, C.A. Heywood, and A.D. Milner. Separate channels for processing form, texture, and color: Evidence from fMRI adaptation and visual object agnosia. *Cerebral Cortex*, 20:2319–2332, 2010.

[13] R.W. Conners and C.A. Harlow. A theoretical comparison of texture algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2:204–222, 1980.

[14] K.J. Dana, B. van Ginneken, S.N. Nayar, and J.J. Koenderink. Reflectance and texture of real-world surfaces. *ACM Transactions on Graphics*, 18:1–34, 1999.

[15] J.G. Daugman. Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *Journal of the Optical Society of America. A, Optics and image science*, 2:1160–1169, 1985.

[16] K.S. Deshmukh. Color image sementation: A review. In *Proceedings of SPIE - Second International Conference on Digital Image Processing*, Singapore, February 2010.

[17] A. Drimbarean and P.F. Whelan. Experiments in colour texture analysis. *Pattern Recognition Letters*, 22:1161–1167, 2001.

[18] R.O. Duda, P.E. Hart, and D.G. Stork. *Pattern Classification*. John Wiley & Sons, 2001.

[19] A. Fernández, O. Ghita, E. González, F. Bianconi, and P.F. Whelan. Evaluation of robustness against rotation of LBP, CCR and ILBP features in granite texture classification. *Machine Vision and Applications*, 2010. In press.

[20] B.V. Funt and G.D. Finlayson. Color constant color indexing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17:522–528, 1995.

[21] T. Gevers and A. W. M. Smeulders. A comparative study of several color models for color image invariant retrieval. In *Proceedings of the first international workshop on Image databases and Multi Media Search*, pages 17–26, Amsterdam, The Netherlands, August 1996.

[22] S. Gibson, R. Harvey, and G. Finlayson. Convex colour sieves. *Lecture Notes in Computer Science*, 2695:550–563, 2003.

[23] R.C. González, R.E. Woods, and S.L. Eddins. *Digital Image Processing Using MATLAB*. Pearson, 2004.

[24] A. Hanbury, U. Kandaswamy, and D.A. Adjeroth. Illumination-invariant morphological texture classification. In *Proceedings of the 7th International Symposium on Mathematical Morphology*, Paris, France, April 2005.

[25] R.M. Haralick, K. Shanmugam, and I. Dinstein. Textural fetaures for image classification. *IEEE Transactions on Systems, Man, and Cybernetics*, 3:610–621, 1973.

[26] M. Hauta-Kasari, J. Parkkinen, T. Jaaskelainen, and R. Lenz. Generalized cooccurrence matrix for multispectral texture analysis. In *Proceedings of the 13th International Conference on Pattern Recognition*, volume 2, pages 785–789, Vienna, Austria, August 1996.

[27] E. Hayman, B. Caputo, M. Fritz, and J.O. Eklundh. On the significance of real-world conditions for material classifications. *Lecture Notes in Computer Science*,

3024:253–266, 2004.

[28] P.S. Hiremath, S. Shivashankar, and J. Pujari. Wavelet based features for color texture classification with application to CBIR. *International Journal of Computer Science and Network Security*, 6:124–133, 2006.

[29] M.H. Hoang, J. Geusebroek, and A.W.M. Smeulders. Color texture measurement and segmentation. *Signal Processing*, 85:265–275, 2005.

[30] D. Iakovidis, D. Maroulis, and S. Karkanis. A comparative study of color-texture image features. In *Proceedings of the 12th International Workshop on Systems, Signals & Image Processing*, pages 203–207, Chalkida, Greece, September 2005.

[31] D.E. Ilea and P.F. Whelan. Image segmentation based on the integration of colour texture descriptors: a review. *Pattern Recognition*, 44:2479–2501, 2011.

[32] A. Jain and G. Healey. A multiscale representation including opponent color features for texture recognition. *IEEE Transactions on Image Processing*, 7:124–128, 1998.

[33] B. Julesz. Textons, the elements of texture perception, and their interactions. *Nature*, 290:91–97, 1981.

[34] G.K. Kanji. *100 Statistical Tests*. SAGE Publications, 1999.

[35] S. Karkanis, K. Galousi, and D. Maroulis. Classification of endoscopic images based on texture spectrum. In *Proc. of Workshop on Machine Learning in Medical Applications, Advance Course in Artificial Intelligence*, pages 63–69, Chania, Greece, July 1999.

[36] N. Kingsbury. Image processing with complex wavelets. *Philosophical Transactions of the Royal Society A: Physical, Mathematical and Engineering Sciences*, 357:2543–2560, 1999.

[37] K. Konstantinidis, A. Gasteratos, and I. Andreadis. Image retrieval based on fuzzy color histogram processing. *Optics Communications*, 248:375–386, 2005.

[38] S. Kukkonen, H. Kälviäinen, and J. Parkkinen. Color features for quality control in ceramic tile industry. *Optical Engineering*, 40:170–177, 2001.

[39] R. Pradeep Kumar and P. Nagabhusan. Multiresolution knowledge mining using wavelet transform. *Engineering Letters*, 14:EL–14–1–30, 2007.

[40] L.I. Kuncheva. *Caombining Pattern Classifiers. Methods and Algorithms.* Wiley Interscience, 2004.

[41] E.V. Kurmyshev and M. Cervantes. A quasi-statistical approach to digital binary image representation. *Revista Mexicana de Física*, 42:104–116, 1996.

[42] E.V. Kurmyshev and R.E. Sánchez-Yañez. Comparative experiment with colour texture classifiers using the CCR feature space. *Pattern Recognition Letters*, 26:1346–1353, 2005.

[43] E.V. Kurmyshev, R.E. Sánchez-Yañez, and A. Fernandez. Colour texture classification for quality control of polished granite tiles. In *Proceedings of IASTED Visualization, Imaging and Image Processing*, Benalmádena, Spain, September 2003.

[44] D. Lafon and T. Ramananantoandro. Color images. *Image Analysis and Stereology*, 21:61–74, 2002.

[45] F. Lahajnar and S. Kovacic. Rotation-invariant texture classification. *Pattern Recognition Letters*, 24:1151–1161, 2003.

[46] L. Lepistö, I. Kunttu, and A. Visa. Color-based classification of natural rock images using classifier combinations. *Lecture Notes in Computer Science*, 3540:901–909, 2005.

[47] L. Lepistö, I. Kunttu, and A. Visa. Rock image classification using color features in Gabor space. *Journal of Electronic Imaging*, 14, 2005.

[48] B. Li and M.Q.-H. Meng. Capsule endoscopy images classification by color texture and support vector machine. In *Proc. of the IEEE International Conference on Automation and Logistics*, pages 126–131, Hong Kong / Macau, August 2010.

[49] S. Li and J. Shawe-Taylor. Comparison and fusion of multiresolution features for texture classification. *Pattern Recognition Letters*, 26:633–638, 2005.

[50] F. López, J.M. Prats, A. Ferrer, and J.M. Valiente. Defect detection in random colour textures using the MIA t^2 defect maps. *Lecture Notes in Computer Science*, 4142:752–763, 2006.

[51] F. López, J.M. Valiente, R. Baldrich, and M. Vanrell. Fast surface grading using color statistics in the CIE Lab space. *Lecture Notes in Computer Science*, 3523:666–673, 2005.

[52] M.J. Álvarez, E. González, F. Bianconi, J. Armesto, and A. Fernández. Colour and texture features for image retrieval in granite industry. *Dyna*, 77:121–130, 2010.

[53] R. Manthalkar, P. K. Biswas, and B. N. Chatterji. Rotation invariant color texture classification in perceptually uniform color spaces. In *Proceedings of the 3rd Indian Conference on Computer Vision, Graphics and Image Processing*, Ahmadabad, India, December 2002.

[54] A.N. Marana, L.F. Costa, R.A. Lotufo, and S.A. Velastin. On the efficacy of texture analysis for crowd monitoring. In *Proceedings of the International Symposium on Computer Graphics, Image Processing, and Vision*, pages 354 –361, October 1998.

[55] M. Masotti. A ranklet-based image representation for mass classification in digital mammograms. *Medical Physics*, 33:3951–3961, 2006.

[56] M. Masotti and R. Campanini. Texture classification using invariant ranklet features. *Pattern Recognition Letters*, 29:1980–1986, 2008.

[57] F. Mendoza, P. Dejmek, and J.M. Aguilera. Colour and image texture analysis in classification of commercial potato chips. *Food Research International*, 40:1146–1154, 2007.

[58] F. Mendoza, N.A. Valousa, P. Allenb, T.A. Kennyb, P. Wardb, and Da-Wen Sun. Analysis and classification of commercial ham slice images using directional fractal dimension features. *Meat Science*, 81:313–320, 2009.

[59] T. Mäenpää and M. Pietikäinen. Classification with color and texture: jointly or separately? *Pattern Recognition*, 37, 2004.

[60] T. Mäenpää and M. Pietikäinen. Texture analysis with Local Binary Patterns. In C.H. Chen, L.F. Pau, and P.S.P. Wang, editors, *Handbook of Pattern Recognition and Computer Vision (2nd Edition)*, pages 197–216. World Scientific Publishing, 2005.

[61] B. Meyer. The many faces of inheritance: a taxonomy of taxonomy. *Computer*, 29:105–108, 1996.

[62] M. Mirmehdi and M. Petrou. Segmentation of color textures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:142–159, 2000.

[63] P. Nammalwar, O. Ghita, and P.F. Whelan. A generic framework for texture segmentation. *Sensor Review*, 30:69–79, 2010.

[64] M. Niskanen, O. Silvén, and H. Kauppinen. Color

and texture based wood inspection with non-supervised clustering. In *Proceedings of the 12th Scandivanian Conference on Image Analysis*, pages 336–342, Bergen, Norway, June 2001.

[65] T. Ojala, M. Pietikäinen, and T. Mäenpää. Multiresolution gray-scale and rotation invariant texture classification with Local Binary Patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24:971–987, 2002.

[66] T. Ojala, M. Pietikäinen, T. Mäenpää, 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*, Quebec (Canada), 2002.

[67] W.S. Ooi and C.P. Lim. Fusion of colour and texture features in image segmentation: An empirical study. *Imaging Science Journal*, 57:8–18, 2009.

[68] R. Paget and I.D. Longstaff. Open-ended texture classification for terrain mapping. In *Proc. of the International Conference on Image Processing*, volume 3, pages 584–587, 2000.

[69] C. Palm. Color texture classification by integrative cooccurrence matrices. *Pattern Recognition*, 37:965–976, 2004.

[70] G. Paschos. Fast color texture recognition using chromaticity moments. *Pattern Recognition Letters*, 21:837–841, 2000.

[71] G. Paschos. Perceptually uniform color spaces for color texture analysis: An empirical evaluation. *IEEE Transactions on Image Processing*, 10:932–937, 2001.

[72] N. Pican, E. Trucco, M. Ross, D.M. Lane, Y. Petillot, and I.T. Ruiz. Texture analysis for seabed classification: Co-occurrence matrices vs self-organizing maps. In *Proc. of IEEE Conference and Exhibition on Engineering for Sustainable Use of the Oceans*, Nice, France, September-October.

[73] M. Pietikäinen, T. Mäenpää, and J. Viertola. Color texture classification with color histograms and local binary patterns. In *Proceedings of the 2nd International Workshop on Texture Analysis and Synthesis*, pages 109–112, Copenhagen, Denmark, June 2002.

[74] M. Pietikäinen, S. Nieminen, E. Marszalec, and T. Ojala. Accurate color discrimination with classification based on features distributions. In *Proceedings of the 13th International Conference on Pattern Recognition*, volume 3, pages 833–838, Vienna, Austria, August 1996.

[75] A. Porebski, N. Vandebrouke, and L. Macaire. Comparison of feature selection schemes for color texture classification. In *Proceedings of the 2nd International Conference on Image Processing Theory, Tools and Applications*, pages 32–37, Paris, France, July 2010.

[76] T. Randen and J. Håkon Husøy. Filtering for texture classification: A comparative study. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21:291–310, 1999.

[77] R. Schalkoff. *Pattern Recognition. Statistical, Structural and Neural Approaches*. John Wiley & Sons, 1992.

[78] O. Sertel, J. Kong, G. Lozanski, A. Shana'ah, U. Catalyurek, J. Saltz, and M. Gurcan. Texture classification using nonlinear color quantization: application to histopathological image analysis. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Image Processing*, pages 597–600, March-April 2008.

[79] M. Singh, A. Javadi, and S. Singh. A comparison of texture features for the classification of rock images. *Lecture Notes in Computer Science*, 3177:179–184, 2004.

[80] F. Smeraldi. Ranklets: orientation selective nonparametric features applied to face detection. In *Proceedings of the 16th International Conference on Pattern Recognition (ICPR'02)*, volume 3, pages 379–382, Washington DC, USA, 2002. IEEE Computer Society.

[81] R.E. Sánchez-Yáñez, E.V. Kurmyshev, and F.J. Cuevas. A framework for texture classification using the Coordinated Clusters Representation. *Pattern Recognition Letters*, 24:21–31, 2003.

[82] P. Southam and R. Harvey. Texture classification via morphological scale-space: Tex-Mex features. *Journal of Electronic Imaging*, 18:043007–1–16, 2009.

[83] E.W. Steyerberg, F.E. Harrell, G.J.J.M. Borsboom, M.J.C. Eijkemans, Y. Vergouwe, and J.D.F. Habbema. Internal validation of predictive models: Efficiency of some procedures for logistic regression analysis. *Journal of Clinical Epidemiology*, 54:774–781, 2001.

[84] M.J. Swain and D.H. Ballard. Color indexing. *International Journal of Computer Vision*, 7:11–32, 1991.

[85] The Object Management Group. Unified Modeling Language. Version 2.3, May 2010. Available at: http://www.omg.org/spec/UML/2.3/.

[86] O. Tuzel, L. Yang, P. Meer, and D.J. Foran. Classification of hematologic malignancies using texton signatures. *Pattern Analysis and Applications*, 10:277–290, 2007.

[87] Various Authors. Matlab implementation of wavelet transforms. Available at http://taco.poly.edu/WaveletSoftware/. Accessed on Dec 29, 2010.

[88] Various Authors. SIMPLEGABOR - Simple Gabor Feature Space. Available at http://www2.it.lut.fi/project/simplegabor/. Accessed on Dec 29, 2010.

[89] M. Varma and A. Zisserman. Texture classification: Are filter banks necessary? In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 691–698, Madison, USA, June 2003.

[90] C. Vertan and N. Boujemaa. Color texture classification by normalized color space representation. In *Proceedings of the 15th International Conference on Pattern Recognition*, volume 3, pages 580–583, Barcelona, Spain, September 2000.

[91] H. Wackernagel. *Multivariate Geostatistics. An Introduction with Applications*. Springer, 2003.

[92] J.S. Weszka, C.R. Dyer, and A. Rosenfeld. A comparative study of texture measures for terrain classification. *IEEE Transactions on Systems, Man and Cybernetics*, 6(4):269–285, 1976.

[93] P.F. Whelan and O. Ghita. Colour texture analysis. In M. Mirmehdi, X. Xie, and J. Suri, editors, *Handbook of texture analysis*, pages 129–163. Imperial College Press, 2008.

[94] G. Wyszecki and W.S. Styles. *Color Science. Concepts and Methods, Quantitative Data and Formulae. Second Edition*. Wiley-Interscience, 1982.

[95] X. Xie. A review of recent advances in surface defect detection using texture analysis techniques. *Electronics Letters on Computer Vision and Image Analysis*, 7:1–22, 2008.

[96] X. Xie and M. Mirmehdi. A galaxy of texture features. In M. Mirmehdi, X. Xie, and J. Suri, editors, *Handbook of texture analysis*, pages 375–406. Imperial College Press, 2008.

[97] X. Yuan, Z. Yang, G. Zouridakis, and N. Mullani. SVM-based texture classification and application to early melanoma detection. In *Proc. of the 28th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 4775–4778, New York City, USA, September 2006.

[98] http://www.outex.oulu.fi/index.php?page=test_suites

[99] http://www.uea.ac.uk/is/it/ResearchComputing/HPCCluster

[100] http://dismac.dii.unipg.it/ctc